Subhash Bhalla (Ed.)

LNCS 3433

Databases in Networked Information Systems

4th International Workshop, DNIS 2005
Aizu-Wakamatsu, Japan, March 2005
Proceedings

Springer

Lecture Notes in Computer Science 3433

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Subhash Bhalla (Ed.)

Databases in Networked Information Systems

4th International Workshop, DNIS 2005
Aizu-Wakamatsu, Japan, March 28-30, 2005
Proceedings

Springer

Volume Editor

Subhash Bhalla
University of Aizu
Department of Computer Software, Database Systems Laboratory
Tsuruga, Ikki Machi, Aizu-Wakamatsu City, Fukushima, 965-8580, Japan
E-mail: bhalla@u-aizu.ac.jp

Library of Congress Control Number: Applied for

CR Subject Classification (1998): H.2, H.3, H.4, H.5, C.2

ISSN 0302-9743
ISBN 3-540-25361-0 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11401339 06/3142 5 4 3 2 1 0

Preface

Information systems in healthcare and public utility services depend on computing infrastructures. Many research efforts are being made in related areas, such as wireless computing (e.g., Auto-ID Laboratories, and projects at MIT), Web-based computing, and information accesses by Web users. Government agencies in many countries plan to launch facilities in education, healthcare and information support as a part of e-government initiatives. In this context, information interchange management has become an active research field. A number of new opportunities have evolved in design and modeling based on new computing needs of users. Database systems play a central role in supporting networked information systems for access and storage management aspects.

The 4th International Workshop on Databases in Networked Information Systems (DNIS 2005) was held on March 28–30, 2005 at the University of Aizu in Japan. The workshop program included research contributions, and invited contributions. A view of research activity in Information Interchange Management and related research issues was provided by the session on this topic. The invited contribution was contributed by Dr. Umeshwar Dayal. The workshop session on Web Data Management Systems had invited papers by Prof. Elisa Bertino, Prof. Masahito Hirakawa, and Prof. William I. Grosky. The two sessions on Networked Information Systems included invited contributions by Prof. Sushil Jajodia, Dr. Cyrus Shahabi, Prof. Divyakant Agrawal and Dr. Harumi Kuno. I would like to thank the members of the Program Committee for their support and all authors who considered DNIS 2005 in making research contributions.

The sponsoring organizations and the Organizing Committee deserve praise for the support they provided. A number of individuals contributed to the success of the workshop. I thank Dr. Umeshwar Dayal, Prof. J. Biskup, Prof. D. Agrawal, Prof. Elisa Bertino, Dr. Cyrus Shahabi, Prof. Sushil Jajodia and Dr. Mark Sifer for providing continuous support and encouragement.

The workshop received invaluable support from the University of Aizu. In this context, I thank Prof. Ikegami, President of the University of Aizu. I thank Prof. Daming Wei, Head of the Department of Computer Software, for making the support available. I express my gratitude to the members and chairman of the International Affairs Committee, for supporting the workshop proposal. Many thanks are also due to the faculty members at the university for their cooperation and support.

March 2005 Subhash Bhalla

Organization

The DNIS 2005 international workshop was organized by the Database Systems Laboratory, University of Aizu, Aizu-Wakamatsu City, Fukushima, 965-8580, Japan.

Executive Committee

Honorary Chair	T. L. Kunii, Hosei University, Japan
Program Chair	S. Bhalla, University of Aizu, Japan
Organizing Chair	Hui Wang, University of Aizu, Japan
Executive Chair	N. Bianchi-Berthouze, University of Aizu, Japan

Program Committee

D. Agrawal, University of California, USA
F. Andres, National Center for Science Information Systems, Japan
N. Berthouze, University of Aizu, Japan
E. Bertino, Purdue University, USA
S. Bhalla, University of Aizu, Japan
P.C.P. Bhatt, Indian Institute of Information Technology, Bangalore, India
J. Biskup, University of Dortmund, Germany
L. Capretz, University of Western Ontario, Canada
M. Capretz, University of Western Ontario, Canada
U. Dayal, Hewlett-Packard Laboratories, USA
W.I. Grosky, University of Michigan-Dearborn, USA
J. Herder, University of Applied Sciences, Fachhochschule Dusseldorf, Germany
M. Hirakawa, Shimane University, Japan
Q. Jin, University of Aizu, Japan
A. Kumar, Pennsylvania State University, USA
H. Kuno, Hewlett-Packard Laboratories, USA
J. Li, University of Tsukuba, Japan
K. Myszkowski, Max-Planck-Institut fuer Informatik, Germany
T. Nishida, Kyoto University, Japan
A. Pasko, Hosei University, Tokyo, Japan
L. Pichl, University of Aizu, Japan
V. Savchenko, Hosei University, Japan
C. Shahabi, University of Southern California, USA
M. Sifer, Sydney University, Australia
H. Wang, University of Aizu, Japan

Sponsoring Institution

International Affairs Committee, University of Aizu,
Aizu-Wakamatsu City, Fukushima, 965-8580, Japan

Table of Contents

Information Interchange and Management Systems

Web Data Management Systems

Networked Information Systems: Applications

Networked Information Systems: Implementations

Data Management Systems

Predictive Business Operations Management

Malu Castellanos, Norman Salazar, Fabio Casati,
Umesh Dayal, and Ming-Chien Shan

Hewlett-Packard,
1501 Page Mill Road, MS 1142,
Palo Alto, CA, 94304, USA
firstanme.lastname@hp.com

Abstract. Having visibility into the current state of business operations doesn't seem to suffice anymore. The current competitive market forces companies to capitalize on any opportunity to become as efficient as possible. The ability to forecast metrics and performance indicators is crucial to do effective business planning, the benefits of which are obvious – more efficient operations and cost savings, among others. But achieving these benefits using traditional forecasting and reporting tools and techniques is very difficult. It typically requires forecasting experts who manually derive time series from collected data, analyze the characteristics of such series and apply appropriate techniques to create forecasting models. However, in an environment like the one for business operations management where there are thousands of time series, manual analysis is impractical, if not impossible. Fortunately, in such an environment, extreme accuracy is not required; it is usually enough to know whether a given metric is predicted to exceed a certain threshold or not, is within some specified range or not, or belongs to which one of a small number of specified classes. This gives the opportunity to automate the forecasting process at the expense of some accuracy. In this paper, we present our approach to incorporating time series forecasting functionality into our business operations management platform and show the benefits of doing this.

1 Introduction

The current competitive market situation where only the best can survive has triggered the radar for opportunities to become more efficient than the competitors. One such opportunity is given by good business planning with the goal of better meeting business objectives. Having visibility into the past and current state of business operations, as traditionally done by monitoring applications, is not enough anymore. More and more, there is a pressing need to project future performance. In particular, the ability to forecast metrics and performance indicators is crucial to do effective business planning, the benefits of which are obvious – more efficient operations and cost savings, among others. In consequence, forecasting (or *predictive analytics*, as it is now being called in the industry) is increasingly being considered by companies as a key technology to remain or become competitive. We are witnessing this trend (along with a mindset change in high-level managers to embrace prediction

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 1 – 14, 2005.

technology) in virtually every domain, from telecom to supply chain, from transportation to finance. For example, transportation companies need to forecast the number of packages they will need to dispatch from a certain hub (or at least whether that number will exceed a given threshold) to plan for trucks. Correct planning allows more efficient shipments, avoids delays and saves costs. In a nutshell, forecasting provides the foundation for good planning inside and across organizations. The need for forecasting tools is so high that companies are willing to spend millions of dollars in consulting efforts to obtain these forecasts.

Traditionally, forecasting has been a human-driven task where domain forecasting experts interactively massage historical data, analyze it, try different algorithms, iterate over this cycle and finally come up with a forecasting model. This however, is absolutely impractical when there is the need of forecasting many different metrics and when the business conditions change dynamically so that prediction models need to be updated. In this situation automating the forecasting process seems to be the only practical solution. The problem is that automation can only come at the expense of accuracy and therefore in applications where extreme accuracy is required, there is no other way than to continue with the human-driven process whose output is expected to be a perfectly fit model. For example, accuracy is extremely important when providing forecasts of a company's revenue to financial analysts, because even a slight deviation of the actual revenue delivered from the forecast can have severe business consequences such as a drop in the company's stock price.

Fortunately, there are other applications, like the one discussed in this paper, where such great accuracy is not required and an approximate prediction model obtained quickly and automatically is preferred to a more accurate model obtained after long, expensive consulting efforts. In this case, automation of the forecasting process seems to be appropriate. For example, when metrics defined over business processes have thresholds that indicate undesired business operations behavior, forecasts that indicate whether future values are likely to be in the red, yellow or green zones are enough to decide the next action to take, for instance, proactively trying to prevent a metric from reaching a given value. As can be observed in situations like this, the predictions have to be made with reasonable confidence, but 100% accuracy is not required.

The goal of our work is that of developing a *completely automated* forecasting solution that could endow operation monitoring tools with the capability of providing reasonable accurate, and up-to-date predictions. Our work is characterized by three basic ingredients:

i) We have bundled and integrated into a single software application multiple algorithms that cover the entire forecasting lifecycle. Whenever a prediction is needed, our software identifies and applies all the appropriate algorithms, evaluates the confidence with which each prediction is made, and selects the most accurate one.

ii) We have developed new algorithms and use state of the art optimization technologies to enable complete automation with efficiency and performance improvement.

iii) We have architected the algorithms into a forecasting engine (G-TSFE) that can be embedded into monitoring and management software so that, whenever users monitor metrics, as long as the monitoring system keeps a log of metrics values, they can also access forecasts for such metrics. For example, if the transportation company has a monitoring system that tracks daily shipments, then the forecasting engine can be embedded into this software to make predictions for shipments and any other metric monitored by the same system. We are planning to integrate this engine into various HP management products, starting with OpenView Business Process Insight (OV-BPI) [1] to provide forecasting on business process-related metrics.

In this paper we focus on business process monitoring applications to illustrate the concepts. In this environment, there are two kinds of prediction that are of interest:

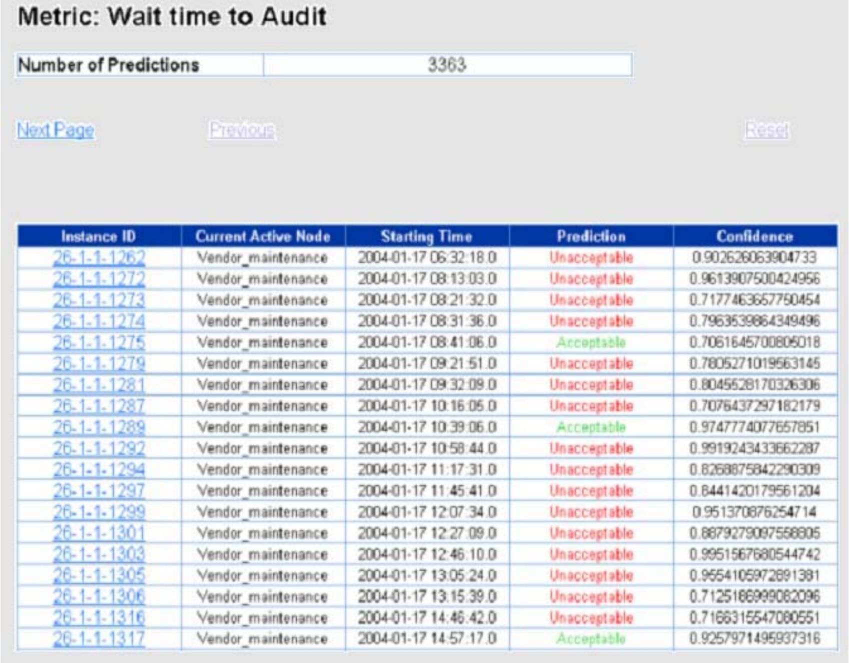
a) Predictions of metric values that active process instances will have at the end of their execution. For example, the total duration of processing an order currently in the middle of execution. Even though this kind of prediction is not the main topic of the paper, we briefly address it for completing the picture.
b) Predictions of aggregated metric values of future instances. For example, the number of orders that will be placed next Monday or even their average processing time. This is the main focus of the paper.

The remainder of the paper is organized as follows. Section 2 gives an example of the usage of an automated prediction solution in a business process monitoring environment. Section 3 gives a technical overview of our prediction engine concentrating more on its forecasting functionality. Section 4 summarizes the competitive landscape, and in Section 5 we conclude with the current state and next steps.

2 Example

Let's assume that one of the business objectives of a hypothetical manufacturing company ACME is to gain a good reputation for being punctual in making payments to its suppliers. This objective will allow ACME to be in a favorable position to negotiate with vendors the terms of the payment period in their contracts. The business operation on which this objective depends is "Pay to Vendor" and the process underlying this operation is called "Payables". Let's assume also that ACME has a 10 days payment policy from the moment it receives a vendor invoice. For ACME to keep its good standing it is important to keep monitoring the duration of the Payables process and react if the process is not meeting the payment policy. However, monitoring the current state of the process is not enough, since it only gives opportunity to "react" after the problem has occurred, which is too late. What ACME needs is the ability to be "proactive" so that problems can be prevented. Specifically, it needs to leverage its monitoring platform with predictive functionality. Two kinds of prediction are useful in this context:

a) For invoices that are currently being executed, ACME will be able to have predictions about their total duration. This will enable ACME to take action before the problem arises in order to prevent its occurrence or at least minimize the damage caused. In this case, try to speed up the processing of those invoices likely to last more than 10 days, for example, by increasing their priority. This kind of prediction is *per instance,* as illustrated in Figure 1 where those instances (each instance corresponding to the processing of an invoice) predicted to exceed the maximum duration say “unacceptable” (in red but not noticeable in this black and white paper). Notice that each prediction is accompanied by a confidence value.



Metric: Wait time to Audit

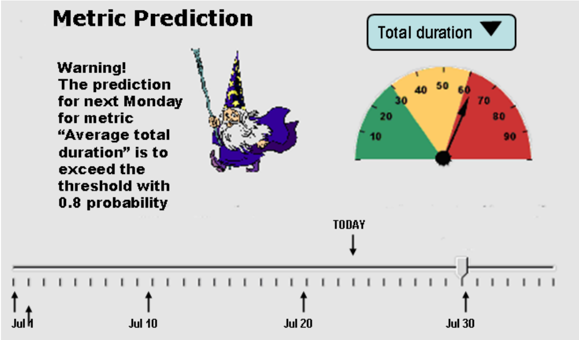
Number of Predictions	3363

Instance ID	Current Active Node	Starting Time	Prediction	Confidence
26-1-1-1262	Vendor_maintenance	2004-01-17 06:32:18.0	Unacceptable	0.902626063904733
26-1-1-1272	Vendor_maintenance	2004-01-17 08:13:03.0	Unacceptable	0.961390750042495
26-1-1-1273	Vendor_maintenance	2004-01-17 08:21:32.0	Unacceptable	0.717746365775045
26-1-1-1274	Vendor_maintenance	2004-01-17 08:31:36.0	Unacceptable	0.796353986434949
26-1-1-1275	Vendor_maintenance	2004-01-17 08:41:06.0	Acceptable	0.706164570080501
26-1-1-1279	Vendor_maintenance	2004-01-17 09:21:51.0	Unacceptable	0.780527101956314
26-1-1-1281	Vendor_maintenance	2004-01-17 09:32:09.0	Unacceptable	0.804552817032630
26-1-1-1287	Vendor_maintenance	2004-01-17 10:16:05.0	Unacceptable	0.707643729718217
26-1-1-1289	Vendor_maintenance	2004-01-17 10:39:06.0	Acceptable	0.974777407765785
26-1-1-1292	Vendor_maintenance	2004-01-17 10:58:44.0	Unacceptable	0.991924343366228
26-1-1-1294	Vendor_maintenance	2004-01-17 11:17:31.0	Unacceptable	0.826887584229030
26-1-1-1297	Vendor_maintenance	2004-01-17 11:45:41.0	Unacceptable	0.844142017956120
26-1-1-1299	Vendor_maintenance	2004-01-17 12:07:34.0	Unacceptable	0.951370876254714
26-1-1-1301	Vendor_maintenance	2004-01-17 12:27:09.0	Unacceptable	0.887927909755880
26-1-1-1303	Vendor_maintenance	2004-01-17 12:46:10.0	Unacceptable	0.995156768054474
26-1-1-1305	Vendor_maintenance	2004-01-17 13:05:24.0	Unacceptable	0.955410597289138
26-1-1-1306	Vendor_maintenance	2004-01-17 13:15:39.0	Unacceptable	0.712518699908209
26-1-1-1316	Vendor_maintenance	2004-01-17 14:46:42.0	Unacceptable	0.716631554708055
26-1-1-1317	Vendor_maintenance	2004-01-17 14:57:17.0	Acceptable	0.925797149593731

Fig. 1. Predictions of metric Wait Time to Audit for active process instances

b) ACME needs to know whether the average total duration of processing invoices the next day (or week, or month) is likely to be above a certain threshold in order to make plans for the resources that it will assign to this process. The idea is to provide ACME with the ability to monitor the daily (or weekly, or monthly) average total duration not only for the present and past, but also for the future. This is illustrated in Figure 2, shows a possible GUI where a user can move the slider along the time scale to set the time for the metric prediction, which is displayed on the dial chart. This kind of prediction is more in the style of Time Series Forecasting.

Instance prediction is specific to a business process monitoring environment, while time series forecasting applies to any monitoring environment.



Metric Prediction

Warning! The prediction for next Monday for metric "Average total duration" is to exceed the threshold with 0.8 probability

Fig. 2. Forecasting total duration metric average for the time set on the slider

It can be appreciated that these two kinds of prediction functionality empower business managers to do effective planning and decision making to better meet the goals of their business operations. They constitute a powerful tool that leverages the business process monitoring platform and gives its users the ability to see the metrics' most probable future behavior.

One of the main features of such a tool is that its design responds to the need that users have for a prediction tool that is easy and intuitive to use. The user doesn't need to write code, or be an expert in predictive technologies; rather she simply clicks a button and obtains the requested predictions.

3 Technical Overview

In this section, we explain the architecture, main concepts and principles underlying the prediction functionality of *Metric Predictor* (MP), our prediction engine to automate the prediction of metric values in a monitoring environment.

Metric Predictor has been designed with three goals in mind: ease-of-use, generality and scalability. It is

- easy to use so that business managers do not need to be experts in data mining or statistics, but can simply obtain forecasts at the click of a button;
- generic so that it works for any kind of metric in different domains; and
- scalable so that it can handle large numbers of frequent predictions.

3.1 System Architecture

Metric Predictor is an engine that can easily be integrated into a monitoring platform, enhancing it with predictive functionality. In general terms, a monitoring platform that supports metrics receives its input from external instrumentation infrastructure, allows its users to define metrics on top of data fed by the instrumentation, computes metric values from this data according to the metric definitions, and provides reports to its users. Seamlessly integrating Metric Predictor into such a platform essentially means tailoring it for the platform repository schema, creating the interfaces to let it read and write to and from the repository, and extending this repository with the necessary data structures for the prediction models as well as for the predictions themselves. Figure 3 shows the overall architecture of a monitoring platform enriched with *predictive* intelligence.

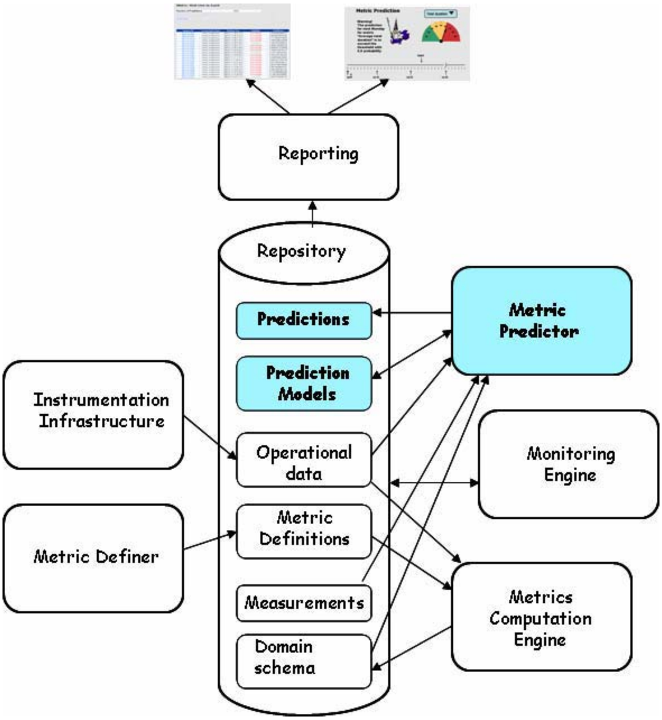


Fig. 3. General architecture of a predictive monitoring platform

In the case of a business process monitoring platform such as OV-BPI, whose task is to track the progress of process executions, the instrumentation infrastructure captures events that signal the start and end of the activities corresponding to the different process steps, as shown in Figure 4. For this purpose the platform allows its users to define abstract views of business processes as well as events and their mappings to the start and end of the steps of such abstract process views. In addition, it lets users define metrics on top of these process views. Metrics are computed on the process execution data that the monitoring engine logs after interpreting the occurrence of events according to its mappings.

In the rest of the paper, we concentrate on a Metric Predictor whose development was motivated by the need to incorporate prediction capabilities into our business operations management platform, *Business Cockpit* (built on top of OV-BPI). Details about other functionalities of this platform can be found in [2].

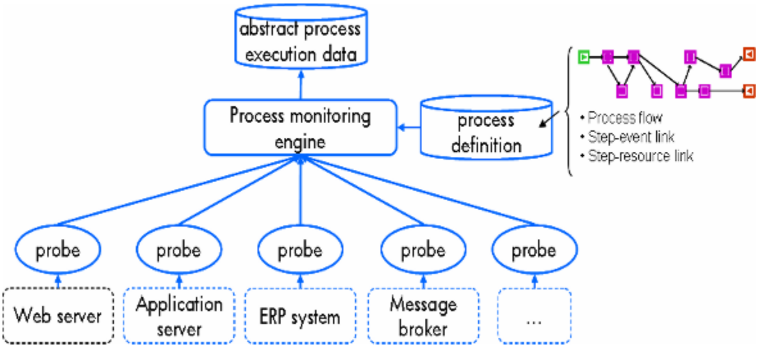


Fig. 4. Process monitoring instrumentation

Metrics Predictor consists of two main modules whose functionalities correspond to the two kinds of predictions explained in the previous section:

a) *Instance Predictor Module* (IPM), applies data mining techniques on execution data of previously executed process instances to learn patterns and embed them into prediction models. As a given process instance progresses in its execution, the appropriate model can be retrieved and applied to its execution data to predict its future value for the corresponding metric.

b) *Generic Time Series Forecasting Module* (G-TSFM), incorporates proprietary and improved techniques to automate the forecasting of time series. Forecasting models are generated from historical time series of aggregated metric values. At any moment a forecasting model can be applied to predict the value of an aggregated metric at a given time.

3.2 Instance-Based Prediction

IPM, the instance-based prediction module, makes data mining accessible to the business user. Most data mining tools require considerable experience to interact with the tool throughout the entire data mining process lifecycle, i.e., to select the relevant attributes (features), to select the technique, to tune the parameters, and so on. However, IPM being aware of the data semantics has been tailored for the specific business process environment in such a way that all the hard work typically done by a data mining expert when creating mining models has already been done when building IPM. Specifically, the models are based on decision trees. Generating prediction models in IPM is an efficient and scalable process. Figure 5 illustrates this process. Notice that being instance-based it has to be tailored to the specific kind of instances that it deals with, in this case, process instances. Different models are generated for different execution stages, so that at prediction time, according to the execution stage of a given instance, the corresponding model can be retrieved and applied to the execution data that exists up to that moment.

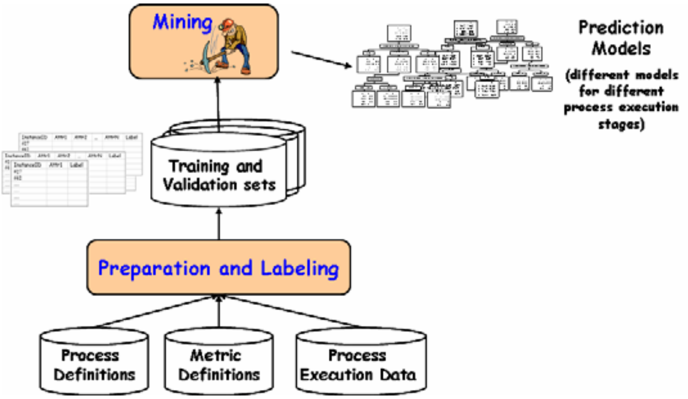


Fig. 5. Instance-based prediction model generation

The technology underlying IPM has been reported somewhere else [3], so we won't discuss it here anymore.

3.3 Time Series Prediction

The goal of G-TSFM is to make time series forecasting accessible to the end user. Most time series forecasting tools either require user experience in visually analyzing time plots of the series to characterize them and decide which techniques, specific models and parameters to use to prepare the series and generate forecasting models, or else blindly apply all the models included in every situation. The first approach requires considerable experience, while the second one is absolutely inefficient and therefore not scalable. Instead, G-TSFM replaces human interaction with advanced techniques that manipulate and analyze data to characterize the time series and apply the most suitable model(s) and parameters automatically. This means, that the components (e.g., seasonality, trend) present in the series are first identified to categorize it, and then a tournament between the suitable models for that type of series is conducted. In addition, the model parameters are optimized ensuring that the best-fitted model is found. Finally the winner model may be evaluated to see if it can be refined to improve its forecasting accuracy by additionally fitting another model. G-TSFM also includes the capability to derive diverse time series from the same data stream to respond to the diverse time granularity forecasting requirements (i.e., hour, day, week, month, quarter, year).

G-TSFM is aimed at time series derived from data streams that do not show significant discontinuities (in those cases only informed guesswork is likely to be better than statistical projections). A few useful definitions follow:

Time series.- a time series is a sequence of observations ordered in time. It can be continuous (when there is an observation at every instant of time, e.g., electrocardiograms), or discrete (when there is an observation at (usually regularly) spaced intervals, e.g., stock prices).

Trend.- is a long term movement in a time series. It is the underlying direction (an upward or downward tendency) and rate of change in a time series.

Seasonality.- it is the component of variation in a time series which is dependent on the time of year. Describes any regular fluctuations with a period of less than one year.

G-TSFM deals with discrete regularly spaced time series that may include trend and/or seasonality. In order to generate a forecasting model it first applies some techniques to characterize the series and according to such characterization it decides which forecasting algorithms are appropriate to apply. It uses a double optimization process in which not only each model is optimized in terms of the settings of its parameters, but it also selects the best of these models. Next we briefly describe the components of the model generation part of the G-TSFM architecture, shown in Figure 6.

- *Aggregator*: aggregates observations (measurements in our case) by different time units to derive time series.
- *Random Tester*: time series are tested for randomness to determine forecasting viability.
- *Outlier Detector & Adjuster*: performs transformations and statistical analysis to identify different types of outliers in the data and makes adjustments so that forecasting accuracy is not jeopardized.
- *Seasonality Finder*: seasonality is a systematic component that can appear in a time series. Since some techniques cannot deal with seasonality and those that can need seasonal parametric information, G-TSFM incorporates a novel algorithm to identify seasonal patterns. Not only it detects whether seasonality exist or not (to discard inappropriate techniques), but it even identifies any other non redundant periodicity, as well as their lengths, which is valuable information for the forecasting algorithms. Other tools require the user to identify such information from a plot of the time series, but the random component of the series may make it impossible to visualize periodicities even for experts.
- *Trend Finder*: trend is another systematic component in time series which is identified in G-TSFM to determine the suitability of forecasting techniques. To find a trend it is necessary to first filter the series with a specific kind of moving average and then fit the filtered series with linear regression and measure the slope.
- *Model Generator*: it is based on an algorithm that first categorizes the series according to its components. Then, starts a two-phase forecasting model generation process. In the first phase a tournament between the appropriate models is conducted to find the one that has the highest accuracy. In turn, each model is tuned by optimizing its internal parameters to minimize the forecasting error. In the second phase the forecasting errors of the winning model are analyzed to determine if certain refinements can be made. If that is the case, a model is fitted to the errors, the new errors are evaluated and if accuracy improved the model is adjusted accordingly. At this moment our engine has three different kinds of forecasters:

- *Seasonal Forecaster:* contains forecasting algorithms for time series with seasonality (with or without trend).
- *Trended Forecaster:* its algorithms are best suited for time series with a trend component (no seasonality).
- *Miscellaneous Forecaster:* it includes various models for stationary series (and no trend nor seasonality).

- *Parameter Optimizer*: finding the best set of parameters for the forecasting models is critical. It is a time consuming task often done manually by experts. In G-TSFE this task is realized with efficient mathematical and genetic algorithms based optimization techniques.

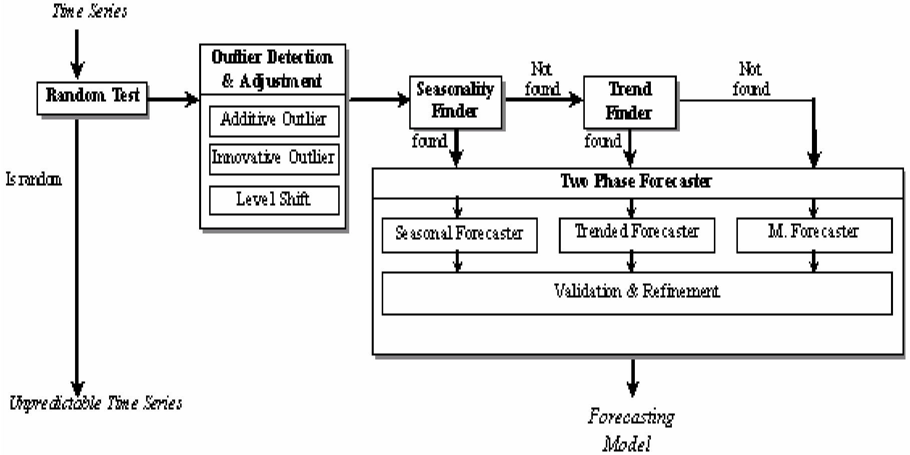


Fig. 6. G-TSFM Architecture

Once a model has been selected as the best one for a given time series, it is stored in a repository (the business operations management repository in our case) so that it can be retrieved at any moment that a forecast is requested.

At forecasting time, the appropriate model is retrieved from the repository and applied to the metric time series so that a numeric forecast is produced. This numeric forecast is mapped to the class corresponding to the range implicitly defined by the thresholds set at metric definition time. For example, if a metric has thresholds defined for “low”, “medium” and ‘high” values, and a forecast has a value that falls in the range corresponding to “high”, then the numeric forecast is mapped to the “high” class. Therefore, TSFM also has a module whose function is to provide forecasts. This module (not shown in Figure 6), has the following main components:

- *Model Retriever*: according to the information in the forecast request (see the API below) retrieves the appropriate forecasting model from the repository.
- *Aggregator*: same as above.
- *Forecaster*: applies the model to the time series derived by the aggregator. Produces a numeric forecast along with a confidence value.

- *Class Mapper*: it maps the numeric forecast into the class corresponding to the range of values within which the numeric value lies. These classes are implicitly defined when metric thresholds are set.
- *Confidence Estimator*: it calculates the confidence on the class forecast from the confidence of the numeric forecast and the forecast distance to the thresholds corresponding to the class' range limits.

G-TSFM can work as a standalone tool or can be integrated into a monitoring platform. In the specific case of Business Cockpit, forecasts requests are for the aggregated value (e.g., average, sum) of a given metric defined on a specific business process at a given time (e.g., next day, next week). For example, the value of the average number of payments that will require manual audit next week. Here, the aggregator is 'average', the metric is 'number of payments with manual audit', the business process is 'payables' and the time is 'next week'. In fact, in this context even though the forecast returns a numeric value, what really matters is whether the forecast exceeds some threshold or not. And this is why we can be somewhat forgiving with respect to accuracy.

The API of G-TSFM for Business Cockpit consists of two methods:

- *generate_model (process_id, metric_name, time_unit, aggregator)*,

where:
process_id is the unique identifier of the business process,
metric_name is the unique name of the metric,
time_unit is the granule of the time series, e.g., hour, day, week, month, year
aggregator is the operator applied to metric values to aggregate them by time unit

- *predict (process_id, metric_name, time_unit, number_of_time_units, aggregator)*

where:

- *number_of_time_units* indicates the time (expressed in time units) for which the forecast needs to be done;
- the other parameters have the same semantics as for the method *generate_model.*

In fact, the same methods but with different parameters exist for the instance-based prediction, therefore *generate_model* and *predict*, are overloaded methods.
For technical details about the algorithms used see [4].

3.4 Experimental Results

To validate our approach we have experimented with many public domain time series used in other approaches to compare with their results. Table 1 is representative of a comparative analysis of the performance obtained with G-TSFM and some semi-automatic approaches of Exponential Smoothing (simple, double or Holtz and triple or Holtz-Winters) (ES) and Box-Jenkins (BJ) which required manual input of some

Table 1. Comparison between the different forecasting approaches

Series	Type	ES	BJ	G-TSFM	Description
Passengers	Seasonal & Trended	0.104	0.181	**0.098**	Monthly airline passengers.
Paper	Seasonal & Trended	0.035	0.076	**0.003**	Monthly sales of paper (France).
Max-Temp	Seasonal	0.137	0.186	**0.133**	Maximum temperature in Melbourne.
Chemical	Trended	0.830	0.861	**0.782**	Chemical concentration readings.
Prices	Trended	1.000	1.006	**0.983**	Daily IBM stock closing prices.
Sunspots	Non linear	0.762	**0.434**	0.745	Annual Wolf Sunspot Numbers.
Kobe	Non linear	0.823	**0.027**	0.557	Seismograph of the Kobe earthquake.

parametric information. G-TSFE didn't require any external information (i.e., totally automatic). Performance was measured in terms of the errors[1] obtained in the forecasting (non-seen) part of the series.

It can be seen that G-TSFM beats the others in most cases, with the exception of the non-linear (i.e., series dominated by random variation) ones for which Box-Jenkins (not implemented yet in G-TSFM) if better suited. Our experience with other series, including some derived from data captured with OpenView, has been equally satisfactory. This means that for series with seasonality or other periodicities, and/or trend, our approach not only enables total automation but also gives comparable performance than others that require some human interaction.

We still need to validate our approach in the business operations domain where numeric forecasts are mapped to classes.

4 Competitive Landscape

There are no tools that offer out of the box process instance based prediction. A complex mining process would have to be put in place to develop applications to make this kind of predictions. This process would be costly, complex and require data mining expertise. In contrast, in our approach we propose to integrate this functionality into the monitoring platform to make it readily available to the end users. This means that all the steps of the mining lifecycle need to be automated and tailored for the business process domain, which is what has been done to develop IPM.

[1] To make the errors comparable, we used the Theil's U statistic where zero represents a perfect forecast, while unity equals a naïve no change forecast.

With respect to time series prediction, there are a number of packages that include forecasting models but they are meant to be used by statisticians and do not automate the forecasting process. In fact, each step of the process needs to be done separately and often using more than one product. The user needs to visually study the series plot to determine which techniques to apply to transform and analyze it. This human interaction is often present throughout the entire forecasting lifecycle, which is the reason why time series forecasting has traditionally been a domain exclusive to statisticians. In response to this, some products have appeared with the goal of automating the forecasting. However, such products have several limitations. Some are applications embedding forecasting functionality tailored to the specific application domain (e.g., Oracle demand planning [6]) and therefore only available within the application. Others still require some user interaction to input parametric data and even though they try to make the forecasting task easier, they are still aimed to experts (e.g., IMSL [5]). The only product that to our knowledge automates the whole forecasting lifecycle is the recently announced SAS High Performance Forecasting [7] which is an integrated member of a larger analytical suite. However, this capability can only be used within the suite and integrating it with other applications such as OpenView management software seems to be difficult.

5 Status and Next Steps

We have presented in this paper the need that companies have for readily available prediction functionality as a way to improve their business operations through better planning and decision making. We have also presented our approach and architecture to build a prediction engine for a business process monitoring environment. The main goal of our work is to automate the prediction processes in order to make metric predictions readily available to business users. We distinguish two kinds of predictions, instance-based and time series based, for which two main modules, IPM and G-TSFM respectively, have been incorporated into our prediction engine, Metric Predictor. IPM, which automates the predictions for active process instances, has been implemented and integrated into our process monitoring platform, Business Cockpit, whereas for G-TSFM although most of the components have been implemented, it still needs some refinements and extensions before being integrated into Business Cockpit (or any other monitoring platform). All developed components have been subject to experimentation and the results are very promising. We plan to integrate Metric Predictor into OV-BPI and extend the techniques to handle data streams efficiently without having to recompute the models from scratch.

References

[1] Hewlett-Packard. OpenView Business Process Insight. Information available from http://www.managementsoftware.hp.com/products/bpi

[2] Malu Castellanos, Fabio Casati, Umesh Dayal, Ming-Chien Shan. iBOM: A Platform for Business Operation Management. *Proc. 21st International Conference on Data Engineering* (ICDE 05). Tokyo, Japan. June 2005.

[3] D. Grigori, F. Casati, U. Dayal, M. Castellanos, M. Sayal, and M.C. Shan. Business Process Intelligence. Computers in Industry Volume 53, Issue 3 (April 2004). Special issue on Process Mining.
[4] Malu Castellanos, Norman Salazar. A forecasting engine for monitoring applications. HP Technical Report. In preparation.
[5] http://www.vni.com/solutions/forecasting/autoArima.html.
[6] http://www.oracle.com/applications/planning/DP.html.
[7] http://www.sas.com/technologies/analytics/forecasting/hpf/

Conversation Quantization for Conversational Knowledge Process

Toyoaki Nishida

Dept. of Intelligence Science and Technology
Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan
nishida@i.kyoto-u.ac.jp
http://www.ii.ist.i.kyoto-u.ac.jp/~nishida/

Abstract. In this paper, I present a computational approach to understanding and augmenting the conversational knowledge process that is a collective activity for knowledge creation, management, and application where conversational communications are used as a primary means of interaction among participating agents. The key idea is conversation quantization, a technique of approximating a continuous flow of conversation by a series of conversation quanta that represent points of the discourse. Conversation quantization enables to implement a rather robust conversational system by basing it on a large amount of conversational quanta collected from the real world. I survey major results concerning capturing, accumulating, presenting, and understanding conversation quanta.

1 Introduction

Knowledge creation, circulation, and utilization is the central concern in the knowledge society. Although the information infrastructure permits us to exchange knowledge beyond spatial and temporal constraints, we are still suffering from what we call the understanding and communication bottlenecks caused by the limitation of human cognitive capability. Even though a huge amount of knowledge is accessible, we can understand and communicate only a little portion of knowledge at a time.

The understanding and communication bottlenecks may become a major cause that hinders the entire knowledge process. The knowledge cannot be effective unless it is understood and action is taken based on it. The existence of knowledge does not necessarily mean that individual people may be aware of knowledge and actively take action for it, even though the information infrastructure is built. In risk communication, for example, various kinds of risks in the society need to be identified, understood, and shared so that the society as a whole can recognize and act to minimize them. People might be indifferent to earthquake disaster prevention, or they might not be aware of the existence of knowledge about earthquake disaster prevention, or they may not understand what is implied by the knowledge even though they can access it. Even active or knowledgeable people may not be able to find a proper means for accomplishing their goal. Information processing capability of individual is so limited that each individual cannot cope with complexity of the modern world.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 15 – 33, 2005.

We consider a community plays a key role in knowledge creation and sharing [1]. A community is defined to be a group of people who are tied together by weak ties such as common interest or common work practice. Community plays a critical role in the network age in the sense that it provides people with a competence of dealing with the complex and dynamic nature of the information world. Community helps people not only collect and interpret information from the world but also communicate their idea to the world.

The primary asset of a community is human relationship and knowledge that are created and harnessed in the community. We would like to develop an information and communication technology for helping people communicate with each other, to create fruitful knowledge.

I push the envelope of conversational communications on the net, for conversation is the most natural means for us to communicate with each other. In fact, conversational communication has many advantages. First, conversation provides an opportunity for the participants to collaboratively build a brilliant story by putting together small pieces of ideas and experiences. Second, conversation allows participants to mutually adjust and coordinate the information flow to meet their requirement using various kind of nonverbal communication behaviors, such as gestures or eye gaze. Third, conversation has a powerful mechanism of having participants involved in the subject so that they engage as active participants. People have excellent skills in communicating in a conversational fashion but little portion of those skills can be applied in online communication.

Conversational Informatics is a field of research aiming at investigating human conversational behaviors as well as designing conversational artifacts that can interact with people in a conversational fashion. Based on the foundation of Artificial Intelligence, Pattern Recognition, and Cognitive Science, we attempt to establish a new technology consisting of environmental media, embodied conversational agents, and management of conversational contents.

The application area of Conversational Informatics involve knowledge management and e-learning. Although we seek to develop a new information and communication technology that can contribute to the increase of both individual and social intelligence, we need to look carefully at the current level of our technology. Unfortunately, it is still too hard to build a fluent conversation system with the current technology. At the same time, however, we can build a useful system that can augment our conversational behaviors in a significant way, if we limit our goal to the less ambitious level.

In this paper, I present a computational approach to understanding and augmenting the conversational knowledge process that is a collective activity for knowledge creation, management, and application where conversational communications are used as a primary means of interaction among participating agents.

The key idea is conversation quantization, a technique of approximating a continuous flow of conversation by a series of conversation quanta that represent points of the discourse. Conversation quantization enables to implement a rather robust conversation system by basing it on a large amount of conversation quanta collected from the real world.

This paper describes the conceptual framework of conversation quanta, how to extract them, and the system that supports conversational knowledge process based on conversation quanta.

2 Conversation Quantization

The basic idea underlying the conversation quantization method is to approximate the continuous stream of conversation as a discrete sequence of coherent conversation fragments called conversation quanta [2]. We define a conversation quantum to be an entity that contains a minimal amount of contextual information. In other words, it means that each conversation quantum makes a minimal sense even though it may be presented in an inappropriate context. This aspect of conversation quanta allows us to implement a data-driven, robust and rich conversational system that will not totally fail even in the case where the speech recognition technology does not work properly. Given a conversational situation, a conversation quantum that best matches it will be sought from the collection of conversation quanta, and a role of the participants of the retrieved conversation quantum can be replayed by an embodied conversational agent, and other roles will be mapped to the participants in the given conversational situation. Such an algorithm is relatively easy to implement and rather robust in nature.

The framework of the conversation quantization consists of a spiral of extracting conversation quanta from conversation, accumulating them into a database, and applying them to other conversational situations on demand (Fig. 1). Embodied conversational agents are introduced to replay the conversational behaviors on behalf of an actor recorded in a conversation quantum.

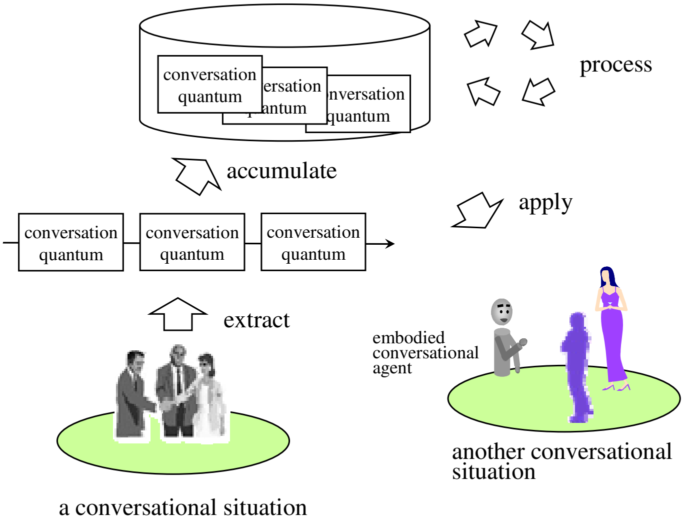


Fig. 1. The framework of conversation quantization

In addition to basic operations on conversation quanta, complex operations will produce a more content-oriented services such as summarization, complex story building, or spatial arrangement of conversation quanta. There are generally two methods of presenting the content of a conversation quantum. One is temporal expansion that generates a temporal sequence of information presentation actions. By dynamically switching among multiple conversation quanta upon the other participants' utterances, we can give the system an interactive flavor. An alternative method is spatial expansion that generates information landscape which is a spatial representation of the content of a conversation quantum. An information landscape enables the user to visually grasp the global nature of knowledge, explore the information space, and accommodate new information at an appropriate place. Conversion between other information media such as documents makes the conversation quantization method more useful. Furthermore, quantitative measurement of conversational processes is needed to quantitatively understand the effect of the system.

The implementation of the conversation quantization theory depends on the data structure for representing conversation quanta. One may use plain video clips as representation but its utility in retrieving and processing would be quite limited and a large cost would be required in retrieving, editing, and applying conversation quanta. Alternatively, one could use a deep semantic representation such as logical formulas or case frames. Such semantic representation should be properly augmented due to the limitation of the expressive power.

The knowledge card circulation model is an elementary implementation of the conversation quantization theory. It is based on a simple data structure called knowledge card that consists of text-based description of transcript and a referenced image or video. One or more knowledge cards may be connected into a story that represents a complete story with a plot. There are a suit of software package for creating, accumulating, presenting, editing, exchanging, converting, and evaluating knowledge cards, as follows:

(1) Conversation quanta capture

A conversation quanta capture is a system that will (semi-)automatically produce a knowledge card representation for a given conversation scene.

(2) Conversation quanta editor

A conversation quanta editor permits the user to browse and edit knowledge card representation of conversation quanta.

(3) Embodied conversational agents

Embodied conversational agent (ECA) is a program for playing stories consisting of knowledge cards. An ECA employs paralinguistic and nonlinguistic conversational means to control information flow in a conversation. In normal settings, each ECA stands for an existing real person. Accordingly, the conversation partners can recognize whom the ECA talking in front of them stands for. This provides proper background information for interpreting messages from the ECA. Meanwhile, we might well employ anonymous ECAs in certain situations where we want to facilitate communication.

(4) Knowledge channel
Knowledge channel is a mechanism for strategically controlling the flow of knowledge cards. This will contribute to designing long-term interaction by controlling the flow of knowledge cards. Knowledge channel is a conduit connecting a server and a client where knowledge card flow is controlled strategically. Both the knowledge card server and the knowledge card client can control the knowledge card flow using the knowledge channel control policy.

(5) Memory manifold
Memory manifold serves as a container that allows for spatially accumulating memory items for producing landscapes.

(6) Knowledge card Servers
A knowledge card server stores knowledge cards and provides them for knowledge card clients. Knowledge card servers can be classified into those for personal and community use.

(7) Media converters
A media converter translates knowledge card representation from/to other information media such as ordinary documents.

(8) Conversation evaluators
Conversation evaluators may be used to evaluate aspects of conversation, such as proficiency, by integrating various features measured from conversation.

The software tools mentioned above are partly implemented and tested. In the subsequent sections, I will overview several embodied conversational agents we have implemented and tested based on the knowledge card circulation model. Then, I will show a memory manifold system that provides the user with a place for accumulating a large collection of knowledge cards. I will also describe a preliminary work on conversation quanta capture.

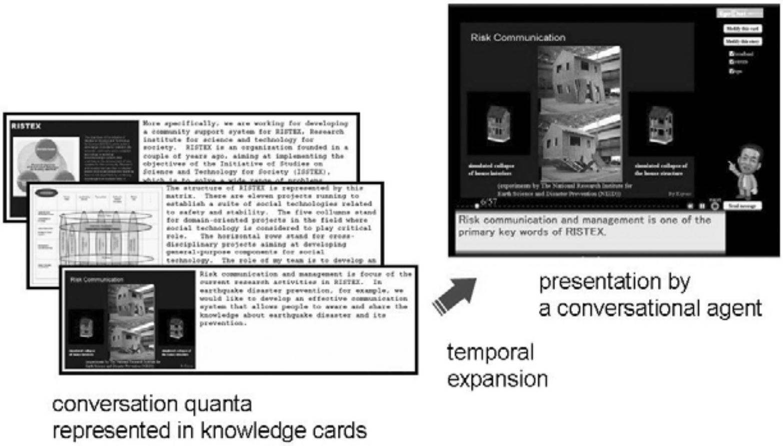
3 Embodied Conversational Agents Based on the Knowledge Card Circulation Model

In Artificial Intelligence and Human Interface, the researchers have already invented embodied conversational agents which are computer programs that can talk with people by integrating verbal and nonverbal communication means. Unfortunately, conventional embodied conversational agents are not enough as a mediator among people, for they are not designed to help a person communicate a large amount of information to another person. Even though the user wanted have the embodied conversational agents speak a lot, it is very expensive to transfer a large amount of information from the user to an agent so that the agent can speak of it. Embodied conversational agents based on the knowledge card circulation model allows for drastically decreasing the cost for information circulation by embodied conversational agents.

3.1 EgoChat

EgoChat is a system for enabling an asynchronous conversational communication among community members [3,4]. EgoChat implements both temporal and spatial expansion of conversation quanta.

In EgoChat, a simple implementation of conversation quantization using data structures called knowledge cards, stories and knowledge channel was employed. A knowledge card is relatively self-contained package of tacit and explicit knowledge, enabling one to implement a rather robust conversational system without introducing a complex discourse generation algorithm (Fig. 2). A story is a sequence of knowledge cards, representing a complete story based on a plot. In a story, preceding and succeeding knowledge cards give a discourse to each knowledge card.



More specifically, we are working for developing a community support system for RISTEX, Research institute for science and technology for society. RISTEX is an organization founded in a couple of years ago, aiming at implementing the objectives of the Initiative of Studies on Science and Technology for Society (ISSTEX),

The structure of RISTEX is represented by this matrix. There are eleven projects running to establish a suite of social technologies related to safety and stability. The five collumns stand for domain-oriented projects in the field where social technology is considered to play critical role. The horizontal rows stand for cross-disciplinary projects aiming at developing general-purpose components for social technology. The role of my team is to develop an

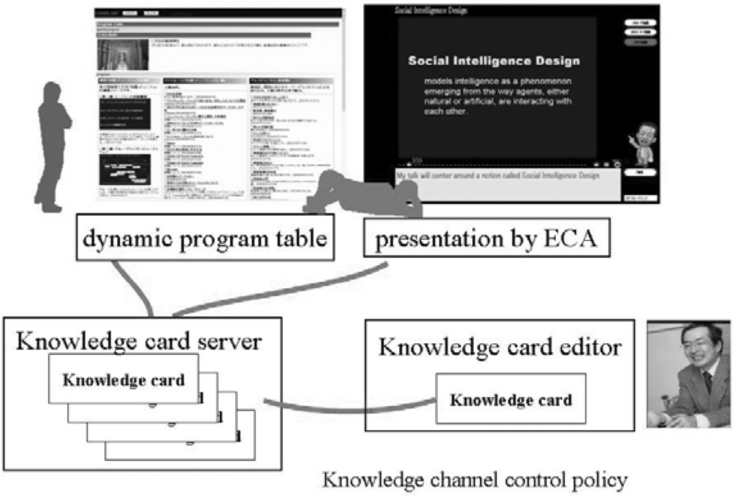
Risk communication and management is focus of the current research activities in RISTEX. In earthquake disaster prevention, for example, we would like to develop an effective communication system that allows people to aware and share the knowledge about earthquake disaster and its prevention.

Risk communication and management is one of the primary key words of RISTEX.

Fig. 2. Knowledge cards as an implementation of conversation quanta

EgoChat is based on the talking-virtualized-egos metaphor. A virtualized ego is a conversational agent that talks on behalf of the user. Each virtualized ego stores and maintains the user's personal memory as a collection of knowledge cards and presents them on behalf of the user. It helps the user not only develop her/his personal memory but also better understand other members' interest, belief, opinion, knowledge, and way of thinking, which is valuable for building mutual understanding. We use a powerful dialogue engine that permits a virtualized ego to answer questions by searching for the best match from a potentially large list of question-answering pairs prepared in advance. The architecture of the EgoChat system is shown in Fig. 3.

EgoChat provides a couple of unique features. First, it integrates personal and interpersonal knowledge life cycles. At earlier stages of the lifecycle when knowledge is not well captured, the user might want to describe her/his idea as one or more knowledge card and have the virtualized ego present them for personal review. After a while when the knowledge becomes clear and evident to the user, s/he might want to delegate her/his virtualized ego to her/his colleagues to present the idea and ask for



Social Intelligence Design

models intelligence as a phenomenon emerging from the way agents, either natural or artificial, are interacting with each other.

My talk will center around a notion called Social Intelligence Design

Fig. 3. The architecture of the EgoChat system [3,4]

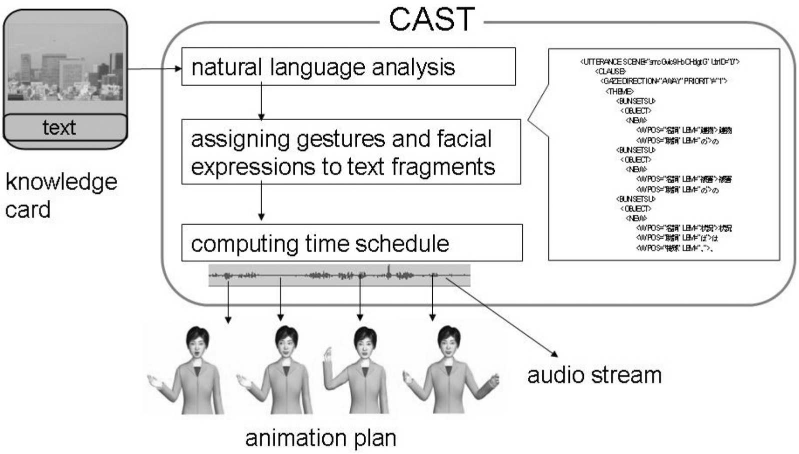
critiques. The automatic question answering facility of EgoChat will encourage the audience to ask questions or give comments. The virtualized ego can reproduce the question-answering session with the audience so the owner can review the interactions. It will highly motivate the owner to supply more knowledge or improve existing knowledge for better competence.

Second, EgoChat allows for specifying a channel policy that is used to define the control strategies of the sender and the receiver. Four types of strategies are identified depending on whether the strategy is about the order of programs in a single program stream or about the way multiple program streams are mixed, and whether the program scheduling is static or dynamic. The skeleton of the actual flow structure of knowledge cards for a given pair of the sender and receiver is determined by resolving constraints of their channel policies. It can be visually presented to the user by a dynamic program table.

3.2 SPOC

SPOC (Stream-oriented Public Opinion Channel) is a web-based multimedia environment that enables novice users to embody a story as multimedia content and distribute it on the Internet [5,6]. A sophisticated presentation generation from the plain-text representation of conversation quanta specifying utterances of participants in the conversation is addressed. The system produces both digital camera work and agent animations according to linguistic information in a given natural language text (Fig. 4).

We have collected and analyzed presentations by seven people and identified nine features as factors of predicting gesture occurrence. The analysis is reflected in the set of rules for determining the gestures of an embodied conversational agent. For example, one rule specifies that if an enhancement is encountered, a “beat gesture” (a simple flick of the hand or fingers up and down) will be generated.


CAST
text
knowledge card
natural language analysis
assigning gestures and facial expressions to text fragments
computing time schedule
audio stream
animation plan

```
<UTTERANCE SCENE="smcGvlc9HbCHdgtG" UttID="0">
  <CLAUSE>
    <GAZEDIRECTION="AWAY" PRIORITY="1">
     <THEME>
       <BUNSETSU>
        <OBJECT>
         <NEW>
            <W POS="名詞" LEM="建物">建物
            <W POS="助詞" LEM="の">の
       <BUNSETSU>
        <OBJECT>
         <NEW>
            <W POS="名詞" LEM="被害">被害
            <W POS="助詞" LEM="の">の
       <BUNSETSU>
        <OBJECT>
         <NEW>
            <W POS="名詞" LEM="状況">状況
            <W POS="助詞" LEM="は">は
            <W POS="特殊" LEM="、">、
```

Fig. 4. Generating Animation by CAST [5,6]

The animation generator called CAST implements the mechanism for determining appropriate agent behaviors according to the linguistic information contained in Japanese text as. CAST consists of the Agent Behavior Selection Module (ABS), the Language Tagging Module (LTM), a Text-to-Speech engine (TTS), and a Flash-based character animation system RISA (RIStex animated Agent system). When CAST receives a text input, it will forward it to the ABS. The ABS selects appropriate gestures and facial expressions according to linguistic information calculated by the LTM. The ABS calculates a time schedule for the set of agent actions based on the timing information obtained by accessing the TTS. The output from the ABS is a set of animation instructions that can be interpreted and executed by the RISA animation system.

As an optional function in CAST, we have also built a component called the Enhancement Generator that automatically adds highlighting animations to agent gesture animations after the gestures are determined by the ABS. Two types of highlighting methods are incorporated to emphasize synchronization between verbal (speech) and nonverbal (gesture) behaviors.

One is superimposition with beat gesture. Beat gestures simply emphasize one part of an utterance without representing the meaning of a word. To visualize synchronization between the emphasized words and a beat gesture, the Enhancement Generator adds a superimposition of the emphasized words to the agent's beat gesture animation.

The other is illustrative animation with metaphoric gesture. When a specific shape of gesture is assigned to a metaphoric gesture, it will be emphasized by illustrative animations, such as an arrow and a line. If the emphasized concept implies motion or movement, such as "increase" or "decrease," the direction of the movement will be illustrated by an arrow animation. If the emphasized concept expresses a static state, a motionless picture will be used to emphasize the gesture. For example, when the agent is performing a "long" gesture, a rectangle shape is shown near the agent's hands to emphasize the length.

3.3 IPOC

IPOC (Immersive Public Opinion Channel) [7,8] is a successor of SPOC. IPOC allows for expanding conversation quanta in a virtual immersive environment. Users can interact with conversational agents in a story-space, which is a panoramic picture background and stories are embedded in the background (Fig. 5). The embedded stories are presented on demand by the user or spontaneously according to the discourse. Stories are used to represent a discourse structure consisting of more than one conversation quantum.

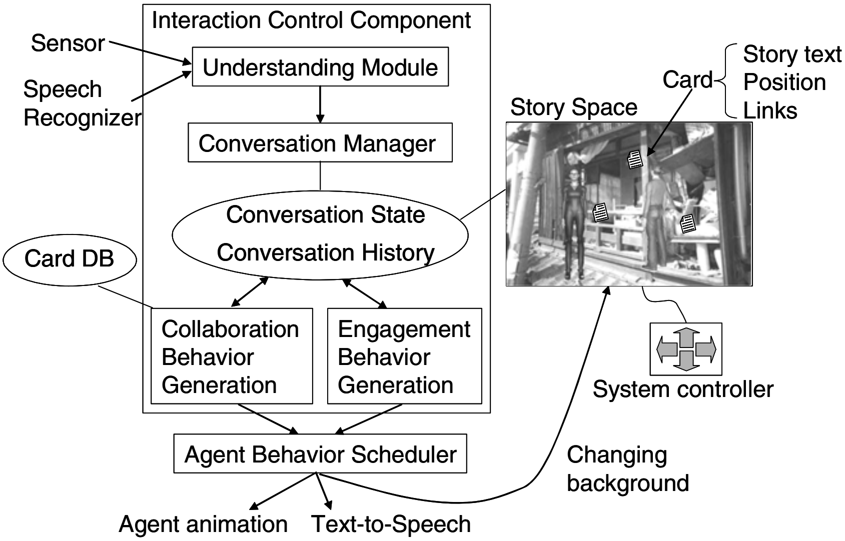


Fig. 5. The architecture of IPOC [7, 8]

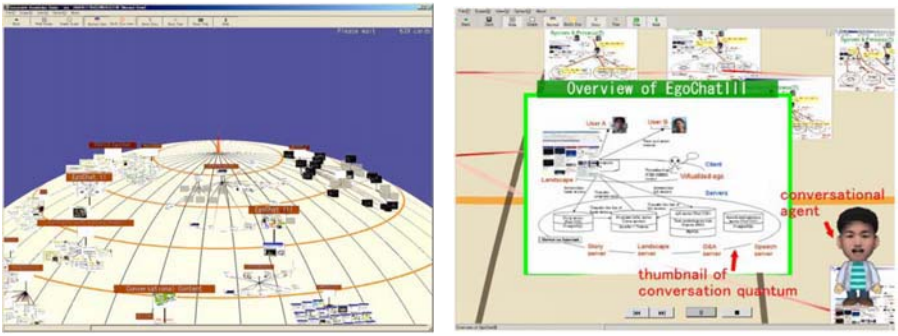
In order to generate a more complex set of nonverbal behaviors in an immersive environment, theories of nonverbal communication are extensively studied and incorporated in the agent behavior generation system. Sidner proposed conversation management, collaboration behavior, and engagement behaviors as communicative capabilities required for collaborative robots [9]. Conversation management includes abilities of turn taking, interpreting the intentions of participants in the conversation, and updating the state of the conversation. Collaboration behavior determines agent's next action in order to accomplish the goal for the conversation and the collaboration with the user. Engagement behaviors consist of initiating a collaborative interaction, maintaining the interaction, and disengaging from the interaction.

The Interaction Control Component (ICC) of the IPOC system interprets inputs from a speech recognizer and a sensor system, and generates verbal and nonverbal behaviors performed by conversational agents. The major components of ICC are the Conversation Manager (CM) that maintains the history and the current state of the conversation, the Collaboration Behavior Generation Module (CBG) that selects the next Card to be read and determines agents' behaviors in telling a story, and Engagement behavior generation (EBG) that determines appropriate engagement behaviors according to the state of the conversation.

4 Spatial Accumulation of Conversation Quanta

A persistent memory system is an approach to spatio-temporal expansion of conversation quanta. By establishing a long-term relationship with a persistent memory that can coevolves with the user's biological memory, the user, as we believe, will be able to find easily an appropriate place to accommodate new information and make it ready for later use.

We have developed a system called the Sustainable Knowledge Globe (SKG) [10] that permits the user to build her/his own customizable intellectual world by spatially arranging information (Fig. 6a). SKG also enables the user to retrieve knowledge through the temporal representation by using an embodied conversational agent (Fig. 6b).



(a) A bird's-eye view

(b) presentation in an immersive view

Fig. 6. A snapshot of an SKG window [10,11]

To persistently support the conversational knowledge process of persons, both the spatial representation and the temporal representation of conversation quanta are indispensable. SKG allows the user to switch between the spatial and temporal representations (Fig. 7). The global style is a spatial representation that facilitates global and geometric understanding of the huge content that will be in our conversational activities. In contrast, presentation by an embodied conversational agent in the immersive style is a temporal representation of knowledge in the sense that the major axis behind the representation is a temporal evolution of a story. The temporal representation allows for in-depth causal understanding of the issue. It stirs around old and new content and enriches content in conversational fashion.

SKG was implemented on standard PC hardware with Microsoft .NET Framework 1.1 and Managed DirectX9.0c. This implementation provides a view like a terrestrial globe that includes a sand-colored sphere with latitude and longitude lines, the landmarks for the north and south poles, and the equator. The user can create cards on the globe, import a file from the desktop and exchange her/his cards with other users via the network server system. The user can also move around the globe in search for interesting content and construct virtual landscape.

In the global style, SKG provides scaling and zooming operations to visualize huge content. The scaling operation changes nonessential content small to make more space on the landscape. The zooming operation provides macro and micro views for the user to overlook any scale content.

In the immersive style, the users can retrieve the essence of conversation quanta by talking with an embodied conversational agent. The users can interact with the conversational agent that reads aloud annotations of conversation quanta connected along with a story line. The Q&A function is now partly developed by adopting the result of our EgoChat system [3,4]. The interaction plays essential role in the development of knowledge. The conversation quanta can be enriched by users' annotation and rearrangement in the interaction with the agents. For example, a user can review and improve her/his presentation content by listening the agent's presentation objectively. SKG allows only an image based conversation quantum at present, but it is applicable to a movie based quantum.

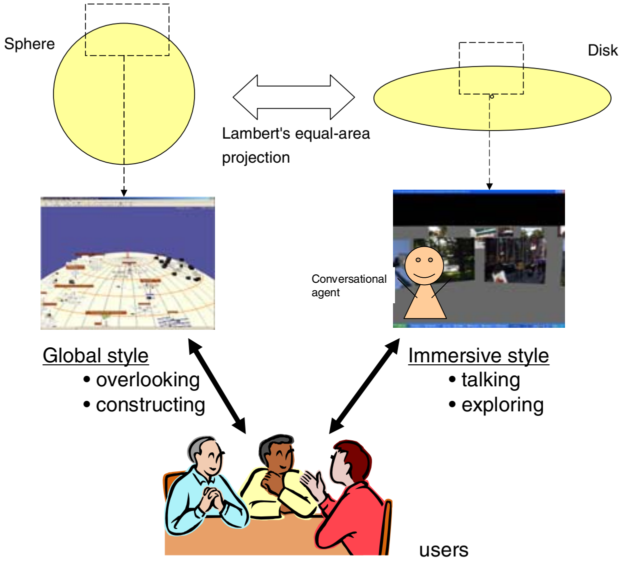


Fig. 7. Switching between spatio-temporal representation of a conversation quantum [11]

We have evaluated the effectiveness of SKG through an experiment in a practical conversational situation. In the experiment, we have recorded the proceedings of a meeting of the ESL project[1] by using SKG. The landscape of SKG was projected on a large screen during the meetings. One operator wrote the speeches down, and recorded on the surface of the globe. The operator also manipulated the globe according to participants' demands that they wanted to focus the specific content. The meetings are held 10 times from August to November in 2004. The total time length of the meetings is 20 hours. The average number of participants is six.

As a result of the experiment, we have acquired 151 content that include speech texts, handouts and drawings. We had an interview with the participants about the effectiveness of SKG and got positive comments as fellows:

- I can overview the whole information in the meetings.
- The landscape is useful for me to look again the past proceedings.

We also got a comment that it is difficult for the speaker to operate SKG by a mouse device. The user should not be engaged with mouse operations in conversational situations because it disturbs communication using natural gestures. We are now developing a novel immersive browser (Fig. 8) where the sphere is projected to the disk by using Lambert's equal-area projection. Disk-shaped world seems to be more immersive than sphere-shaped world. We plan to improve operability of SKG by using physical interfaces like a motion capturing system for immersive browser.

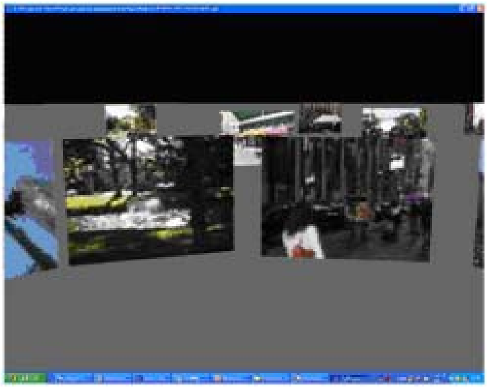


Fig. 8. Immersive browser of SKG [11]

5 Capturing Conversation Quanta

In order to have the knowledge card circulation model effectively applied in practical applications, we would like to reduce the overhead of content production as much as possible, even though the knowledge card circulation model as a framework contributes a lot to information and knowledge circulation. In this section, I will explore the possibility of automating the process of capturing conversation quanta from real-world conversational situation.

[1] Eco Smart Life Research project. See http://esl.sfc.keio.ac.jp/esl_e.html for more details.

5.1 Preliminary Experiment on Identifying Conversation Quanta

In order to understand the process of capturing conversation quanta from real conversations to create a new story, we carried out a preliminary experiment consisting of the following steps [11]:

(1) Setting a practical conversational situation
(2) Capturing conversation by video camera
(3) Extracting conversation quanta from the video stream by hand
(4) Repeating (1) - (3) in different situations
(5) Creating a new video content by combining conversation quanta gathered from past conversations.

A couple of members of the author's group had five meetings. One participant is a master course student (subject A) and the other is a postdoctoral fellow (subject B). Every meeting was held in different places and different weeks. Each of them talked using PowerPoint slides or PDF documents on mobile PC (with a web camera and a microphone) to capture his voices, faces and context. As a result of these meetings, we obtained six hours video of subject A and the same length video of subject B. The topics of the conversation was conversation quantization -- its history, problems, approaches, systems and so on. Almost half of the conversation was presentation and the rest was discussion.

We focused on conversations based on slide presentation (about three and half hours). We assumed that the conversation sequence consists of a time interval consisting of presentation followed by question-answering (Fig. 9). We investigated the conversation, and obtained 41 quanta for subject A and 66 quanta for subject B.

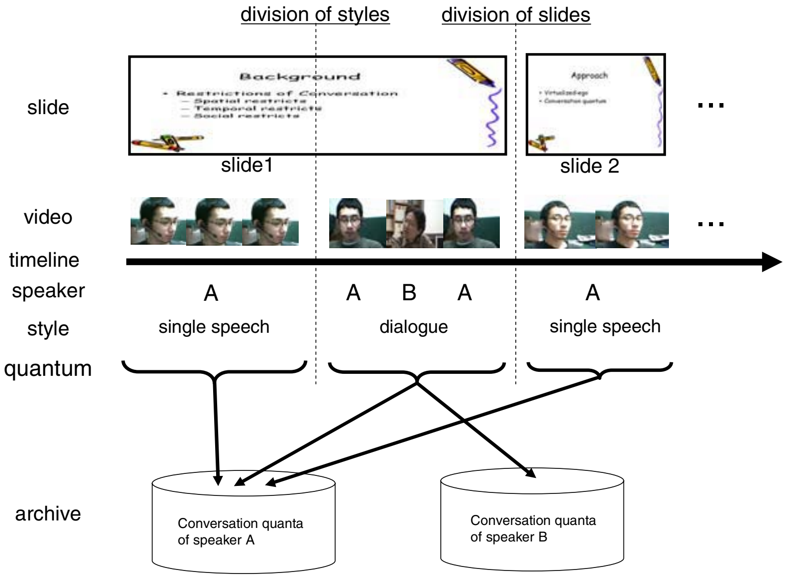


Fig. 9. The first approximate model for extracting conversation quanta [11]

The average time length of each single speech quantum is 42 seconds, however divergence is relatively large. Similar results were obtained for dialogue quanta.

We have made an wizard-of-oz experiment of simulating a conversational system using the obtained conversation quanta. We arranged conversation quanta of subject A on the assumption that the system talks with a user on behalf of subject A. Fig. 10 shows the overview of our simulation. Firstly, a user comes in front of a system screen where the face of subject A is displayed. Here, the system begins to talk on behalf of subject A when the user asks for his interest ("Greeting"). The system talks by arranging past conversation quanta that are related to the interest of the user ("Quantum 1" and "Quantum2"). While the system is talking, the user can ask any questions ("Question"). Then the system can answer the question by searching an answering conversation quantum ("Quantum 3"), and keep on talking (Quantum 4).
We have obtained some useful insights about conversation quanta from the simulation above. In Fig.10, Quantum1, Quantum 2, Quantum 4 were acquired in different rooms. Thus, we can make new conversational content from the past conversation

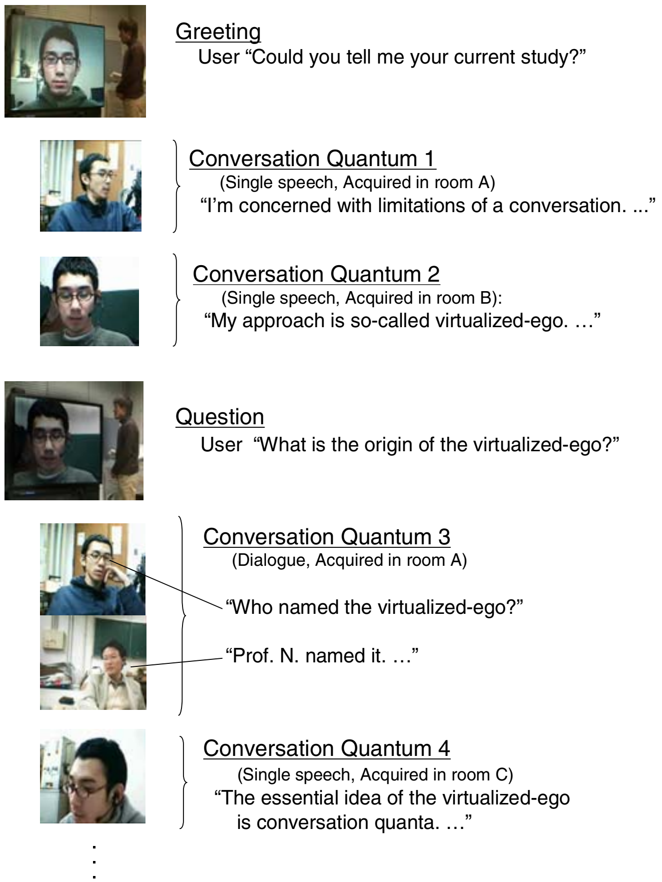

Greeting
User "Could you tell me your current study?"
Conversation Quantum 1
(Single speech, Acquired in room A)
"I'm concerned with limitations of a conversation. ..."
Conversation Quantum 2
(Single speech, Acquired in room B):
"My approach is so-called virtualized-ego. …"
Question
User "What is the origin of the virtualized-ego?"
Conversation Quantum 3
(Dialogue, Acquired in room A)
"Who named the virtualized-ego?"
"Prof. N. named it. …"
Conversation Quantum 4
(Single speech, Acquired in room C)
"The essential idea of the virtualized-ego
is conversation quanta. …"

Fig. 10. A simulation of a conversational video content using conversation quanta [11]

quanta that were got in different situations. A dialogue style quantum is interesting because it contains Q&A pair, jokes and any other pretty conversational rhythms. When we searched conversation quanta that are suitable for a user, thinking about the background knowledge of the user was very important. The conversation in Fig. 10 is understandable by the user because he is a colleague of the speakers on the screen. However, the information about the places and the speakers in a conversation quantum should be explained additionally if the user doesn't know such context.

Although the detailed investigation of the nature of conversation quantization is left for future, we conjecture, based on experiments made so far, that each conversation quantum roughly corresponds to a small talk often identified in the discourse of daily conversations.

5.2 Towards Automated Capturing of Conversation Quanta

Conversation quanta can be extracted from the real world conversation by expanding the ubiquitous sensor room that is proposed by Sumi et al [12]. They implemented a smart room environment where conversational activities can be captured by environment sensors (such as video cameras, trackers and microphones ubiquitously set up around the room) and wearable sensors (such as video cameras, trackers, microphones, and physiological sensors). In order to supplement the limited capability of sensors, LED tags (ID tags with an infrared LED) and IR tracker (infrared signal tracking devices) are used to annotate the audio/video data with the positional information. Significant intervals or moments of activities are defined as interaction primitives called "events". Currently, five event types are recognized, including "stay," "coexist," "gaze," "attention," and "facing" (Fig. 11). Events are captured by the behavior of IR trackers and LED tags in the room. For example, a temporal interval will be identified as a (joint) attention event when an LED tag attached to an object is simultaneously captured by IR trackers worn by two users, and the object in focus will be marked as a socially important object during the interval.

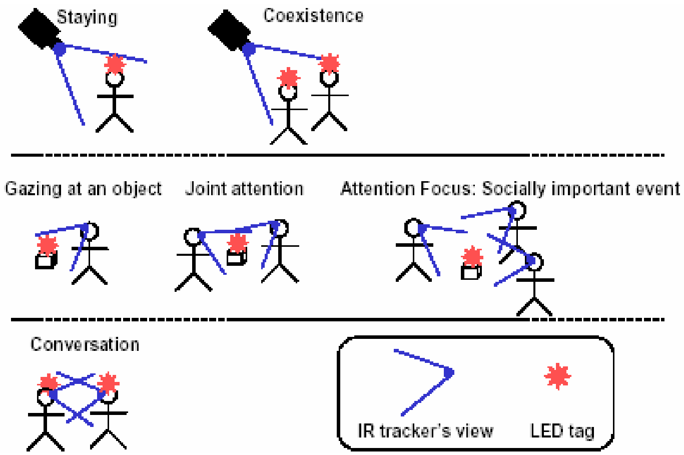


Fig. 11. Interaction Primitives [12]

The ubiquitous sensor room can be extended to capture conversation quanta. The idea is that a series of touch actions on a poster panel would be a good clue to estimate conversational segments and their topics. We might use a touch panel presentation system (Fig. 12). At a poster exhibition site, we supposed that the interaction patterns between a presenter and a visitor are classified into two categories. "Lecture" category means single speech of the presenter and "Interaction" category means dialogue between the presenter and the visitor. In the ubiquitous sensor room, "Lecture" and "Interaction" are roughly segmented into single speech segments and dialogue segments by capturing visitor's "stay" events and dividing voice & video stream into speech segments. However, the single speech segments involve a lot of disordered segments that cannot be reuse in other situations.

Our touch panel presentation system can infer reusable segments by capturing a series of touch actions of the presenter and matching it with typical series of touch actions. The presenter divides the touch panel area into semantic sections of the poster and plays her/his typical presentation in advance. A series of touch actions on the sections are captured while the presenter is explaining the poster to the visitor. The segment is inferred as a conversation quantum ("Lecture Unit") when its series of touch actions are matched well with that of the typical presentation.

Moreover, the system can acquire the conversation topics from the text on the touched poster area. Such information is helpful for retrieving conversation quanta. We have conducted an experiment of our method by using hand-labeled data acquired in ATR[2] Open House 2003. The purpose is to evaluate accuracy of the extraction of "Lecture Unit". In the experiment, a presenter talked with visitors using the poster in the ubiquitous sensor room. The poster was partitioned into 4 rectangle areas by semantic section. The total time length of the presentation was four and a half hours.

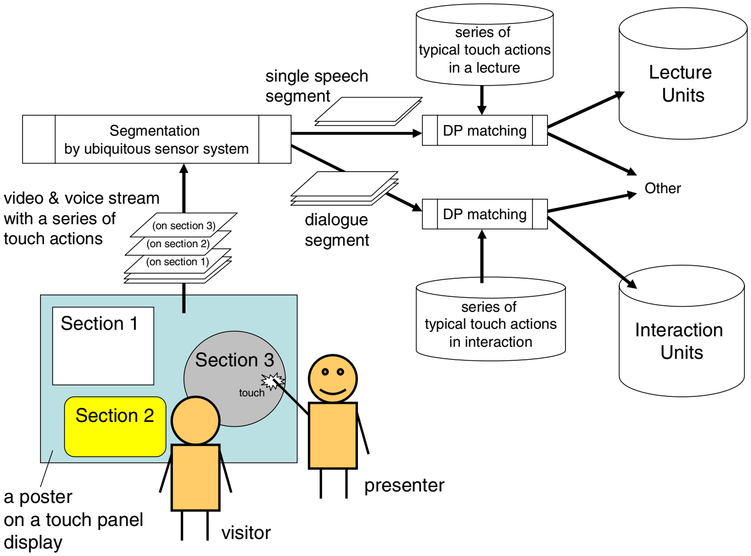

series of typical touch actions in a lecture
Lecture Units
single speech segment
Segmentation by ubiquitous sensor system
DP matching
Other
video & voice stream with a series of touch actions
(on section 3)
(on section 2)
(on section 1)
dialogue segment
DP matching
series of typical touch actions in interaction
Interaction Units
Section 1
Section 3
touch
Section 2
presenter
a poster on a touch panel display
visitor

Fig. 12. Conversation quantizing system in a real world poster session [11]

[2] http://www.atr.co.jp/index_e.html

After that, we obtained 61 series of touch actions of the presenter by analyzing the ubiquitous video. The accuracy by using DP matching method between acquired series and typical series is 81% (39/48) in precision and 64% (39/61) in recall. This result appears promising as the first attempt.

6 Future Work

There are many interesting work left for the future research. Among others, we need to build a more detailed and elegant theory of conversation quantization. We have already obtained an empirical characterization of conversation quanta, but we do not have a systematic and theoretical account of conversation quanta. A more sophisticated theory of conversation quanta will permit us to better design the representation and basic operation for conversation quanta. It may well enable us to predict the cost and effect of building a conversation system based on conversation quantization.

Another big concern is automatic capture of conversation quanta. In addition to the work described so far in this paper, there are a number of interesting works. Minoh and Kakusho developed a robust method for automatically recognizing communicative events in a real class room by integrating audio and visual information processing [13]. Rutkowski studied a method for monitoring and estimating efficiency of the human-human communication based on recorded audio and video by computing correlation between sender activities and receiver responses [14]. Taniguchi and Arita used computer vision techniques to realize the notion of real time human proxy to virtually create a classroom for distributed learning environments [15]. Kurohashi and Shibata integrated robust natural language processing techniques and computer vision to automatically annotate videos with closed caption [16]. Nakamura developed an intelligent video production system that can compute the best matching between the given index-scenario and video stream captured by camera to produce an annotated video [17]. Those works should be incorporated into the theory of conversation quantization.

We would like to integrate SKG with IPOC in future. So far we have developed the global style of SKG, but the immersive style is insufficient. The immersive style is expected to entrain participants to the subject by embodied fashion and improve operations in landscape by using physical interface. IPOC allows for expanding conversation quanta in a photo realistic immersive environment. Users can interact with conversational agents in a story-space, which is a panoramic picture background and stories are embedded in the background.

7 Conclusion

In this paper, we have presented an approach to conversational knowledge process based on conversation quantization, which is a technique of approximating a continuous flow of conversation by a series of conversation quanta that represent points of the discourse. Conversation quantization enables to implement a rather robust conversation system by basing in on the large amount of conversational quanta collected from the real world. I have overviewed several embodied conversational agents we have implemented and tested based on the knowledge card circulation model. Then, I have shown a memory manifold system that provides the user with a place for accumulating a large collection of knowledge cards. Finally, I have described a preliminary work on conversation quanta capture.

References

1. Toyoaki Nishida, Social Intelligence Design for Web Intelligence, Special Issue on Web Intelligence, IEEE Computer, Vol. 35, No. 11 (2002) 37-41
2. Toyoaki Nishida, Conversational Knowledge Process for Social Intelligence Design, Keynote Speech, The 2004 IFIP International Conference on Intelligence in Communication Systems (INTELLCOMM 04), in: A. Aagesen et al. (Eds.): INTELLCOMM 2004, LNCS 3283 (2004) 28–42
3. Hidekazu Kubota, Yasuyuki Sumi, Toyoaki Nishida, Sustainable Knowledge Globe; A System for Supporting Content-oriented Conversation, in Proceedings AISB 2005 Symposium Conversational Informatics for Supporting Social Intelligence & Interaction, 2005 (to appear).
4. Hidekazu Kubota, Jaewon Hur, Toyoaki Nishida, Agent-based Content Management System, in Proceedings of the 3rd Workshop on Social Intelligence Design (SID 2004), CTIT Workshop Proceedings, pp. 77-84, 2004.
5. Nakano, Y. I., Murayama, T., and Nishida, T.: Multimodal Story-based Communication: Integrating a Movie and a Conversational Agent. Vol.E87-D No.6 pp. 1338-1346 2004/6.
6. Q. Li, Y. Nakano, M. Okamoto, and T. Nishida: Highlighting Multimodal Synchronization for Embodied Conversational Agent, the 2nd International Conference on Information Technology for Application (ICITA 2004), 2004.
7. Yukiko I. Nakano, Masashi Okamoto, and Toyoaki Nishida: Enriching agent animation with Gestures and Highlighting Effects, presented at International Workshop on Intelligent Media Technology for Communicative Intelligence in Warsaw, Poland September 13-14, 2004
8. Yukiko I. Nakano, Toshiyasu Murayama, and Toyoaki Nishida: Engagement in Situated Communication By Conversational Agents, presented at 1st International Workshop on "Intelligent Media Technology for Communicative Intelligence", affiliated with 4th National Conference on Multimedia and Network Information Systems, Szklarska Poreba, Poland, September 16-17, 2004.
9. Sidner, C. L., C. Lee, and N. Lesh: Engagement Rules for Human-Robot Collaborative Interactions, in Proc. IEEE International Conference on Systems, Man & Cybernetics (CSMC), Vol. 4, pp. 3957-3962, 2003
10. Hidekazu Kubota, Jaewon Hur, and Toyoaki Nishida: Agent-based Content Management System, in Proceedings of the 3rd Workshop on Social Intelligence Deisgn (SID 2004), CTIT Workshop Proceedings, pp. 77-84, 2004.
11. Hidekazu Kubota, Masashi Takahashi, Ken Satoh, Yohei Kawaguchi, Satoshi Nomura, Yasuyuki Sumi, and Toyoaki Nishida, Conversation Quantization for Informal Information Circulation in a Community, The Fourth International Workshop on Social Intelligence Design (SID 2005), (to be presented)
12. Y. Sumi, K. Mase, C. Mueller, S. Iwasawa, S. Ito, M. Takahashi, K. Kumagai, Y. Otaka, Collage of Video and Sound for Raising the Awareness of Situated Conversations, in Proceedings of International Workshop on Intelligent Media Technology for Communicative Intelligence (IMTCI 2004) (2004) 167-172.

13. Michihiko Minoh, Satoshi Nishiguchi, Environmental Media - In the Case of Lecture Archiving System, in Proceedings of Int. Conf. Knowledge-Based Intelligent Information & Engineering Systems (KES2003), Vol.II, pp.1070-1076, 2003.
14. T.M. Rutkowski, S. Seki, Y. Yamakata, K. Kakusho, M. Minoh, Toward the Human Communication Efficiency Monitoring from Captured Audio and Video Media in Real Environment, in Proceedings of Int. Conf. Knowledge-Based Intelligent Information & Engineering Systems (KES2003), Vol.II, pp.1093-1100 , 2003.
15. Daisaku Arita, Rin-ichiro Taniguchi, Non-verbal Human Communication Using Avatars in a Virtual Space, in Proceedings of Int. Conf. Knowledge-Based Intelligent Information & Engineering Systems (KES2003), pp. 1077-1084, 2003.
16. T. Shibata, D. Kawahara, M. Okamoto, S. Kurohashi, T. Nishida, Structural Analysis of Instruction Utterances, in Proceedings of Int. Conf. Knowledge-Based Intelligent Information & Engineering Systems (KES2003), pp. 1054-1061, 2003.
17. M. Ozeki, H. Izuno, M. Itoh, Y. Nakamura, Y. Ohta, Object Tracking and Task Recognition for Producing Interactive Video Content --- Semi-automatic indexing for QUEVICO, in Proceedings of Int. Conf. Knowledge-Based Intelligent Information & Engineering Systems (KES2003), pp.1044-1053, 2003.

Wrapper Generation for Automatic Data Extraction from Large Web Sites

Nitin Jindal

Department of Computer Science and Engineering,
Indian Institute of Technology Delhi
csu01124@cse.iitd.ernet.in

Abstract. The paper investigates techniques for extracting data from large set of dynamic web pages. Dynamically generated web pages from a single web site have a common semi structure for all the data objects. A wrapper of these dynamic web pages is defined as a common template for these pages with different data objects embedded in each web page. Information Extraction is done in three steps: (a) Data Rich Section Extraction from each web page (b) Automated generation of wrapper (c) Data extraction from each web page by comparing it with the wrapper. Wrapper generation is the most important part of this process. Our focus was on developing new improved techniques for wrapper generation. Our technique is fully automated and we were able to achieve good increase in accuracy and speed.

Keywords: Design of Web Based Information Systems, WWW and Database Systems, Web Application Systems, Data and Web Mining, Semi-Structured Data.

1 Introduction

The amount of information available on World Wide Web is very large and is growing very fast. Most of the information is available in the form of “Hidden Web”. Hidden Web is made up of the contents of thousands of specialized searchable databases that one can search via the Web. The search results from many of these databases are delivered in web pages that are generated just in answer to one’s search. Such pages are called “dynamically generated” pages. Such pages very often are not stored anywhere: it is easier and cheaper to dynamically generate the answer page for each query than to store all the possible pages containing all the possible answers to all the possible queries a person could make to the database. Search engines cannot find or create these pages. Even the few available search results are not very selective and it is not possible to exploit the structure in the data objects of a web page.

Information Extraction (IE) systems are web search systems that apply database search technology to web pages by identifying and extracting data objects present in the pages and then using the query language to query the extracted data object. A key component of these systems is wrapper, which is a set of extraction rules that helps extract data from the web pages and put them in a structured format.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 34 – 53, 2005.

Fortunately, most data rich web pages are dynamically generated. That is, after receiving a user request, a server script program fetches data from a back-end database and fills the data into a predefined HTML template. Such pages usually carry multiple instances of data objects that follow a particular alignment and format. This template is called the wrapper.

The problem of Information Extraction can be divided into four parts: (a) Crawling of dynamic web pages (b) Extraction of data rich section of a page (c) Construction of wrapper for data rich web pages with nested data objects (d) Using this wrapper to extract data from each page. This paper addresses the second and third parts of the above problem.

Commercial web pages usually contain noisy data in the form of advertisements, navigation panel, and logos and so on. Such information maybe helpful for navigating the website from user's point of view, but it complicates the process of data extraction. These are removed by comparing two web pages, identifying their common sub sections and filtering them. We investigated existing techniques for data rich section extraction and studied their advantages and disadvantages in various cases. The section on data rich section extraction explains these techniques and the result and analysis section analyzes these techniques.

The next phase involves construction of wrapper for the set of web pages. It is the most important part of data extraction. We developed a new technique to extract wrapper from a large number of html pages. It is based on matching technique called ACME [1], Align, Collapse, Match and Extract. Every HTML page is converted into XHTML [6], which is a reformulation of HTML 4 in XML 1.0 with the help of existing tools available, then it expressed as a regular expression. The wrapper is the least upper bound on these regular expressions. The results show good levels of accuracy and the constant upper bound on the wrapper was achieved as compared to other techniques.

The rest of the paper is organized as follows. Section 2 presents an overview of our approach. Section 3 explains data rich section extraction algorithm. Section 4 is on wrapper generation from the source pages, the ones with noisy data removed. Section 5 is results and analysis of our approach. Section 6 is conclusion.

2 System Overview

The domain of our research is data intensive websites. The data is stored in the form of a database on the website. Whenever a user sends a query through a form on the website, the relevant data is picked and displayed on the web page. This whole process is done dynamically, so the web page generated is dynamic web page. It is created on the fly based on user's request. This approach saves a lot of storage space, required for static web pages and is faster.

These data intensive websites have fairly regular structure. They carry multiple instances of data objects that follow a particular alignment or format. Information Extraction means inferring a grammar from HTML pages and then using this grammar to parse the web pages to retrieve the data objects in structured format.

Exact inferring of the target grammar (or equivalently, identification of the corresponding Deterministic Finite State Automata (DFA)) is a hard problem. Gold [7, 8] showed that the problem of identifying the minimum state DFA consistent with a presentation S comprising of a finite non-empty set of positive examples S+ and possibly a finite non-empty set of negative examples S- is a NP-hard problem. As a result, a large body of research that originated from Gold's seminal work has concentrated on the development of efficient algorithms that work in the presence of additional information (typically a set of labeled examples or a knowledgeable teacher's response to queries posed by the learner).

So a lot of pragmatic approaches have been followed for wrapper generation. Wrapper is generated by using additional information like a set of labeled examples or external tools. Sometimes, a priori knowledge about the schema is assumed, and also assumed that the data is in plain format. Most of these systems use single HTML page at a time to extract the wrapper.

We need a mechanism to discover not only those data objects with fixed number of attributes and values (plain structure) but also those with variable number of attributes and values (nested structure). For example, in a web site containing information on books, the tags enclosing the book will remain same from web page to web page. But, if any web page displays information on more then one book, these tags will be repeated in that web page.

This problem can be solved based on the close correspondence between nested types and Union Free Regular Expressions (UFRE). A regular expression consists of constants and operators that denote sets of strings and operations over these sets, respectively. Given a finite alphabet Σ the following constants are defined:

- (*empty set*) *∅* denoting the set *∅*
- (*empty string*) *ε* denoting the set *{ε}*
- (literal character) #pcdata in *Σ* denoting the set *{"#pcdata"}*

And the following operations:

1. (*concatenation*) *RS* denoting the set *{αβ* ǀ *α in* R *and β in* S*}.*
2. (*set union*) *R U* S denoting the set union of *R* and *S*.
3. (*Kleene star*) *R** denoting the smallest superset of *R* that contains *ε* and is closed under string concatenation. This is the set of all strings that can be made by concatenating zero or more strings in *R*.
4. *R*+ denoting *R* U R*.
5. *R?* denoting *R* ǀ *ε*.

Regular expressions have a straight forward mapping to nested structures of a web page. "#pcdata" maps to string fields, "+" maps to lists, possibly nested, "?" maps to optional fields.

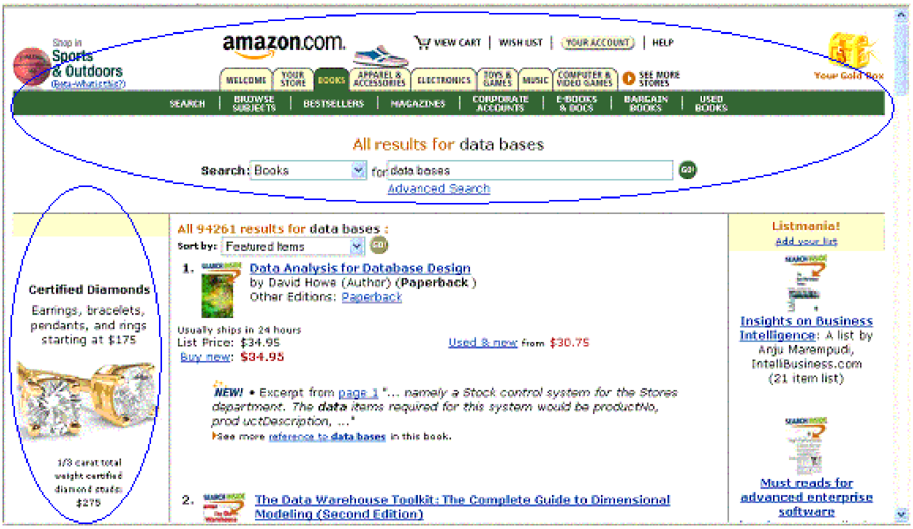
It is possible to show that, given a set of HTML strings s1, s2, . . . sk, corresponding to encodings of a source dataset, i.e., of a collection of instances i1, i2, . . . ik of a nested type, we can discover the type by inferring the minimal union-free regular expression whose language, L(σ), contains the strings s1, s2, . . . sk, and then taking Γ = type(σ). Also, we can use σ as a wrapper to parse s1, s2, . . . sk and reconstruct the source dataset i1, i2, . . . ik. Therefore, solving the schema finding and data extraction process amounts to finding the minimal UFRE (if this exists) whose language contains the input HTML strings, s1, s2, . . . sk. If we consider a lattice of UFRE with a containment relationship, such that σ1 ≤ σ2 iff L(σ1) is proper subset of L(σ2), then the UFRE we look for is the least upper bound of the input strings, i.e., σ = LUB(s1, s2, . . . sk). Since it is known that operator LUB is associative, this in turn amounts to computing the least upper bound of UFREs σ1 and σ2, LUB(σ1, σ2).

Though UFREs do not catch the full diversity of structure present in HTML pages, they have been shown to be [10] working well for describing structure of pages in fairly regular web sites. So, we concentrate on data intensive websites as they tend to have fairly regular structure.

Given various instances of a data object in a regular expression form, we can discover the minimal union free regular expression i.e. the least upper bound UFRE by iteratively going through each instance.

2.1 Data Rich Section Extraction

Template generated web pages contain lot of noisy data in the form of advertisements, navigational panel, logos and so on. Such information maybe helpful for navigating the website from user’s point of view, but it complicates the process of data extraction and label assignment. In Fig 1, the regions marked with ellipses are noisy sections. Bigger ellipse is navigation menu and smaller ellipse is advertisement.


Shop in Sports & Outdoors
amazon.com.
VIEW CART | WISH LIST | YOUR ACCOUNT | HELP
WELCOME
YOUR STORE
BOOKS
APPAREL & ACCESSORIES
ELECTRONICS
TOYS & GAMES
MUSIC
COMPUTER & VIDEO GAMES
SEE MORE STORES
Your Gold Box
SEARCH | BROWSE SUBJECTS | BESTSELLERS | MAGAZINES | CORPORATE ACCOUNTS | E-BOOKS & DOCS | BARGAIN BOOKS | USED BOOKS
All results for data bases
Search: Books
for data bases
GO!
Advanced Search
Certified Diamonds
Earrings, bracelets, pendants, and rings starting at $175
1/3 carat total weight certified diamond studs $275
Listmania!
Add your list
Insights on Business Intelligence: A list by Anju Marempudi, IntelliBusiness.com (21 item list)
Must reads for advanced enterprise software

All 94261 results for data bases :

Sort by: Featured Items

1. Data Analysis for Database Design
by David Howe (Author) (Paperback)
Other Editions: Paperback

Usually ships in 24 hours
List Price: $34.95 Used & new from $30.75
Buy new: $34.95

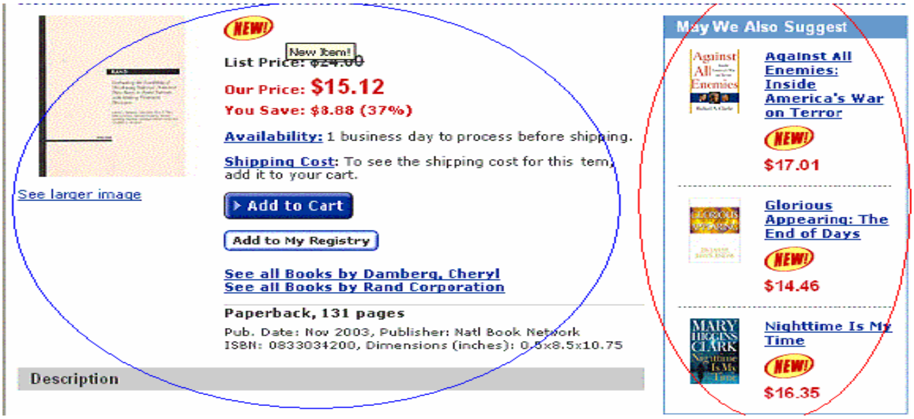
NEW! • Excerpt from page 1 "... namely a Stock control system for the Stores department. The data items required for this system would be productNo, prod uctDescription, ..."
See more reference to data bases in this book.

2. The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling (Second Edition)

Fig. 1. It is a webpage taken from amazon.com. Part of web page in elliptical regions is noisy data. *Horizontal ellipse* is navigation panel. *Vertical ellipse* shows an advertisement

The current approach is based on the fan out of a web page. Fan-out refers to the number of instances of a source dataset. It assumes that the useful information is hidden in that section of web page which has highest fan out. This approach handles only one web page at a time and extracts the data rich section from that web page.

This approach can go wrong in some cases. It does not work well in case of data pages with very small size of useful data. Let us consider a small example of a web page which consists of only 2 tables. The first one contains three entries of navigation panel (say, *home*, *previous* and *next*) and the second table contains only 2 entries of data. This approach tends to pick the first table due to larger fan out, even if it is the noisy data. For example in Fig 2, the region in left circle is useful data and the region in right ellipse is useless data. But, fan out of region marked by right ellipse is higher as it contains information about three books, so the current approach tends to pick up the wrong region.



NEW!

New Item!

List Price: ~~$24.00~~

Our Price: $15.12

You Save: $8.88 (37%)

Availability: 1 business day to process before shipping.

Shipping Cost: To see the shipping cost for this item, add it to your cart.

See larger image

▸ Add to Cart

Add to My Registry

See all Books by Damberg, Cheryl
See all Books by Rand Corporation

Paperback, 131 pages

Pub. Date: Nov 2003, Publisher: Natl Book Network
ISBN: 0833034200, Dimensions (inches): 0.5x8.5x10.75

Description

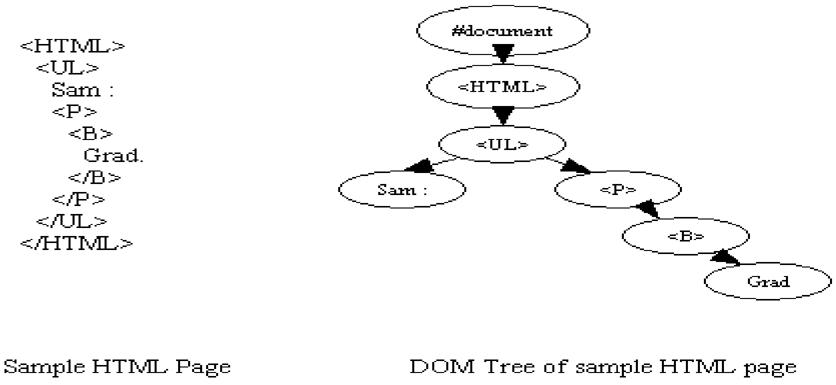
May We Also Suggest

Against All Enemies: Inside America's War on Terror
NEW!
$17.01

Glorious Appearing: The End of Days
NEW!
$14.46

Nighttime Is My Time
NEW!
$16.35

Fig. 2. It is a part of a web page taken from amazon.com. It shows disadvantage of fan-out approach for filtering noisy data. The region in *left ellipse* is actual data. *Right ellipse* contains secondary data. Since the fan-out of data in *left ellipse* is very small, it will get filtered with the above described approach



```
<HTML>
  <UL>
    Sam :
    <P>
      <B>
        Grad.
      </B>
    </P>
  </UL>
</HTML>
```

Fig. 3. It shows a *HTML page* and the corresponding *DOM Tree* build from that page, using JTIDY

The new approach exploits the fact that the useless information like navigation panels and advertisements has similar structure in all the web pages. They are placed at exactly the same position and depth in each dynamically generated web page, since it eases the user navigation and gives definite structure to the web site. The new approach is based on comparing two web pages. It takes two web pages as input and parses them to obtain the DOM (Document Object Model) Tree [3] of both the web pages. HTML file consists of tags and text between the tags, so it can be represented as a tree structure referred to as DOM. An example DOM Tree is shown in Fig 3.

We worked on simplified DOM tree of HTML pages because not all tags provide structural information for a page [5]. Given two DOM Trees, one matches the similar structures of these web pages. Basic idea is to traverse the two trees using depth first approach and then eliminate common subsections (same nesting of attributes and same string values) between the two trees.

2.2 Wrapper Generation

Our paper carries forward the matching technique used in [1] for wrapper generation. Most works on wrapper generation assume that target pages contain a collection of flat records with fixed number of attributes [11, 12, 13]; in other cases [9] the system may also handle nested data, but it needs to know what the attributes to extract are and how they are nested. Also, most works use one web page at a time to generate the wrapper. Whereas, the technique discussed in this paper uses two source pages at a time to generate the wrapper from them. Also, this technique is fully automated, i.e. it requires no user intervention. It also does not require any priori knowledge of schema of the web page, like labels, etc.

The technique discussed in [1] does not use any algorithm for generating wrapper from single page, but we used both the techniques of generating wrapper from a single page and as well as generating wrapper from two pages. The idea behind using the algorithm for generating wrapper from a single page is based on following observation. If a page contains two tables occurring adjacently, then it means that their can be multiple occurrences of table field in other html pages. For example, an HTML page with string representation <BODY><TABLE>….</TABLE><TABLE>… </TABLE> </BODY> can be represented as <BODY> (<TABLE> ….. </TABLE>) * </BODY>. The algorithm for generating wrapper from two pages will not be able to do this because it does not analyze the structure of a single page alone. The trick is to consider the two tables as separate source datasets and then to generate the generalized table, i.e. the wrapper of these two source datasets. So, the algorithm for generating wrapper from two HTML pages can be used here. So, the two tables can be considered as two separate source pages and their wrapper can be calculated by algorithm for wrapper generation from two pages.

This approach has varying levels of accuracy depending on many domain specific examples as compared to just generating wrapper from single page or generating wrapper from two pages. Moreover, the upper bound on the wrapper is achieved much earlier i.e. it requires lesser number of web pages belonging to a particular template, to generate the wrapper. This saves lot of time and storage requirements.

Our approach does not require any priori knowledge about the schema of web site or any user specified examples. The entire process is fully automated. This is discussed in details in results and analysis section. Next two sections discuss in detail the algorithms for Data Rich Section Extraction and Wrapper Generation.

3 Data Rich Section Extraction (DSE)

This section discusses the algorithm for Data Rich Section Extraction. The technique is based on the one discussed in [2]. The algorithm first converts the HTML pages into the DOM trees. Then the nodes from both DOM trees that are at the same depth are compared. All the subtrees that are at the same depth in both the Dom trees and are identical are eliminated. It is due to the fact that the noisy data is redundant; it tends to have same depth and same structure in every web page, to aid user navigation and updating of the web page easily. Only, the real data will differ from web page to web page.

Following functions are defined for node comparison.

1. RightSibling(parent_node, node_i) returns the right sibling of node_i that shares parent_node with node_i
2. TagName(node_i) returns the tag name of node_i
3. Attr(node_i) returns the attribute of the node_i, if the node is <A> or <IMG>, otherwise it returns null
4. Same(node_i, node_j) = 1, IF TagName(node_i) = TagName(node_j) AND Attr(node_i) = Attr(node_j) = 0, otherwise.

The algorithm works recursively. For two nodes, if they are internal nodes and are the *Same* (as defined above), then the algorithm will go down one level to match their children from the leftmost to the rightmost one. If they are leaf nodes and are the *Same*, then they will be removed from the trees. If the nodes are not the *Same*, the algorithm will return to their parent and continue to compare the other children, if any. If all of the children of two parent nodes have been compared, then they will be removed only if all of their children have been removed.

It is worth mentioning that HTML pages vary so much that even two pages that look the same in a browser can have different tag structures. For example, <TABLE><A>…</A></TABLE>, and <TABLE><TABLE><A>…</A></TABLE> </TABLE> result in different subtrees, but they look the same in a browser. Unfortunately, such variations of tag combinations can be infinite. In our tree-matching algorithm, we do not consider such cases. However, since the basic assumption of the DSE algorithm is that the same template generates HTML pages from the same web site, therefore, the tree structures of those web pages should be the same.

4 Wrapper Generation

After the data rich sections have been extracted from the html pages, the next task is to build a wrapper from these source pages.

The input that we have is a set of web pages extracted from a website. Let us call them as sample web pages. Initially any sample web page can be assumed as the starting wrapper. Then this wrapper is compared with other sample web pages, one at a time, thus generalizing the wrapper. The algorithm is based on the matching technique described in [1].

4.1 The Matching Technique

This section is devoted to the algorithm for wrapper generation from source pages, i.e. after extracting data rich sections. It is derived from matching technique called ACME [1], for Align, Collapse under Mismatch, and Extract. To avoid missing tags in the source web pages one can assume that the HTML code compiles to the XHTML [6] specification, a restrictive variant of HTML in which tags are required to be properly closed and nested. We also assume that source web pages have been preprocessed by a lexical analyzer to transform into a list of tokens, where each token is either an HTML tag or a string value.

The algorithm proceeds by comparing the existing wrapper and a new web page from the sample set, generalizing the wrapper in the process. Let us refer to the new input web page as Sample.

While comparing a Wrapper and a Sample their can be two types of mismatches

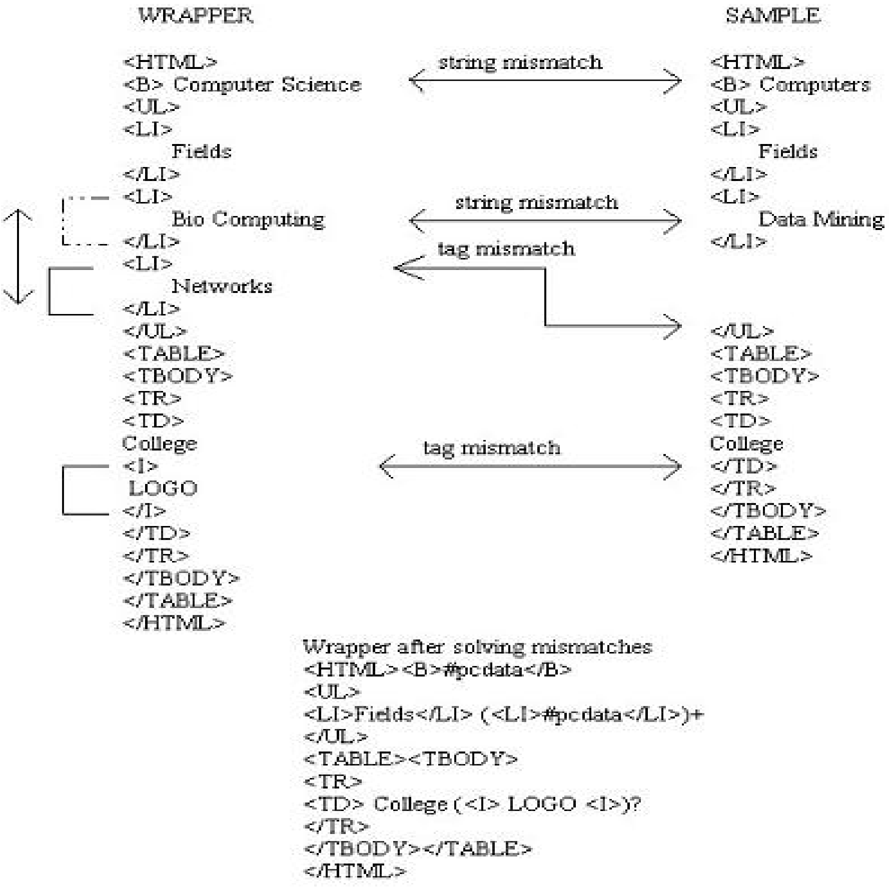
1. String Mismatch
2. Tag Mismatch

String Mismatch
If the text nodes do not match then the occurrence field of text node of wrapper is made #PCDATA at that node.

Tag Mismatch
Their can be three cases:-

1. Wrapper has starting tag and Sample has ending tag.
2. Wrapper has ending tag and Sample has starting tag.
3. Wrapper and Sample both have starting tags.

When a tag mismatch occurs we have to first identify the candidate square. Candidate square is the part of the sample that is discordant with the wrapper. It can also be a part of a wrapper that does not justify the sample page as the wrapper's specific case. It is a subtree or group of adjacent subtrees of the DOM tree. It is of the form <UL>……</UL>, <A>……</A><TABLE>……</TABLE>, etc. One has to generalize the wrapper, so that it includes the sample as its specific case. Now, after finding the candidate square, their can be two cases. Either the candidate square is repetitive or it is optional. If the candidate square is of the wrapper, then if, a similar square exists in the wrapper just above the candidate square, then it is repetitive, else it is optional. Similarly, if the candidate square is of a sample, then if a similar square exists in the sample just above the candidate square, then it is repetitive else it is optional. Also, we want to keep the optional tags minimum because they add redundant cases to the system For example, let us say, we have two samples *"ABC"* and *"ADC"*, the ideal wrapper should be *"A(B|D)C"*. But since we do not calculate *"OR"* relation, our wrapper will be *"AB?D?C*". This wrapper satisfies an extra string *"AC"* which is not in the sample set. So, we try to avoid as many optional tags as possible. That is why the repetitive tags are given preference over optional tags. Say, we have samples *"ABBC"* and *"ABDC*". When tag mismatch occurs at third position, instead of making *"D*" optional we make *"B*" repetitive. So, the resulting wrapper is *"A(B)*(D)?C*" instead of *"AB(D)?(B)?C*". The problem of finding the most appropriate candidate square and deciding whether it is optional or repetitive, has many cases and sub cases within them. Now we will discuss the tag mismatch cases separately.



```
<HTML>
<B> Computer Science
<UL>
<LI>
    Fields
</LI>
<LI>
    Bio Computing
</LI>
<LI>
    Networks
</LI>
</UL>
<TABLE>
<TBODY>
<TR>
<TD>
College
<I>
 LOGO
</I>
</TD>
</TR>
</TBODY>
</TABLE>
</HTML>
```

```
<HTML>
<B> Computers
<UL>
<LI>
    Fields
</LI>
<LI>
    Data Mining
</LI>
</UL>
<TABLE>
<TBODY>
<TR>
<TD>
College
</TD>
</TR>
</TBODY>
</TABLE>
</HTML>
```

Wrapper after solving mismatches

```
<HTML><B>#pcdata</B>
<UL>
<LI>Fields</LI> (<LI>#pcdata</LI>)+
</UL>
<TABLE><TBODY>
<TR>
<TD> College (<I> LOGO <I>)?
</TR>
</TBODY></TABLE>
</HTML>
```

Fig. 4. A simple example showing *wrapper* generation by mismatch

Tag Mismatch Case 1

Wrapper has starting tag and Sample has ending tag. Their can be two cases, i.e. Wrapper Candidate square can be either repetitive or optional. In case 1, i.e. when Wrapper candidate square is repetitive, their can be many cases:

1. Repetition of form (CS1) (CS1) where candidate square is (CS1).
2. Repetition of form (CS1)(CS2) (CS1)(CS2) where candidate square is (CS1)(CS2).
3. Repetition of form (CS1)(CS2)(CS3) (CS1)(CS2)(CS3) where candidate square is (CS1)(CS2)(CS3).

This can go on to any length of candidate square. We can limit this two maximum length of three based on our experience with real sites; otherwise it makes the maintaining and updating of web pages very tedious.

Wrapper candidate square is optional, if tag of Wrapper doesn't match the immediately proceeding tag.

Tag Mismatch Case 2
Wrapper has ending tag and Sample has starting tag. Same as *Tag Mismatch Case 1* except Sample has starting tag and Wrapper has ending tag.

Tag Mismatch Case 3
Wrapper and Sample both have starting tags. Their can be four sub cases:-

1. Wrapper candidate square is repetitive
2. Sample candidate square is repetitive
3. Wrapper candidate square is optional
4. Sample candidate square is optional.

Case 1 and *Case 2* are same as we discussed in *Sub Case 1* of *Tag mismatch Case 1*. The algorithm discussed in [1] does not discuss any sub cases in *Case3 and Case4*; instead it simply makes the candidate square as optional. Optional candidate squares can cause a lot of trouble in data extraction phase, as they can lead to large number of unwanted cases, as we discussed earlier. So, our aim was to minimize the optional tags. With some heuristics the candidate square to be made optional can be chosen efficiently. For example, important tags like <TABLE>…</TABLE> are less likely to be optional as compared to less important tags like <A>…</A>. *Case3* and *Case4* have many sub cases. Before we discuss these sub cases, let us assume:

- WC1 be Wrapper candidate square.
- SC1 be Sample candidate square.
- WC2 be Wrapper candidate square containing WC1.
- SC2 be Sample candidate square containing SC1

For example, consider <TABLE><TR>….</TR><A>…</A></TABLE>. If WC1 (or SC1) is <TR>…</TR>, then WC2 (or SC2) is <TABLE>…</TABLE>.

Four sub cases for *Case3* and *Case4* are:

1. If WC1 occurs in SC2 after SC1 but SC1 does not occurs in WC2 after WC1 then SC1 is optional. Since, their can be a possibility that WC1 of wrapper matches with WC1 of sample. So, it is better to make SC1 as optional node.
2. If SC1 occurs in WC2 after WC1 but WC1 does not occurs in SC2 after SC1 then WC1 is optional

3. If WC1 occurs in SC2 after SC1 and so do SC1 occurs in WC2 after WC1 then their can be two cases. If WC1 is already optional then it is kept optional and we proceed further. We want to keep the optional tags minimum in the final wrapper due to reason explained above. If WC1 is not optional then any of WC1 or SC1 is made optional.
4. If neither of WC1 and SC1 occurs in SC2 and WC2 respectively, then any one can be made optional candidate square. Here we can decide which candidate square to make optional based on their size or the preference order of the tags like table, UI, etc. We can use some observations like important tags like <TABLE>…</TABLE> are less likely to be optional as compared to less important tags like <A>…</A> and so on.

After determining the nature of candidate square, whether it is optional or repetitive, one has to generalize the wrapper to include this case. If the candidate square is optional, then it is inserted in the wrapper with optional node around it, like if *"A"* is optional, then it can be represented as *"A?"*, which means that *"A"* may or may not occur in the sample pages. If the candidate square is repetitive, then the square just above it in the wrapper is identified that matches with it. Then the candidate square is made a sample, the other square is made the wrapper, and the whole process is repeated on them. Then the more generalized wrapper is induced back to the larger wrapper, of which it was part of originally. Fig 4 shows a simple example for generating wrapper by above technique.

4.2 Wrapper of a Single Sample Page

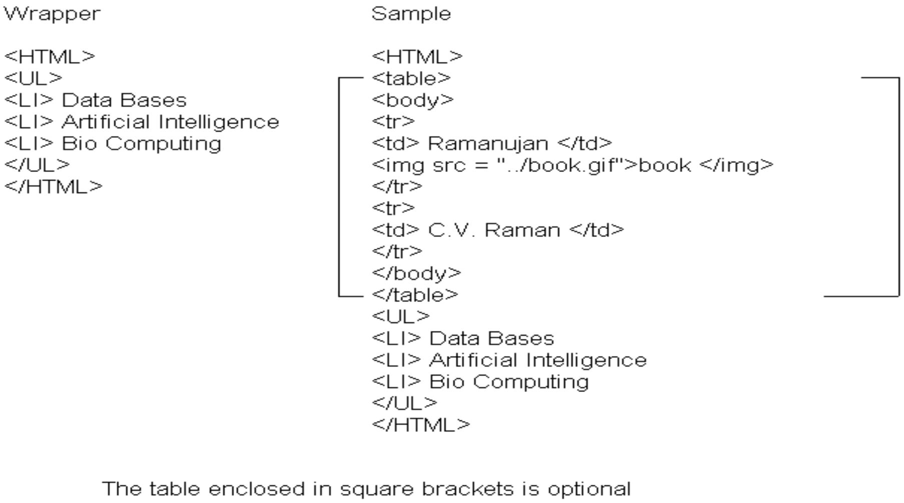
The starting wrapper, which we choose from any of the available sample pages, can also be generalized to find the least upper bound of that sample page alone. Similarly, when we add a new optional candidate square to a wrapper or make the existing one optional, then this candidate square can also be generalized before inserting in the wrapper. The generalized form is called wrapper of single page, since it is produced by examining only one page.

The algorithm for the wrapper for single page can be used in following two ways:-

1. To generalize the starting wrapper. As said earlier, the starting wrapper is a randomly chosen web page from the set of web pages.
2. To generalize any sub tree that is added to the wrapper as an optional. To illustrate this point, consider Fig 5 and Fig 6. In Fig 5, while comparing the wrapper and the sample, the table (candidate square identified when tag mismatch occurs in position 2) in the sample is optional. So, before inserting it in the wrapper, it is first generalized, as shown in Fig 6. We can insert the table in the wrapper as it is or we first generalize it and then insert it in the Wrapper. The best choice is to insert the compressed table in the original wrapper because it represents the template of sample page more closely.

The algorithm for generating the wrapper of a single source page uses the algorithm for generating a wrapper from two source pages. It proceeds by examining adjacent subtrees of the DOM tree of the source page starting at depth 1. If the root nodes of adjacent subtrees match, these subtrees are treated as separate source pages and their wrapper is calculated using the technique for generating the wrapper from two pages. Following is the algorithm for generating wrapper for single source page:

```
INPUT    Root node of Dom Tree of the source page
OUTPUT   Wrapper of the source page
PROCEDURE GenerateWrapperSinglePage (Node node_i) ::
BEGIN
FOR EACH childNode of node_i
 GenerateWrapperSinglePage (node_i.childNode)
END FOR
IF (node_i = node_i.nextSibling) THEN
wrapper := node_i
sample := node_i.nextSibling
IF (wrapper.startingTag = sample.startingTag) THEN
delete node.nextSibling
GenerateWrapperManyPages (wrapper, sample)
node_i = wrapper
GenerateWrapperSinglePage(node_i)
ELSE
GenerateWrapperSinglePage (node_i.nextsibling)
END IF
ELSE
GenerateWrapperSinglePage(node_i.nextSibling)
END IF
END PROCEDURE
```



Wrapper

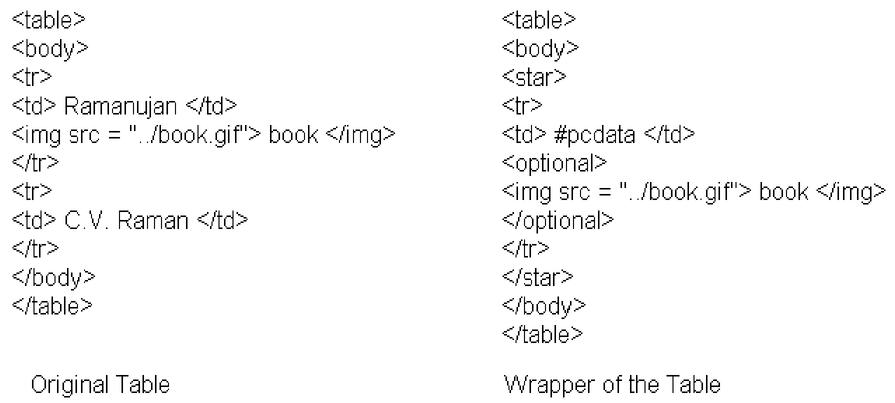
```
<HTML>
<UL>
<LI> Data Bases
<LI> Artificial Intelligence
<LI> Bio Computing
</UL>
</HTML>
```

Sample

```
<HTML>
<table>
<body>
<tr>
<td> Ramanujan </td>
<img src = "../book.gif">book </img>
</tr>
<tr>
<td> C.V. Raman </td>
</tr>
</body>
</table>
<UL>
<LI> Data Bases
<LI> Artificial Intelligence
<LI> Bio Computing
</UL>
</HTML>
```

The table enclosed in square brackets is optional

Fig. 5. On left side of picture is the *wrapper* and on right is the *sample pag*e, which is being used currently to update the wrapper



```
<table>
<body>
<tr>
<td> Ramanujan </td>
<img src = "../book.gif"> book </img>
</tr>
<tr>
<td> C.V. Raman </td>
</tr>
</body>
</table>
```

Original Table

```
<table>
<body>
<star>
<tr>
<td> #pcdata </td>
<optional>
<img src = "../book.gif"> book </img>
</optional>
</tr>
</star>
</body>
</table>
```

Wrapper of the Table

Fig. 6. It shows the use of wrapper for single page algorithm. On left is the *table* and on right is the *wrapper* of the *table* produced by wrapper generation for single page algorithm. <*star*> refers to "*" and <*optional*> refers to *"?"*

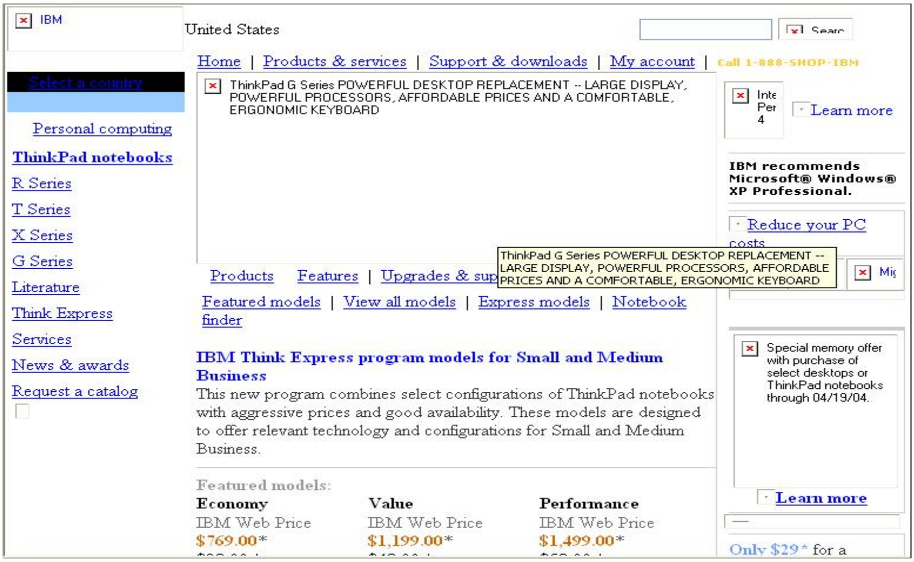
5 Experimental Results and Analysis

Our system is written in java and our experiments were performed on PC equipped with an AMD athlon XP 1.6GHz processor with 256Mbytes of RAM, running Windows XP and Sun JDK 1.3. HTML pages from several data intensive web sites were downloaded to run our experiments. The dynamic web pages were extracted from IBM website using a form crawler. Five Hundred dynamic web pages were used to carry the experiments. We used JTidy, a Java port of HTML Tidy [4] to convert HTML to XHTML [6] and clean our data sources. HTML Tidy is a library for cleaning HTML documents.

We analyzed following techniques discussed in this paper: 1) Data Rich Section Extraction (DSE), 2) Wrapper generation as whole and 3) Use of wrapper generation for single web page in wrapper generation algorithm. The sample data set was approx. 500 dynamic web pages crawled from IBM website with the help of a form crawler. These web pages contained product information about various items like notebooks, monitors, handhelds, workstations, etc. So web pages were very data intensive. Each web page contains information about the main product, related products of same category and links to other related category of products, etc.

We will first analyze the performance of DSE (Data Rich Section Extraction) algorithm. The DSE algorithm was analyzed on various kinds of web pages, mainly from Amazon and IBM websites. It worked well on all the cases. For example, Fig 7 is original web page and Fig 8 is the source page after applying DSE algorithm. The top navigation menu, left navigation menu and advertisements tags on the right side of the web page were neatly removed stressing the utility of DSE algorithm.

Now we analyze the wrapper generation technique. The sample data set that was used consisted of approx. 500 dynamic web pages crawled from IBM website through a form crawler. To verify that the wrapper generated is correct, a test was conducted to extract a particular table from the sample web pages with the help of the wrapper.



ThinkPad G Series POWERFUL DESKTOP REPLACEMENT -- LARGE DISPLAY, POWERFUL PROCESSORS, AFFORDABLE PRICES AND A COMFORTABLE, ERGONOMIC KEYBOARD

IBM recommends Microsoft® Windows® XP Professional.

ThinkPad G Series POWERFUL DESKTOP REPLACEMENT -- LARGE DISPLAY, POWERFUL PROCESSORS, AFFORDABLE PRICES AND A COMFORTABLE, ERGONOMIC KEYBOARD

IBM Think Express program models for Small and Medium Business

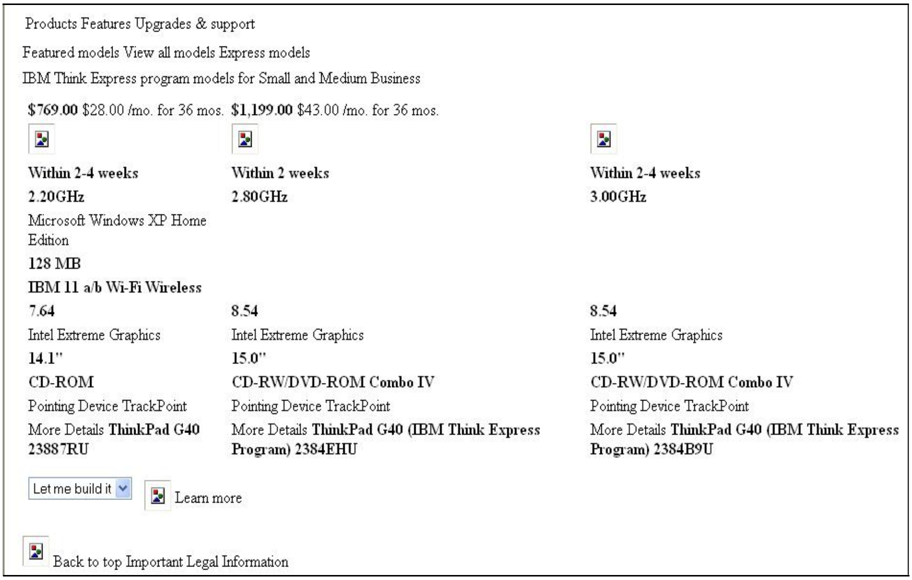
This new program combines select configurations of ThinkPad notebooks with aggressive prices and good availability. These models are designed to offer relevant technology and configurations for Small and Medium Business.

Special memory offer with purchase of select desktops or ThinkPad notebooks through 04/19/04.

Featured models:

Economy	Value	Performance
IBM Web Price	IBM Web Price	IBM Web Price
$769.00*	$1,199.00*	$1,499.00*

Fig. 7. Part of a sample web page taken from ibm.com. It shows product wise information on IBM Think Pads



Products Features Upgrades & support

Featured models View all models Express models

IBM Think Express program models for Small and Medium Business

$769.00 $28.00 /mo. for 36 mos. **$1,199.00** $43.00 /mo. for 36 mos.

Within 2-4 weeks	**Within 2 weeks**	**Within 2-4 weeks**
2.20GHz	**2.80GHz**	**3.00GHz**
Microsoft Windows XP Home Edition		
128 MB		
IBM 11 a/b Wi-Fi Wireless		
7.64	**8.54**	**8.54**
Intel Extreme Graphics	Intel Extreme Graphics	Intel Extreme Graphics
14.1"	**15.0"**	**15.0"**
CD-ROM	**CD-RW/DVD-ROM Combo IV**	**CD-RW/DVD-ROM Combo IV**
Pointing Device TrackPoint	Pointing Device TrackPoint	Pointing Device TrackPoint
More Details **ThinkPad G40 23887RU**	More Details **ThinkPad G40 (IBM Think Express Program) 2384EHU**	More Details **ThinkPad G40 (IBM Think Express Program) 2384B9U**

Fig. 8. Source page obtained after extracting data rich sections from the web page shown in Fig 7. All the labels, navigation menu and advertisements have been removed. Only the data is retained

First, the wrapper was expressed as an XML [14] document. Then the XPATH [15] string of that table was generated from the wrapper. The XPATH string is the path of a subtree in the XML document from the root node. It is substring of the wrapper and also it is in the form of a regular expression. For example, the XPATH string of *"F"* in *"<A><C>E</C><optional><B>F</B></optional></A>"* is *"A[2]optional[1]B[1]"*, where *"A[n]"* means nth child of subtree rooted at *"A"*. Since the wrapper is the generalized form of sample web pages, all the XPATH strings of *"F"* in the sample web pages will satisfy *"A[2]optional[1]B[1]"*. So, all the XPATH strings of that particular table were taken from each sample web page. It was found that the resulting XPATH strings satisfied XPATH string generated from the wrapper of that table (which was in the form of regular expression). This proves the correctness of the algorithm.

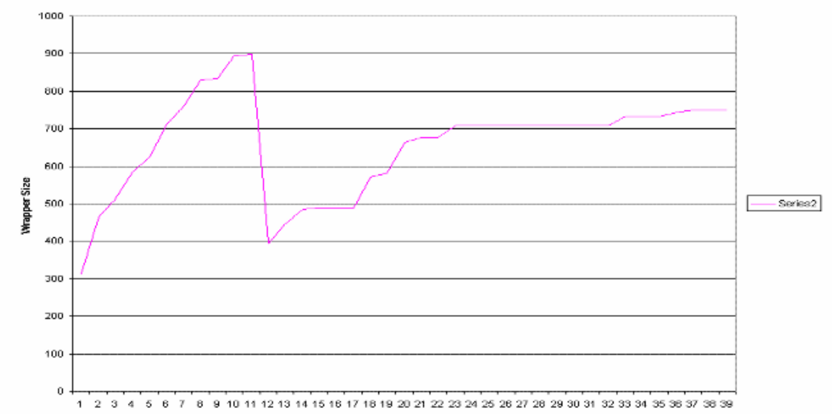


Fig. 9. Plot of Wrapper Size V/S Number of Sample Pages (after extracting data rich sections) with wrapper for single page used. Wrapper size more or less stabilizes after roughly 25 source pages have been used. Wrapper size is roughly 800 tokens (tags and #pcdata)

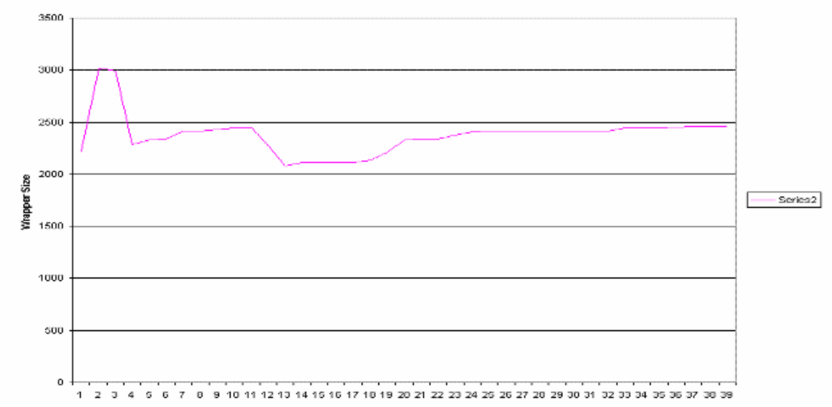


Fig. 10. Plot of Wrapper Size V/S Number of Sample Pages (after extracting data rich sections) with wrapper for single page not used. Wrapper is stabilized after roughly 28 sample pages. Wrapper size is roughly 2500 tokens (tags and #pcdata)

The graph of wrapper size v/s the number of sample web pages analyzed was plotted for following cases:

1. Generating Wrapper with the help of wrapper for single page technique (Fig 9).
2. Generating Wrapper without using wrapper for single page technique (Fig 10).

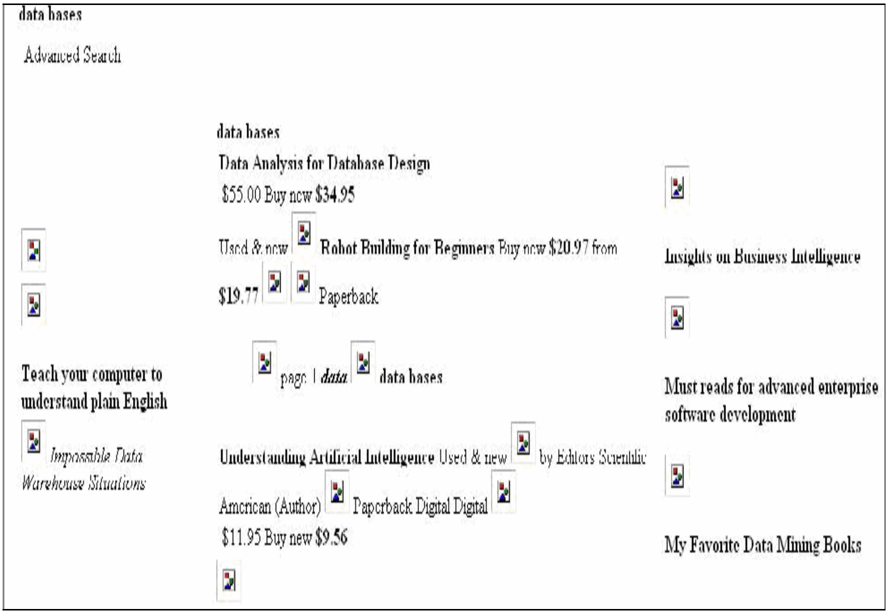
As shown in the figures, the wrapper stabilizes after some number of sample pages in both cases after less than 50 web pages, though we had 500 web pages to analyze. The final wrapper generated does not depend on the order of the sample web pages analyzed which was also proved by the experiments.

Wrapper was generated for different orders of sample set of web pages. It was found that the when the sample web pages were analyzed in decreasing order of token length (number of tags and strings), the stable wrapper was obtained with much less number of web pages. The reason is that the large sample web page will have more number of likely candidate squares. So, the number of new candidate squares to be added to the wrapper decreases more rapidly as we analyze more pages in decreasing order. Thus, the stable wrapper is achieved much earlier.

However the wrapper generation technique has one major disadvantage. It does not includes "OR" attribute in wrapper generation. This can lead to unwanted expressions that can be satisfied by the wrapper. For example, consider the case of two strings "ABD" and "*ACD*". Our algorithm will give "*AB?C?D*" as wrapper. But this wrapper satisfies the string "*AD*", which does not belong to the sample set. This can only be removed by using "*OR*" attribute. Say, if the resulting is "AB|CD", where "*B|C*", means either "*B*" or "*C*".

Now we will discuss advantages as well as disadvantages of using wrapper for single page in the entire wrapper generation algorithm. First we will consider advantages. When a wrapper is generated from a set of sample pages, after examining a fixed number of sample pages, the size of wrapper becomes constant, so their is no need to examine remaining sample pages. When the wrapper for single page is used, this constant size is achieved much earlier. Thus, it saves lot of computation time. Wrapper for single page algorithm was tested for a variety of web pages. It was found that it performed better for web pages with more degree of regular structure. For example consider a book website, say Amazon.com. All the books represented anywhere in the web page will usually have a standard structure consisting of its price, 1st edition, authors, etc. So, it is wise to club together all such data instances into a single field. So, Wrapper for single page helps us to achieve this clubbing to a greater accuracy.

Let us discuss a specific case where wrapper for single page worked to our advantage. In Fig 11, the wrapper without using wrapper for single page is not compressed properly. The tables in the second and the third column both contain information on the books in same format. So, the ideal wrapper should have merged these tables to form (TABLE)*. When wrapper for single page is used, better results are obtained as shown in Fig 12. Both the tables are merged and wrapper is more generalized form.



data bases

Advanced Search

Teach your computer to understand plain English

Impossible Data Warehouse Situations

data bases

Data Analysis for Database Design

$55.00 Buy new $34.95

Used & new Robot Building for Beginners Buy new $20.97 from $19.77 Paperback

page 1 *data* data bases

Understanding Artificial Intelligence Used & new by Editors Scientific American (Author) Paperback Digital Digital

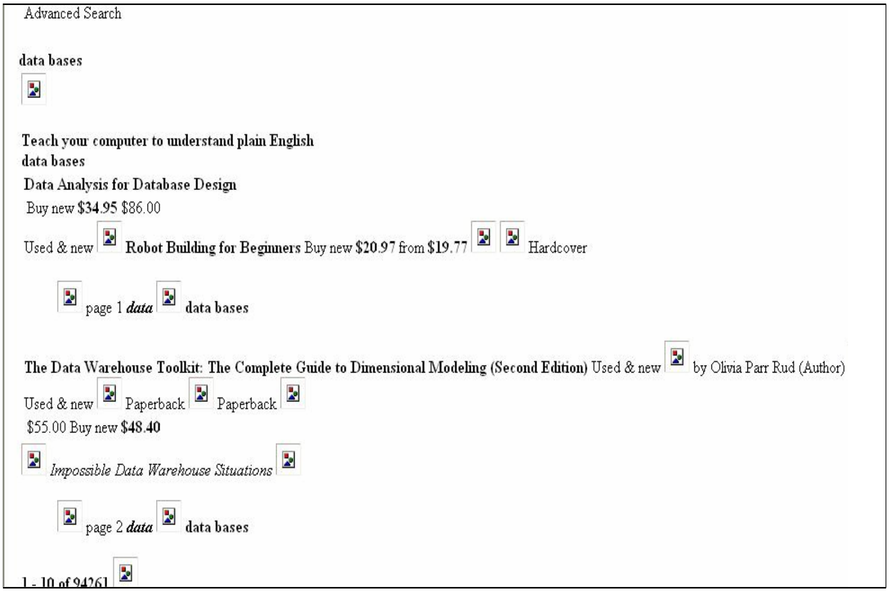
$11.95 Buy new $9.56

Insights on Business Intelligence

Must reads for advanced enterprise software development

My Favorite Data Mining Books

Fig. 11. Wrapper obtained for some sample set of amazon.com web pages. The wrapper for single page is not used. The entries in 2nd column and 3rd column contain information on books in same format. They should have been merged, since they belong to same structure



Advanced Search

data bases

Teach your computer to understand plain English

data bases

Data Analysis for Database Design

Buy new $34.95 $86.00

Used & new Robot Building for Beginners Buy new $20.97 from $19.77 Hardcover

page 1 *data* data bases

The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling (Second Edition) Used & new by Olivia Parr Rud (Author)

Used & new Paperback Paperback

$55.00 Buy new $48.40

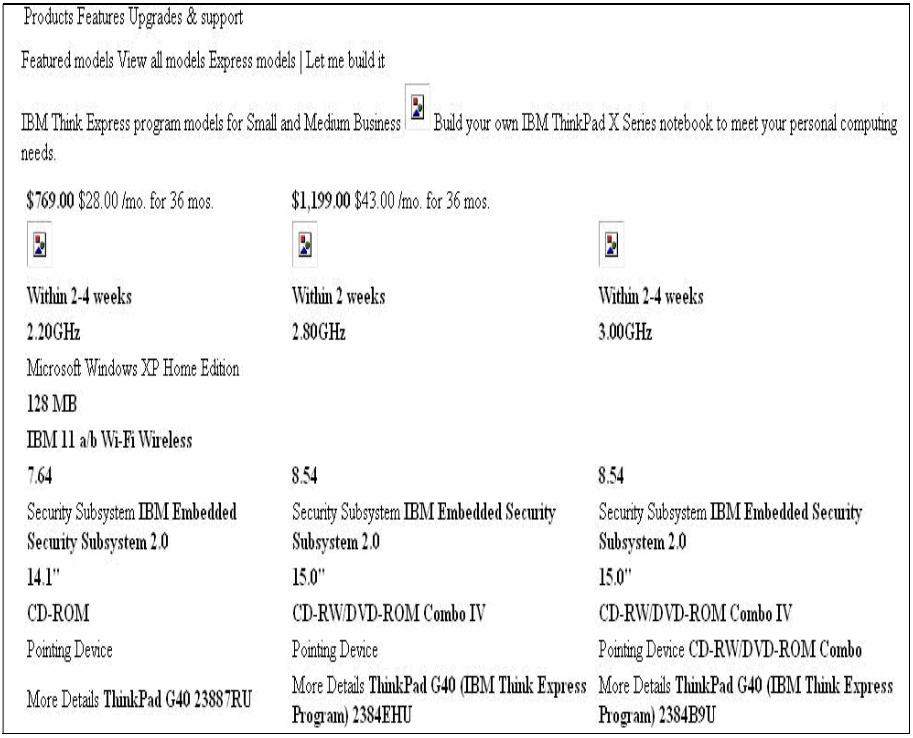
Impossible Data Warehouse Situations

page 2 *data* data bases

1 - 10 of 94261

Fig. 12. Wrapper obtained from exactly the same sample set of amazon.com web pages as used for Fig 11. In this case the wrapper for single page is used. The 2nd and 3rd column in Fig 11 have been merged. So, the resulting wrapper is more probable representation of template of those web pages

However, wrapper for single page failed in some circumstances. Some data intensive websites do not have information on just one category of product, i.e. unlike the case of a book website. For example in Fig 13, the three tables represent three different categories of IBM ThinkPad. So, it is wise to keep these fields separate in the wrapper, which will help in effective retrieval of data instances from sample web pages with the help of the wrapper. So, they are not to be compressed any further. But after applying wrapper for single page, the resulting wrapper is the Fig 14. Clearly, the data extraction and label assignment will become very difficult since the structure of wrapper in immediate figure more closely represents the template of web pages as compared to the one shown after.



Products Features Upgrades & support

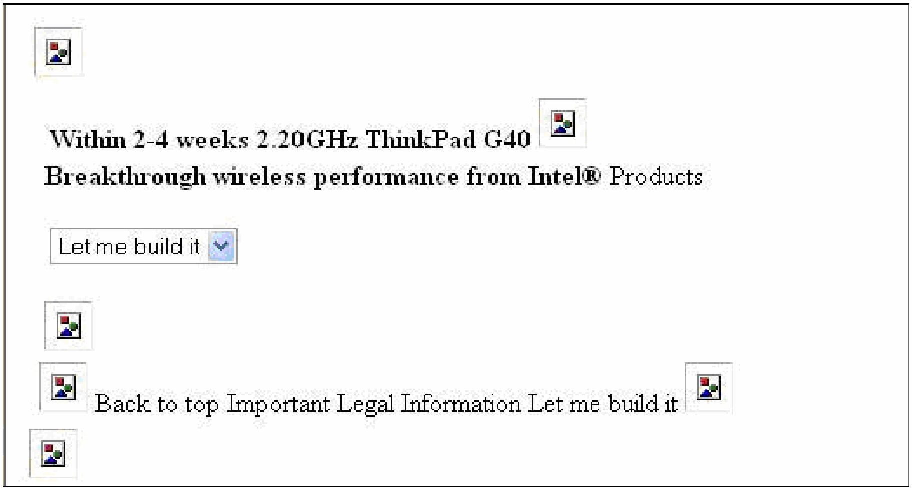
Featured models View all models Express models | Let me build it

IBM Think Express program models for Small and Medium Business Build your own IBM ThinkPad X Series notebook to meet your personal computing needs.

$769.00 $28.00 /mo. for 36 mos.	$1,199.00 $43.00 /mo. for 36 mos.	
Within 2-4 weeks	Within 2 weeks	Within 2-4 weeks
2.20GHz	2.80GHz	3.00GHz
Microsoft Windows XP Home Edition		
128 MB		
IBM 11 a/b Wi-Fi Wireless		
7.64	8.54	8.54
Security Subsystem IBM Embedded Security Subsystem 2.0	Security Subsystem IBM Embedded Security Subsystem 2.0	Security Subsystem IBM Embedded Security Subsystem 2.0
14.1"	15.0"	15.0"
CD-ROM	CD-RW/DVD-ROM Combo IV	CD-RW/DVD-ROM Combo IV
Pointing Device	Pointing Device	Pointing Device CD-RW/DVD-ROM Combo
More Details ThinkPad G40 23887RU	More Details ThinkPad G40 (IBM Think Express Program) 2384EHU	More Details ThinkPad G40 (IBM Think Express Program) 2384B9U

Fig. 13. Wrapper obtained without using wrapper for single page on some sample from ibm.com. Clearly, the *three columns* represent different structure of data. So, they can't be merger further. So, the wrapper is the best possible choice with current algorithm

It can be concluded that the use of wrapper for single page in generating the wrapper should be based on some advance knowledge about the nature of web pages. It should be based on some degree of homogeneity of the web page. If the web pages contain lot of homogenous data, say a book web page which normally contains list of books represented in same format then this approach is useful. But consider another web page like a person's bio data. This web page will contain a lot of heterogeneous information. So, it is not wise to compress this page using wrapper for single page. The information can get jumbled up. So, the number of possible queries needed to extract the information from web pages using the resulting wrapper can jump astronomically.



Within 2-4 weeks 2.20GHz ThinkPad G40

Breakthrough wireless performance from Intel® Products

Let me build it

Back to top Important Legal Information Let me build it

Fig. 14. Wrapper obtained after using wrapper for single page technique on the same set of sample pages as used for generating wrapper in Fig 13. Clearly, the structure of wrapper is completely lost. So, the use of wrapper for single page worked to our disadvantage in this case

6 Conclusion

In this paper we described a technique for extracting a common structure called wrapper from a large set of web pages. We first extract data rich sections from various web pages and then generate the wrapper from the resulting source pages. Wrapper generation is done by examining 2 source pages at a time and also using wrapper generation for a single source page. Our experiments on several dynamically generated web pages indicated good increase in accuracy as compared to previous techniques. We also discussed cases, where our wrapper generation technique faulted.

As future work, we plan to develop a complete model for data extraction from large web sites. We plan to work on generating the database with the help of wrapper, label assignment to various attributes of database table, and query modeling. We also, plan to refine our algorithm to obtain better and more accurate results.

Acknowledgment

This work was done under the guidance and encouragement of Prof S. K. Gupta, Department of Computer Science and Engineering, Indian Institute of Technology Delhi, India.

References

1. Valter Crescenzi, Giansalvatore Mecca, Paolo Merialdo. "ROADRUNNER: Towards Automatic Data Extraction Large Web Sites". *VLDB 2001.*
2. Jiying Wang, Fred H. Lochovsky. "Wrapper Induction based on Nested Pattern Discovery." Technical Report HKUST-CS-27-02, Dept. of Computer Science, Hong Kong U. of Science and Technology, 2002.
3. "Document Object Model Level 3 Core Specification". World Wide Web Consortium *2001.*
4. "HTML Tidy". http://www.w3.org/People/Raggett/tidy/.
5. "HTML 4.01 specifications", http://www.w3.org/TR/REC-html40/. World Wide Web Consortium *1999.*
6. "XHTML 1.0. The Extensible HyperText Markup Language". http://www.w3.org/TR/xhtml1/. *2002.*
7. E. M. Gold. "Language Identification in the limit. Information and Control". *10(5), 1967.*
8. E. M. Gold. "Complexity of automaton identification from given data. Information and Control." *37(3), 1978.*
9. B. Adelberg. "NoDOSE – a tool for semi-automatically extracting structured and semi-structured data from text documents". *SIGMOD 1998.*
10. P. Atzeni, G. Mecca, and P. Merialdo. "To Weave the Web". *VLDB 1997*
11. C. H. Chang, S. C. Lui. "IEPAD: information extraction based on pattern discovery". *World Wide Web 2001.*
12. N. Kushmerick, D. Weld and R. Doorenbos, "Wrapper induction for information extraction," *Artificial Intelligence*, 729-735, 1997.
13. Muslea, S. Minton and C. Knoblock. "A hierarchical approach to wrapper induction". *Autonomous Agents.* 1999.
14. "Extensible Markup Language (XML)". http://www.w3.org/XML/.
15. "XML Path Language (XPATH)". http://www.w3.org/TR/xpath. *World Wide Web Consortium. 1999.*

Efficiently Coding and Querying XML Document

Zhongming Han and Niya Fu

College of Computer Science and Technology of Donghua University,
1882 Yananxi West Road, Shanghai (200051), P.R. China
{hx_zm, funiya}@mail.dhu.edu.cn

Abstract. At the moment most of index structures and query algorithms for XML documents are constructed on region-based numbering scheme. However this numbering scheme suffers from some drawbacks. In this paper, a novel and efficient numbering scheme is presented, which combines the label path information and data path information, and it can efficiently support all kinds of queries. Properties of this numbering scheme are discussed in detail. Query algorithms based an index structure, named HiD, are introduced. At last, comprehensive experiments are conducted to demonstrate the efficiency of our query algorithm.

1 Introduction

Nowadays, several query languages, including XPath [16], Quilt [15], and XQuery [14], have been proposed for XML Document. To efficiently query document by these query languages, efficient index structure and query algorithm are necessary. Furthermore, an efficient numbering scheme provides the foundation for an efficient index structure. A numbering scheme is to encode nodes in an XML document and produce the node identification.

Now, researchers have proposed many kinds of query and index technologies, such as EF-Join, EA-Join, KC-join [5], MPMCJN [6], tree-merge, XPATH Accelerator, Containment Join etc, which are based on the structure join. Other approaches, like RiST and ViST[13] , use tree structures as the basic unit of query to avoid expensive join operations. All of these query approaches are constructed on some numbering scheme on XML documents. Among these numbering schemes, two types of numbering methods stand out, one is the region-based numbering scheme and the other is DataGuide.

There are some problems with the region-based numbering schemes, including some extended preorder numbering schemes. Firstly, they lack of flexibility. It is very difficult to support dynamic nodes inserting, deleting and modifying operations although researchers proposed some mechanisms to enlarge the range of region. Meanwhile, maintenance of numbers of nodes needs a lot of time. Moreover, region-based numbering schemes do not contain path information of nodes. Finally, region numbering schemes do not support queries in the form of document order, which is the relative ordering existing among nodes within a single document, and which is used in XPath and related domains such as XSLT and XQuery.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 54 – 69, 2005.

DataGuide provides another numbering scheme, in which the path information and node position are combined. In this numbering scheme, more information is provided. However, in this numbering scheme, nodes are also represented by the starting number of pre-order traversing XML document, and thus it is still very difficult to support dynamic nodes inserting, deleting and modifying operations.

In order to improve support dynamic operation and performance, in this paper, we propose a new novel node numbering scheme. We distinguish the label path information from the data path information for a node. The label path is transformed into a binary tree and Haveman's code is generated to serve as the label path identification. As for the data path, we adopt DataGuide's numbering scheme. The following is the main contributions.

- A novel and efficient node numbering scheme is proposed in this paper. This node identification contains the label path information and data information and efficiently supports all kinds of queries.
- Query algorithms based on the node scheme and an index structure HiD are introduced.
- The comprehensive experiments are conducted to evaluate query algorithm.

The rest of the paper is structured as follows. In Section 2, basic concepts, including XML document, label path and data path are introduced and the node identification is defined. In Section 3, the index structure is proposed. In Section 4, the focus is on query algorithms. How to handle wildcards "//" and "*" is introduced in this section. In Section 5, through some experimental results, we analyze the presented algorithms and compare them with other existing ones. In Section 6, we review related work. Finally we make some concluding remarks in Section 7.

2 Node Identification

We assume that a single document *D* is a node labeled acyclic tree with the set *V* of nodes and the set *E* of edges, and labels are taken from the set *L* of strings. Furthermore, text values come from a set *T* of text strings and can be attached to any type of nodes, not just leaf nodes. Formally, we define an XML document *D* is a tree (*V,* root, *label*, *children*, *text*), where *V* is a finite set of nodes; root$\in V$ is the root of *D*; *label* is a mapping: $V \rightarrow L$; *children* is a mapping from nodes to a partially ordered sequence of child nodes and induces the set of *E* of edges; *text*: $V \rightarrow T$. **Fig. 1** shows an XML document.

The following definitions introduce some fundamental notions used in the rest of the paper.

Definition 1 (Label Path)
Let n_1, $n_2 \in V$ be two nodes in an XML document. The unique label path from n_1 to n_2, denoted by *lpath*(n_1, n_2), is the label path ($l_1, l_2, \cdots l_k$) where l_1=*label* (n_1) and l_k= *label* (n_2), $l_i \quad (i = 2, \cdots, k - 1)$ correspond to the sequence of labels on the path in *D* connecting the two nodes. If there is no such connecting path, then *lpath*(n_1, n_2)=*e*

(the empty path). The rooted label path of a node n∈*V*, *lpath*(n), is the label path from the root node to the node n.

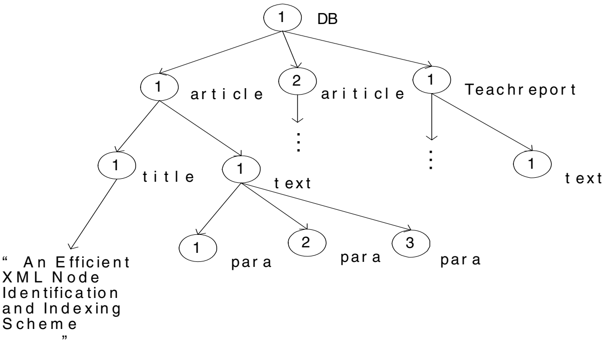


Fig. 1. An XML Document

Definition 2 (Node Position and Arity)
Let n_l and n_2 be two nodes in an XML document. If n_2 is a child node of n_l, i.e. $n_2 \in \mathit{children}(n_l)$, then *position*($n_2$) is the position of node n_2 in $\mathit{children}(n_l)$ with respect to nodes that have the same label *label*(n_2). The arity of node $n_{2,}$ *arity*(n_2), is the number of nodes with label *label*(n_2) in the n_l's list of children.

Definition 3 (Data Path)
The data path from the node n_l to n_k in an XML document *D*, *dpath*(n_l, n_k), is the sequence of positions of nodes along the label path from n_l to n_k in *D*. That is, if (n_l, n_2, ⋯, n_k) $n_i \in V$ (i=1,2…k), is the path from $_l$ to n_k, then the sequence of positions is (*position*(n_l), *position*(n_2), ⋯, *position*(n_k)).

If n_l is the root node, then *position*(n_l)=1. The rooted data path of n_k, *dpath*(n_k), is the data path *dpath*(root, n_k). The alternative representation of data path is *position*(n_l).*position*(n_2). … . *position*(n_k).

2.1 Node Label Path Number

A tree always can be transformed to a binary tree without ambiguity. Thus we transform an XML label path tree to a binary tree. We illustrate the transformation by the following example:

Definition 4 (Label Path Number)
Let n be a node in an XML document, and the bit string *S* be the Haveman's code of the node of the label path tree. Then the label path number of n is the decimal number to which the bit string is transformed.

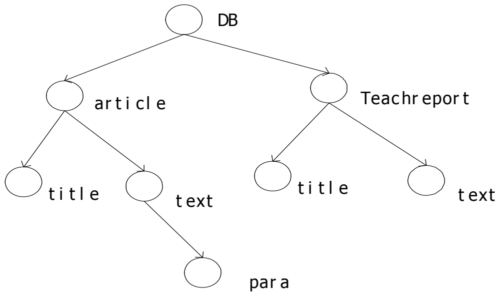


Fig. 2. An XML Label Path Tree

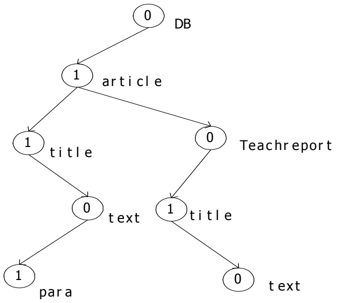


Fig. 3. The Binary Tree with Fig. 2

Fig 2 shows an XML document label path tree of the XML document listed in **Fig. 1**, which consists of all label paths and reflects the structural information in the XML document. A node of the tree corresponds to a label of the document. It is easy to know all the label paths of nodes from this tree. Then we transform this tree into a binary tree shown in **Fig 3.**

Fige 3 shows the binary tree. There are many methods to code a binary tree. However we seek a coding method that can reflect path information from the number of nodes, so we choose Haveman's code. The number labelled with each node is produced according to Haveman's code. In **Table 1**, Haveman's code of nodes can be found. Thus, the label path of a node in a document would be expressed by Haveman’s code of the node in the label path tree.

Table 1. Node Numbering

Node	Code	Arity	Data Path	Data path Number	Node identification
DB	0	1	0	0	(0,0)
Article	01	2	1,2	0,1	(1,0) (1,1)
Teachreport	010	2	1	0	(2,0)
Title	011	1	1	0	(3,0)
Text	0110	1	1.1	01	(6,1)
Para	01101	3	1.1,1.2,1.3	000,001,010	(13,0),(13,1), (13,2)
Title	0101	1			
Text	01010	1	1.1	00	(10,0)

Now we analyse order relationships between brother nodes. To obtain the order relation, we draw the following conclusions:

Proposition 1. Let m be a node and n be the next brother node of node m in a label path tree. Then Haveman's code of n is Haveman's code of m plus 0.

Proposition 2. Let m and n be two brother nodes in a label path tree. Then Haveman's code of two nodes has the following properties:

- Haveman's code of one node is prefix to the other node.
- The remaining parts of Haveman's code are all 0 after removing the common prefix.

Proposition 3. Let a and b be two nodes in an XML document, and m and n be the label path number of a and b respectively. Then a sufficient and necessary condition for a to be an ancestor of b is that

1. The bit sequence of m is the prefix to the bit sequence of n and
2. The first bit of the bit sequence of m shifted to the left by the length of the bit sequence of n is 1.

Due to the space limitation, we omit the proof for these propositions. Now, we briefly analyze the reconstructing-ability. To reconstruct an XML label path tree, we need to know the parent-child relation and the order relation between labels. Firstly, from Haveman's code of a node, we can easily know Haveman's code of the parent node by means of Algorithm 1. Based on this algorithm, we can obtain the label path information of the node.

Algorithm 1. Get parent label path number from a label path number
Input: One label path number
Output: Label path number of parent node.

```
n=bits(label path number)      // n is the bits of node number
if (n[Length{n}]=1) then       // odd number
  parent number=n[1..Length(n)]
else                           //even numner
  i=Length(n)
  While (n[i]==0) do i++
  parent number=n[1..i-1]
end else
return parent number
```

2.2 Data Path Number

In this subsection, we will introduce the definition of data path number. As we know, DataGuide is an encoding approach based on the node number and location number, Each DataGuide code is of form (node number, data path). However, the node number of DataGuide does not contain the path information. Furthermore, ancestor-descendant relation is difficult to identify. Since we want to identify a node by the data path and label path information, we do not need the node number here.

We use a binary number to represent a position in a data path. As a result, the data path can be represented by a binary number sequence. **Fig 4** shows a data path number.

Based on this representation, we formally define the concept of data path number.

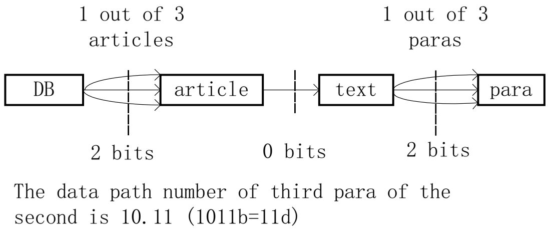

1 out of 3
articles
1 out of 3
paras
DB
article
text
para
2 bits
0 bits
2 bits
The data path number of third para of the
second is 10.11 (1011b=11d)

Fig. 4. One Data Path Number

Definition 5 (Data Path Number).
Let n be a node in an XML document, and bit sequence *S* be the data path of node n. The data path number of n is defined as the decimal number to which the bit sequence is transformed.

To reduce the length of data path, a node is eliminated if the arity of the node is 1. In **Fig. 4**, considering that the data path of the third para. of the second article is 0.10.0.11, the node root and text are eliminated since their arity is 1. The resulting data path becomes 10.11 and the corresponding decimal number is 11. The length of the bit string in encoding a node can be determined by the arity of the node, which is shown in **Table 1**.

An important and extensive application for data path number query is connectedness checking, i.e. given two data path numbers and a node position, we need to check whether or not these two data paths are connected at this position. Here in the following is an algorithm to check connectedness.

Algorithm 2. Connectedness checking algorithm (CC)
Input: Two data path numbers, two arity lists and a position number
Output: true or false.

```
bn1=bit sequence (LPNum1)
bn2=bit sequence (LPNum2)
decompose (bn1,Aritiy list1,positions1[])
decompose(bn2,arity list2, positions2[])
return (positions1[position]==positions2[position])
```

In this algorithm, there is a function, named decompose, which has three parameters. This function decomposes the bit sequence of a data path number into a position list according to arities supplied by user.

To identify a node in an XML document, we need two types of information, one is the structure information of the node, and the other is the data instance of the node. We combine the two types of information to identify a node in an XML document. A node n in an XML document can be uniquely represented by (*ln*, *dn*), where *ln* and *dn* are the label path number and data path number of n respectively.

On the one hand, from a label path number, we get the path information so that the structure information is obtained through all these path numbers. On the other hand, from a data path number, the node position can be located. Hence combination of the label path number and data path number is sufficient for the identification of a node. As a summation, we formally present the following Proposition.

Proposition 4. A node n in an XML document can be uniquely represented by (*ln*, *dn*), where *ln* and *dn* are the label path number and data path number of n respectively.

In **Fig 1**, node DB has three nodes article and each node has one and only one node text, the node text has three nodes para. In order to encode the article, we need 2 bits. The maximum number of the node para is 3, and thus 2 bits can represent the data path of the node para.

Table 1 shows the data path and data path number represented by binary numbers of nodes. From **Table 1**, we know that a node identification is represented by a pair of number (label path number, data path number). We can use the node identification to fully represent an XML document. Based on this representation, we create a HiD index for an XML document.

Towards the end of this section, we briefly explain the construction process of node identification. To create a DataGuid, the involved XML document is parsed twice. It consumes a lot of time, and thus the parsing process should be decreased by just once. During the parsing, we collect all information related to label paths, data paths and the number of nodes. Then the number of bits required for each node is determined, and thus data paths are derived. The next step is to generate label path numbers and data path numbers based on the results obtained in the first step. Here we omit the detailed algorithm because of the space limitation.

3 Index Structure

The index structure in this paper is hybrid index structure, named HiD, and is composed of structure index and value index. The structure index is for document structure and the value index is for node values. The structure index is created based on node identification.

Usually, in a large XML document, there are few label path numbers and a lot of data path numbers. So we firstly construct a Hash table for labels. A label could have more than one label path and label path number. For a complex XML schema, we create an ordered list for the label path numbers with the same label. For data path numbers of a label path number, we construct a B+Tree to index them. The following is a framework of the structure index.

In **Fig. 5**, the index framework contains 3 layers, the first layer is a Hash table, the second layer contains ordered lists, and the last layer contains B+trees. With the Hash table, a label is mapped into an address pointer. Each address pointer points to an ordered list to store all label path numbers of the label. The key of the ordered list is the label path number. Using data path numbers as keys, we create B+trees, which are for data path numbers. To handle some traversing problems rapidly, we build a bi-direction list for leaf nodes of each B+Tree. The list will be called leaf node list.

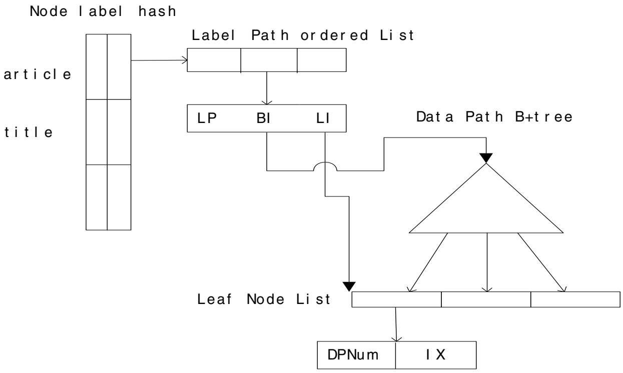


Fig. 5. Structure index

Each element of the label path ordered list is composed of label path number (LP), B+Tree pointer (BI) that points to the root of the B+Tree and leaf nodes list pointer (LI) that points to the leaf node list of the B+Tree. Each leaf node of B+Tree consists of data path number (DPNum) and value index (IX), where IX is the virtual ID of the node in the value index. It is used to connect structure index with value index.

This structure efficiently utilizes different structures in different layers. By Hash table, we can directly access all labels and the corresponding label path numbers. For label paths of a label are few, we create an ordered list rather than a tree structure so that efficiency and space occupied can be balanced. In contrast, the number of data paths of a label is generally great, especially in a large XML document, say, DBLP and XMark. We develop a B+tree for each label path. Since we use data path numbers as keys, we can efficiently handle queries based on position and range, etc. Furthermore, the leaf node list can efficiently support traversing all data paths.

For an XML document, the value of a node is represented by a string. With the development of XML schema, more and more data types are supported in XML documents. Now we propose two types of value indices: *Invert List* and *Number*. We implement a uniform search function: ValueSearch(VID, predication), where VID is a pointer that points to a group of value indices, and predication is a value predication. The function returns a list of elements that satisfy the value predication.

Besides ValueSearch function, we implement another function, named VIValidate, with two input parameters IX and VID. IX is the virtual ID of a node. This function returns the value to that IX points. This function can efficiently support queries, in which data path numbers need to be computed so that IXs are obtained. Since queries contain value predications, values need to be validated.

4 Query Handling

Our intuition is that it is necessary to reduce and eliminate expensive join operations to accelerate query process. Noticing that the numbering scheme contains path information, it is possible to compute label path numbers of related nodes to get the query results. By doing so, join operations can be significantly reduced. We will show how to reduce or even avoid join operations in the coming sub-sections.

Usually, there are two kinds of queries, path pattern query and tree pattern query. For example, query 1: // Document // Author // "Smith", which returns all occurrences of the keyword "Smith" under the path // Document // Author, is a simple path pattern query. Query 2: // Document [./ Abstract//"XML"]// Author [./ Address/Country"USA"] is a tree pattern query. Tree pattern query is more complex than path pattern query. We firstly discuss path pattern query and how to handle wildcard // and *in Section 5.1 and tree pattern query in Section 5.2.

4.1 Path Pattern Query

Path pattern query is the simplest type of queries. Meanwhile it is the base of tree pattern query. We divide path pattern queries into two categories.

The first category is the simple path pattern query that does not contain any node position. Firstly, assume that the query does not contain wildcard."//" and "*". For example, /proceedings [./author="John"] is a simple path pattern query without wildcard "//" and "*". This class of query can be simply handled. Find the label path numbers of leaf node "author" by means of Hash table. Because a label path number is directly related to a rooted label path, based on Algorithm 1, we can follow a bottom-up checking of parent-child relation to validate whether or not the parent node is "proceedings". It is easy to discover all label path numbers that match the given query. Then we avoid expensive join operations produced by structural join algorithms. After getting the true label path numbers, we can return the query results by traversing the leaf node list if the query does not contain a value predication. In our example, there is a predication "author=``John"". We invoke the value index search function Valuesearch in Section 4.2 by which we can get the value index and the data path number, and then the query result can be obtained.

The second category is the path pattern query based on node positions. For example, /Authors/Author [1] /AuthorInfo[2]/ AuthorURL is a path pattern query based on node positions, which requires returning the authorURL of the second AuthorInfo that belongs to the first author. This class of query cannot be handled by the previous region-based numbering approaches. We can efficiently handle this type of queries by node numbering.

Algorithm 3 shows the query process for the path pattern query based on positions. In Algorithm 3, two parameters are input, which are a path pattern query represented by an XPath expression and a position list. The function check is to check whether or not the given label path number match the list of labels of the given query. The function computeDP returns a data path number by the given position list and the corresponding arities list. The function BtreeSearch is to search the data path number on the B+tree.

Towards the end of this section, we briefly discuss how to handle the wildcard "//" and "*". We discuss the two cases: a//b and a*b, and others can be similarly dealt with. a//b means that a is an ancestor of b. Firstly obtain the label path number n and m of a and b respectively. By Theorem 2, we can judge whether or not n is an ancestor label path number of m. Then we can find all the label path numbers of b that satisfy a//b. As for a*b, the handling process is as follows. Firstly get the label path numbers of b. Next, compute label path numbers of the parent node by Algorithm 1. Finally, find this number in the set of label path numbers of a to validate a*b.

Algorithm 3. Algorithm for path pattern query based on position
Input: A path pattern query and position list.
Output: The target element list, represented by (LPNum, DPNum ,IX).

```
n=hash(leaf node)
Fetch each label path number in the label path ordered list pointed by n.pointer
    check(label path number, list of labels)
    GetArity(label path number)
DPNum=computeDP(position list, arities list)
BtreeSearch(DPNum, BI)
Return   (LPNum, DPNum ,IX)
```

4.2 Tree Tattern Query

The existing structural join approaches to handle tree pattern query break up a complete tree pattern query into simple paths and then merge all these simple query results. Our method is to get one branch label path number and, based on branching point and related information, compute other branch label path number and then get final query results. Let's take two examples to explain the algorithm.

Example 1. Query 1: /Article [./title/text][./year] is a simple tree pattern query. This query has two branches, one is /Article/title/text, the other is /Article/year. The branch point is node Article. First we utilize path pattern query algorithm to get two label path numbers LPNum$_1$ of node text and LPNum$_2$ of node year (if the query include wildcard "//" and "*", then there maybe two list rather than two single numbers). Meanwhile we also can get the corresponding data path number list [DPNum$_1$] and [DPNum$_2$] respectively. For these two lists, we implement the connectedness checking of each data path number pair by algorithm 2, and output these pair satisfying connectedness.

Example 2. Query 2: /Article [./title/text="XML"] [./year=2003] is a tree pattern query with a value predication. The main difference between Query 1 and Query 2 is that the latter has two value predications while the former has no predications. So we mainly focus on how to handle value predications with Query 2. After we get the LPNum$_1$ and LPNum$_2$ of the node "text" and the node "year" respectively. The following step is to search the data path numbers that satisfy value predications. We firstly search data path numbers pointed by pointer BI of LPNum$_1$. After we get a list of [DPNum$_1$], the numbers in the list are fewer than those in the list [DPNum$_1$] produced by Query 1. Then instead of searching all the numbers in [DPNum$_2$], we compute a range of data path numbers for the branch /Article/year. Because we can obtain positions of the node "Article" from data path numbers of the node text, so the prefix to data path number of node “year” can be obtained too, and thus the range is [DPNum$_2$, DPNum$_2$’]. In this range, we use the value index to check which numbers are valid. At last, we output results.

Now, we give the tree pattern querying algorithm, algorithm 4 can handle tree pattern queries with two branches. As for more complex tree pattern queries, such as the query with more than 3 branches, we can extend this algorithm to support them.

Algorithm 4. Algorithm for tree pattern query
Input: A tree pattern query Q.
Output: The target element list, represented by (LPNum, DPNum ,IX).

```
1.If (Q has not value prediction) then
2.    Get LPNum$_1$
3.    Get LPNum$_2$
4.    Get [DPNum$_1$]
5.    Get [DPNum$_2$]
6.    fetch each (DPNum$_1$, DPNum$_2$) from [DPNum$_1$] and [DPNum$_2$]
7.        If CC(DPNum$_1$, DPNum$_2$) then output
8. else
9.    Select path with value predication
10.   Get LPNum$_1$
11.   Get LPNum$_2$
12.   Get [DPNum$_1$]
13.   Compute [DPNum$_2$, DPNum$_2$’]
14.   Fetch each (DPNum$_1$, DPNum$_2$) from [DPNum$_1$] and [DPNum$_2$]
15.       If VIValidate (DPNum$_2$) then output
16.end if
```

In Algorithm4, with Row 2 and 3, we get the label path numbers by Hash table. The following step is to get a list of data path numbers. Because the query has no value indices, all the elements in the list are likely valid. Two lists are obtained after Row 4 and 5 are implemented. The last job is to merge the two lists. At this point, most of structural merging algorithms can be adopted. However because a data path number encodes the information of a rooted data path, we choose the semi-join algorithm to accelerate the merging process. The details of the algorithm are beyond our paper and are omitted here.

From Row 9 to Row 15, we handle the tree pattern query with value predictions. In Row 9, a branch is selected from the query, and the selecting process is as follows:

- If only one of two branches has value prediction, then this branch is selected;
- If both of two branches have value prediction, and then the longer path is selected.

Now we have label path numbers of leaf node of this branch. The coming row 12 is different from Row 4. After executing Row 4, all data path numbers pointed by LPNum1 will be returned whereas only those numbers satisfying the value predications are returned after executing Row 12. This can avoid a lot of unnecessary path join operations. Furthermore, unlike the previous algorithm, we can compute data path numbers of the leaf node of other branch instead of joining all data path numbers pointer by LPNum$_2$. The principle of computing data path number is that if two data path numbers are connected by the branch point, then two data path numbers must possess the same prefix which the data path number of the node of the branch point. If both of two branches have value predications, then we need another value index checking process, which is the main job of the last step from Row 14 and Row 15.

5 Experiments

We implemented query algorithm in C++. The XML parser is the Xerces SAX2 parser [26]. The B+Tree API is provided by the Berkeley DB library [19]. Experiments were run on a P4 1.8 GHz CPU PC with 256M main memories, running Windows 2000 Server. We also implemented a node index method similar to XISS [5], but with TwigStack [24] query algorithm on this node index. In addition, we ran ViST [13] on this PC machine for the purpose of comparison.

The data sets in experiments have public XML databases DBLP [20] and the XML benchmark database XMARK [21]. DBLP is a popular computer science bibliography database and is widely used in benchmarking XML index methods. Unlike DBLP, an XMARK dataset is a single record with a very large and complicated tree structure. **Table 2** shows characteristics of the experiment data sets.

Table 2. Characteristics of the experiment data sets

Document	Nodes	Characters	Spaces	Size (MB)
DBLP	1906219	11660704	61485	46.6
DBLP	5920583	95266119	5384135	197
DBLP	6391621	103717843	5817551	209
XMark	2048193	81286567	0	117
XMark	9621573	398304178	0	500

Table 3. Tested Queries

Queries	Data Sets
Q1: /inproceedings/title	DBLP
Q2: /*/author="David"	DBLP
Q3://item/description/keyword="attries"	XMARK
Q4:/site/person/city="New Work"	XMARK
Q5:/ariticle[./author="David"][./year=1996]	DBLP
Q6:/proceedings[./title="XML"][.//author="David"]	DBLP
Q7:/site/item[./location="USA"][./mail]	XMARK
Q8://closed auction[./seller/person="person1"][./date="12/12/2002"]	XMARK

Table 3 lists 8 queries that are tested in our experiments on DBLP data sets and XMARK data sets. These queries are significantly different in terms of complexity, presence of values and structure. We choose four of these queries to run on DBLP data set with size 197MB. Other 4 queries are run on XMARK data set with size 117MB. Because DataGuide cannot answer tree pattern query directly, we do not run query5-query8 by DataGuide.

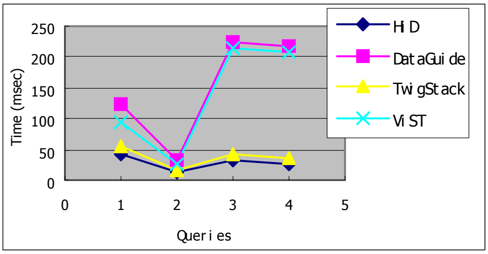


Fig. 6. Elapsed time for Query1-4

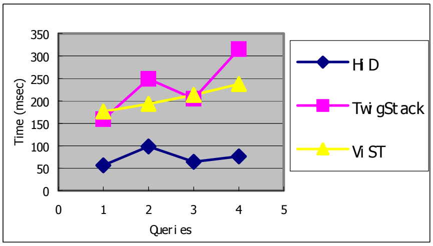


Fig. 7. Elapsed time for Query5 –8

Fig. 6 shows the elapsed time for query 1 to query 4 running by different approaches. From this figure, we know that HiD and TwigStack yield comparable performance and ViST outperforms DataGuide. The most significant performance difference is between query 3 and query 4. DataGuide requires expensive join operations between ordered lists in processing value constraints. So it costs much time. Although TwigStack need join operations too, it skips many unnecessary nodes in the ordered list, and thus it performs better than DataGuide. Moreover, ViST uses top-down transformation of a query that result in a large number of nodes in the virtual trie being examined during subsequence matching. For path pattern query, most of time for our approach is spent on merging output lists. At this point, TwigStack costs as much time as HiD. Unlike DataGuide and ViST, HiD and TwigStack need not much extra time although XMARK data sets have complex structure, which indicates that HiD and TwigStack can efficiently query complex data sets.

The performances of query 5 to query 8 are shown in **Fig. 7**. It is obvious that HiD performs much better than TwigStack and ViST for all tree pattern queries. For complex queries, we can find necessary nodes by computing label path and data path numbers. In this way, we reduce many unnecessary nodes. On the other hand, neither of TwigStack and ViST can efficiently avoid many nodes join operations for tree pattern query. Query 6 and query 8 have wildcard “//”, and ViST performs better than TwigStack, which indicates that ViST handles wildcard more efficiently.

6 Related Work

Node numbering scheme is the foundation for an efficient index structure. There are a lot of researches on this subject. In [1], region based numbering scheme is firstly introduced. In [4,6,10,23] researchers use this type of numbering schemes. Later, the extended preorder numbering scheme [5] is put forward which improves the region based numbering scheme. However it is still a kind of preorder traverse numbering scheme. The results in [8,5,18] have demonstrated that assigning a start number, end number, and level to each element suffices. Each element in an XML document is uniquely identified by its start number and the ID of the document containing the element. In essence, they are still region based numbering schemes.

Because the region based numbering scheme does not support dynamic node inserting or deleting operation, some researchers tried to use statistic technologies to solve this problem. This type of methods [13,17] is based on estimations of the number of attribute values, and other statistical information of the XML document.

To our knowledge, the most efficiently structural join algorithm is the twig join algorithm, including those in [5,8,9,24,25]. These stack-based approaches process the input streams of nodes whose tag appears in the query twig and they speed up join processing by skipping some nodes. The problem with these approaches is that the effectiveness of skipping data depends on the distribution of the matches in the input list.

The DataGuide is proposed in [2, 11]. In the DataGuide, a number is assigned to each node; these node numbers uniquely identify the rooted label path, which is the advantage of the method. However DataGuide cannot answer the query with branching path expressions without accessing the original XML data. In [12], based on DataGuide, a new node identification is proposed.

Another query technology, sequence matching that transforms documents and queries into structured encoded sequences and evaluate queries based on the sequence matching, is recently proposed in [13, 22]. These approaches support flexible queries in query without join operations. But they still have some drawbacks. Firstly, they need a lot of post-processing to guarantee the result accuracy; otherwise they may have false alarms in the query results. The second drawback is that usually these approaches need a lot of storage spaces. Although they eliminate expensive join operations, they need more IO exchange and may lead to declining performance as a consequence.

There are some other researchers concentrating on node coding and index structure. For example, a tagged perfect binary tree is employed to represent an XML document and pre-order traverse binary tree to get the node identification in [7].

7 Conclusions and Future Work

In this paper, we present a new efficient node identification approach and construct an index structure based on this kind of identification called HiD. We also discuss some properties related to our method. For the path pattern query, our querying algorithm avoids join operations. Unlike structural join approaches, the approach handles the tree pattern queries by merging resulting lists of the involved branches. We provide some experimental results to demonstrate the efficiency of our approach.

Actually, the node identification can be used not only to construct HiD index structure and query approach, but also to construct other query approaches such as the sequence matching approach. We are going to develop a sequence matching approach based on our node identification. We also would like to optimize our query algorithms and analyze the time and space complexity in the future.

Acknowledgement

We thank Dr. Haixun Wang in IBM Thomas J. Watson Research Center, for providing the ViST system.

References

[1] Paul F. Dietz. Maintaining order in a linked list. In Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing, pages 122-127, San Francisco, California, 5-7 May 1982.

[2] R.Sacks-Davis, T.Dao, J.A. Thom, J.Zobel. Indexing Documents for Queries on Structure, Content and Attributes. Proc. of International Symposium on Digital Media Information Base (DMIB), Nara, Japan, pages 236–245, 1997.

[3] C. L. A. Clarke, G. V. Cormack, and F. J. Burkowski. An algebra for structured text search and a framework for its implementation. The Computer Journal, 38(1), pages 43-56 1995.

[4] D. D. Kha, M. Yoshikawa, S. Uemura. An XML Indexing Structure with Relative Region Coordinate.In Proceedings of the 17th ICDE, pages 313-320. Heidelberg, Germany, April, 2001.

[5] Q. Li and B. Moon. Indexing and querying XML data for regular path expressions. In Proceedings of the 27th VLDB, pages 361-370. Roma, Italy, September 2001.

[6] C. Zhang, J. F. Naughton, D. J. DeWitt, Q. Luo, and G. M. Lohman. On supporting containment queries in relational database management systems. In Proceedings of the 27th ACM SIGMOD, pages 425-436. Santa Barbara, California, USA, May 2001.

[7] W. Wang. H. Jiang, H. Lu and J. X. Yu. PbiTree Coding and Efficient Processing of Containment Join. In Proceedings of 19th ICDE, pages 391-402. 2003.

[8] Al-Khalifa et al. Structural Joins: A Primitive for Efficient XML Query Pattern Matching. In Proc. of ICDE, San Jose, Feb. 2002.

[9] S.-Y. Chien, Z. Vagena, D. Zhang, V. J. Tsotras, and C. Zaniolo. Efficient structural joins on indexed XML documents. In Proceedings of the 28th VLDB Conference, Hong Kog, China, August 2002.

[10] Alan Halverson, Josef Burger, etc. Mixed Mode XML Query Processing. In Proceedings of the 29th VLDB, pages 361-370. Berlin, Germany, 2003.

[11] Roy Goldman, Jennifer Widom. DataGuides: Enabling Query Formulation and Optimization in Semistructured Databases. In Proceedings of the 23rd VLDB Conference Athens, Greece, 1997

[12] Jan Marco Bremer and Michael Gertz. An Efficient XML Node Identification and Indexing Scheme. Teach report. Department of Computer Science University of California, Davis. Jan.27 2003.

[13] Haixun Wang 1Sanghyun Park Wei Fan Philip S. Yu. ViST: A Dynamic Index Method for Querying XML Data by Tree Structures. In SIGMOD 2003.

[14] D. Chamberlin, D. Florescu, J. Robie, J. Simon, and M. Stefanescu. XQuery: A query language for XML W3C working draft. Technical Report WD-xquery-20010215, World Wide Web Consortium, 2001.

[15] D. Chamberlin, J. Robie, and D. Florescu. Quilt: An XML query language for heterogeneous data sources. In WebDB, May 2000.

[16] J. Clark and S. DeRose. XML path language (XPath) version 1.0 w3c recommendation. Technical Report REC-xpath-19991116, World Wide Web Consortium, 1999.

[17] Edith Cohen, Haim Kaplan, and Tova Milo. Labeling dynamic XML trees. In PODS, pages 271-281, 2002.

[18] Zhang et al. On Supporting Containment Queries in Relational Database Management Systems, SIGMOD Conference, 2001.

[19] Sleepycat Software, http://www.sleepycat.com. The Berkeley Database (Berkeley DB).

[20] Michael Ley. DBLP database web site. http://www.informatik.uni-trier.de/ ley/db.

[21] XMARK: The XML-benchmark project. http://monetdb.cwi.nl/ xml.

[22] Praveen Rao and Bongki Moon PRIX: Indexing And Querying XML Using Prufer Sequences. In ICDE'2004 March 2004.

[23] H.Jiang, H.Lu, W.Wang and B.C.Ooi. XR-Tree:indexing XML Data for Efficent Structural Joins. In ICDE, 2003.

[24] N.Bruno, N.Koudas, D.Srivastava. Holistic Twig Joins: Optimal XML Pattern Matching. In SIGMOD 2002.

[25] H.Jiang, W.Wang, H.Lu. Holistic Twig Joins on Indexed XML Documents. In VLDB 2003.

[26] SAX (Simple API for XML). http://sax.sourceforge.net.

An Interface for Web Based Access to Dynamic Contents

Subhash Bhalla[1], Masaki Hasegawa[1], Enrique Lopez de Lara Gutierrez[1],
Nadia Berthouze[1] and Tomoko Izumita[2]

[1] Graduate School of Computer Science and Engineering,
University of Aizu, Aizu-wakamatsu,
Fukushima 965-8580, (Japan)
{bhalla, d8041201, m5082201, nadia}@u-aizu.ac.jp
[2] Center for Cultural Research Studies,
University of Aizu, Aizu-wakamatsu,
Fukushima 965-8580, (Japan)
izumita@u-aizu.ac.jp

Abstract. Many research efforts aim to provide access to database contents through the medium of the web. At the human end, many web users have little or no database query language skills. Most users are highly skilled at referring to tabular data, as in an Airline time plan. Natural inclination exists among the users for object-by-object traction to find information. For example, for browsing through a book/handbook - users tend to locate objects and path, by following an item-wise (step-by-step) link approach. We propose such an approach based on object based traction (calculations). It is termed as Object-by-Object Calculate (OBOC) approach. Many web users tend to reject the available SQL type or form based black-box approaches.

Our aim is to equip the user with object level query language tools. These facilitate queries and forming queries an easy task at the human end (for unskilled users as well as programmers). The proposed interface introduces simplicity and avoids communication ambiguities. It provides a human-level interface between the user and query languages, such as the SQL. As an intermediate level, the OBOC approach, proposes to support simple Object-to-Object operations with closure property. We compare the proposed approach with the traditional query interfaces such as SQL or QBE. It is an efficient alternative as a high level language interface for web users.

1 Introduction

We propose to address a number of problems that are faced by users while accessing database contents on the web [1], [2], [3]. These are -

- Users are not skilled at using query languages, such as XQuery or SQL.
- Query languages support a complex structure as a statement.
- Web users have difficulty to verify a computation or its outcome.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 70–77, 2005.

- Existing query interfaces do not attempt to take advantage of skills that most users posses [5], [8].
- Web based user interfaces provide inadequate support to express a query. For example, consider the query -

  Find Names of all patients, who were examined by doctors who did a check-up of patient 'James'

  A user may have a fuzzy idea, as one of the doctors who did a checkup of James. In such cases, user rely on Information Requirement Ellicitation' (IRE) [6], [7]. The web user is likely to perform this query in steps, by seeing the names of doctors and selecting one that matches with a sketch in his memory.

We propose an additional alternative set of interactions based on calculator oriented approach using an object-by-object query.

2 Background

2.1 Enquiry by Hierarchical Link for Objects

Existing web-based information systems (WEB-IS) adopt a 'page and link' approach for access to data resources. Users select a related link as per their need. The use of such a hierarchical navigation helps the users in their search for the required piece of information. For example, many e-commerce sites use hierarchical navigation for items on the shopping lists. For example,

Computer Hardware → Parts → Hard Disk Drives → Vendors.

The complexity of hierarchical navigation increases with increase in volume of data and information contents. The approach requires an extensive enumeration of all possible query paths. It is not easy to combine multiple search criteria by using this approach [9].

2.2 Enquiry by Networked Links for Objects

In view of the above difficulties, many WEB-IS provide 'input form and search' approach. The users are permitted to input selected key-words. Based on the inputs, the system attempts to locate the information contents for meeting the users' needs. The 'input form' approach helps to reduce the search difficulties that are posed by hierarchical navigation. But, approach is suitable for restricted domains of information access as in a library or for ticket reservation. With complex queries expressed over large sized databases, the approach tends to have implementation difficulties.

2.3 Enquiry by Object-by-Object Calculate

For accessing the dynamic contents through the medium of the web, the use of database query languages poses a new problem, as most web users are not skilled at the use of database query languages. Existing users at the non-programmer's level use application related forms (Figure 1).

We propose Enquiry by Object-by-Object Calculate. To illustrate the approach, we show by an example, that queries can be expressed in conventional database query languages or by using the proposed approach. The proposed approach introduces an easier to use query interface that can express queries with more steps but with less complications of syntax and expression errors.

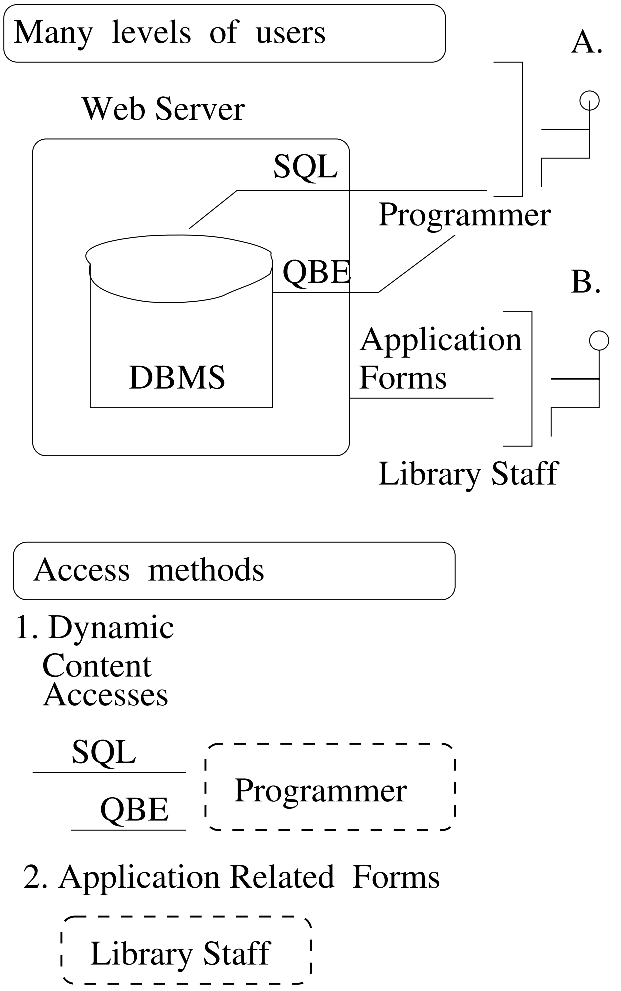


Fig. 1. Web Users and access methods

3 Example Query

Consider a database system with given relations, as -

- Doctor = (DOC-NO, DOCTOR-NAME, SPECIALTY)
- Patient = (PAT-ID,PATIENT-NAME,PHONE-NO, DATE_OF_BIRTH)
- Check-up = (DOC-NO, PAT-ID, DATE, TYPE, ILLNESS, FEE)

Consider an example,
Query

> Find Names of all patients, who were examined
> by doctors who did a check-up of patient 'James'

Such a query, when expressed in database programming languages, such as, SQL (XQuery) or QBE requires complex steps, as shown -

SQL

```
SELECT  P2.Patient-Name
FROM Check-up AS C1, Patient AS P1,
         Check-up AS C2, Patient AS P2
WHERE P1.Patient-Name = 'James'
         AND  P1.Pat-ID =C1.Pat-Id
         AND   C1.Doc-No = C2.Doc-No
         AND   C2.Pat-Id = P2.Pat-Id
```

QBE

Patient	Pat-Id	Patient-Name	Phone-No
	_y	'James'	

Check	Doc	Pat	Date	Type	Ill-	Fee
-up	-No	-Id			ness	
	_x	_y				

Check	Doc	Pat	Date	Type	Ill-	Fee
-up	-No	-Id			ness	
	_x	_z				

Patient	Pat-Id	Patient-Name	Phone-No
	_z	.P	

In case of the proposed approach. The user aims at finding the response to the above query using natural (multiple) steps, as -

a) Names of doctors who did check-up of patient JAMES,
b) Names of patients, examined by these doctors (in part 'a)').

An intermediate user interface may need to support query language with step-by-step closure property to support users calculations. It must allow the user to perform the following steps with ease.

```
1. Choose object.
2. Enquire details.
3. Choose second/next object.
4. Choose an operation on these objects
5. Use closure property to continue at step 3.
```

Using OBOC, the user performs part a) as -

```
OBOC :  Choose Object 1 : Doctor-name
                        : <table 'Doctor'>

        Choose Object 2 : Patient-name;
                          Choose Granularity
                  - 'James' : <'James'>

 Choose operation: [ Join (Doctor - Patient) ]
 - Algebra Operations (on the tabular sets)

       Sysstem Response:
                       : <'Harris' , 'Johnson'>
```

To execute the user specified operation, an SQL statement is generated as,

```
SELECT D1.Doctor-Name
FROM Check-up AS C1, Patient AS P1, Doctor AS D1
WHERE P1.Patient-Name = 'James'
      AND  P1.Pat-ID = C1.Pat-Id
      AND   C1.Doc-No = D1.Doc-No
```

The above users interactions are presented using the following screens,

OBOC : Choose Object 1 : Doctor-name
: < table 'Doctor' >
⟶ Choose Granularity or 2nd Object

1. Object: Doctor-name
2. Choose Object 2 : Patient;
⟶ Choose Granularity or operation
- 'James' : < 'James' >

1. Object: Doctor-name
2. Object : Patient < 'James' >
⟶ Choose Operation: join

Subsequent to receiving a response, using the closure property, the web user can carry out the part b) of the query, as shown by the following screens.

1. Object : Doctor < 'Harris' , 'Johnson' >
⟶ Choose Granularity or Object 2 : Patient-name

1. Object : Doctor < Harris, Johnson >
2. Choose Object 2 : Patient-name
⟶ Choose operation : join

To execute the operation, an SQL statement is generated as,

```
SELECT P1.Patient-Name
FROM Check-up AS C1, Patient AS P1, Doctor AS D1
WHERE D1.Doctor-Name = 'Harris'
        OR D1.Doctor-Name = 'Johnson'
        AND  P1.Pat-ID = C1.Pat-Id
        AND   C1.Doc-No = D1.Doc-No
```

In response the system generates the query response.

Mark
Edwards
Bill
Tracy

4 Implementation Considerations

4.1 Ease of Operations

Skilled programmers may make an error while framing the above query in SQL or QBE. The reason is that, users tend to perform the above computations using a single complex statement or query expression. The multiple steps approach to accessing a database is a naturally occuring, user level approach. It is easy for users (skilled and unskilled) to verify the unit steps and their outcome. Thus, in contrast to the earlier approaches, the proposed apprach based on OBOC is a simpler and easier approach for web users.

4.2 Role of Web Interface

The support provided by the web interface software enables the functions, such as -

1. support for user level objects and operations,
2. 'Information Requirement Ellicitation' [6], [7], for progressive formation of a query in case of the web users,
3. sytem support for prompts to include possibilities of choosing related objects and possible operations, with respect to the chosen (first/available) object.
4. support for choosing granularity and enquiry about the attributes of objects,
5. support for relational algebra with closure property for the user objects

The above features enable the system to offer an easy to use query language interface.

4.3 Implementation Stages

Gradually, for skilled users, the database objects can be accessed using relational algebra operations. These are - Union, Intersection, Join, restrict, project, and rename. The proposed technique is a step-by-step approach for unit step calculations offering a relationally complete query language [4], [9]. A universal relation is formed (linking the two object/entities) using the chosen object entities.

For unskilled users, after selection of the objects,

1. a universal relation can be formed (linking the two object/entities)
2. a system prompt is generated using the possibilities of operations that can be performed using the above objects.
3. Application oriented forms and complex operations can be supported to meet the routine processing requirements.

In this way, the user interface supports skilled as well as unskilled users by providing an easy to use interface.

5 Summary and Conclusions

With the advent of the web based information systems, it has become necessary to support a high level language for user interactions. The medium of user interaction must allow the user to express DBMS queries. The proposed system presents a high level language for user interaction for DBMS applications that are supported through the WWW. The step-wise navigation in the proposed language is based on tracking objects and paths logically and is supported by the SQL support provided by a RDBMS. The proposed QBOC approroach is intuitive. It is relationally complete language as it is based on supporting relational algebraic operations. It does not require the users to obtain programming skills prior to accessing the WEB-IS resources. The users can perform table lookup oriented navigation in unit steps, using natural or logical approach.

References

1. S. Bhalla, T. Hosozawa, H. Sasaki, K. Watanabe, T. Yatsunami, Nadia Berthouze and T. Izumita, "A User Interface for a Web-Geographic Information System", 6^{th} World Multi-Conference on Systems, Cybernetics and Informatics, SCI 2002, July 14 - 18, 2002, Orlando, Florida, USA.
2. S. Bhalla, M. Hasegawa, N. Berthouze, A Framework for a High Level User Interface for Accessing Dynamic Contents on the Web", 6th International Conference on Knowledge-Based Intelligent Information and Engineering Systems, Hosted by University of Milan, 16 - 18 September 2002, Crema, Italy.
3. S. Bhalla, M. Hasegawa, "Query-By-Object Interface for Accessing Dynamic Contents on the Web", IEEE Region 10 Technical Conference on Computers, Communications, Control, and Power Engineering (TENCON'02), 28 - 31 October 2002, Beijing, China.
4. D.D. Chamberlain, et al., "SEQUEL 2: A Unified Approach to Data Definition, Manipulation and Control," *IBM Journal of Research and Development*, vol. 20, no. 5, pp. 560-575, Nov. 1976.
5. M. Derthick, J. A. Kolojejchick, and S.F. Roth, "An Interactive Visual Query Environment for Exploring Data," *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST '97)*, ACM Press, October 1997, pp 189-198.
6. Sun, J. Information Requirement Elicitation in M-Commerce - An Interactive Approach to Facilitate Information Search for Mobile Users, *Communications of ACM*, Vol. 46, No. 12, Dec. 2003, pp. 45-47.
7. Sun, J., H.P. In, and K.A. Sukasdadi, A Prototype of Information Requirement Elicitation in M-Commerce, *Proceedings of IEEE International Conference on E-Comerce (CEC 03)*, pp. 53-56, June 2003.
8. A. Labrinidis and N. Roussoppoulos, "Generating Dynamic Content at Database-backed Web Servers: cgi-bin vs mod_perl," *SIGMOD Record*, Vol. 29, No. 1, March 2000.
9. A. Silberschatz, H. Korth, and S. Sudershan, "Database System Concepts," *McGraw-Hill Book Company*, 2002.

HID: An Efficient Path Index for Complex XML Collections with Arbitrary Links

Awny Sayed and Rainer Unland

Institute for Computer Science and Business Information Systems,
University of Duisburg-Essen,
45117 Essen, Germany
{sayed, unlandR}@cs.uni-essen.de
http://dawis2.informatik.uni-essen.de

Abstract. The increasing popularity of XML has generated a lot of interest in query processing over graph-structured data. To support efficient evaluation of path expressions structured indexes have been proposed. However, most variants of structures indexes ignore inter- or intra-document references. They assume a tree-like structure of XML-documents. Extending these indexes to work with large XML graphs and to support intra-or inter-document links requires a lot of computing power for the creation process and a lot of space to store the indexes. Moreover, the efficient evaluation of ancestors-descendants queries over arbitrary graphs with long paths is a severe problem. In this paper, we propose a scalable connection index that is based on the concept of 2-hop covers as introduced by Cohen el al. The proposed algorithm for index creation scales down the original graph size substantially. As a result a directed acyclic graph with a smaller number of nodes and edges will emerge. This reduces the number of computing steps required for building the index. Thus, computing time and space will be reduced as well . The index also permits to efficiently evaluate ancestors-descendants relationships. Moreover, the proposed index has a nice property in comparison to most other work; it is optimized for descendants-or-self queries on arbitrary graphs with link relationships.

Keywords: XML, indexing, reachability queries, descendants-or-self queries.

1 Introduction

XML has become the universal data exchange format for integrating and exchanging data over intranets and the Internet. The problem of efficiently managing and querying XML documents poses interesting challenges on database research. XML documents not only may have a rather complex internal structure with (ID, IDREF or XLink ([24])) link relationships but can as well be connected to other XML documents via links, mainly by XLink and XPointer ([25]) which led to a complex graph-structured data. At this state XML documents are modeled as a very large graph. XML processing often involves navigating this graph hierarchy using regular path expressions and selecting those nodes that satisfy the path condition. A naïve exhaustive

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 78 – 91, 2005.

traversal of the entire XML documents to evaluate path expressions is very expensive, particularly in a large linked XML documents. Moreover, the absence of schema in XML documents has led to the application developers do not have a complete information about the underlying data, in many cases these developers not know if the information needed to answer the current path expression located in single XML document or needs to composed from several XML documents by following XLink and XPointer techniques. *structural summaries* which derived from the data, play an important role in query evaluation, since they make it possible to answer path expressions directly from them instead of traversing the original documents, which have potentially larger size. But still an important types of path expressions namely, *ancestors-descendants* relationship can not be efficiently evaluated using these *structural summaries* techniques of the underlying graph. *structural joins algorithms* are another techniques found in the market to improve the evaluation of path expressions, but like what existed in the relational database world, joins evaluation consumes a large portion of the query evaluation time. In most cases they are only work with tree- structure data and ignoring the link information and can not handle ancestor (descendant)-or-self queries.

Taking links between documents into account will replace the tree-like structure by a *graph-like* structure which may contain *cycles*. XML query languages usually deploy regular path expressions to query data by traversing through the XML document. Cycles in the graph stress path-indexing algorithms. To efficiently evaluate path queries algorithms usually rely on index structures. While a lot of techniques are known that efficiently construct and maintain such index structures for tree-like structures they usually fail to do the same on graph-like documents. Here they require much too much time to construct the index and need a lot of space to store and work with it. As a consequence connection queries like *ancestors(-or-self)* and *descendants(-or-self)* relationships cannot be answered in reasonable time. When taking *ID/IDREF* attributes or *XLink* and *XPointer* constructs into account two problems arise:

1. The structure of an XML document is no longer tree-like but forms a directed graph which may include arbitrary cycles. In fact, due to possible inter document references a number of original XML documents may merge to larger XML document and cycles can exist on the level of this spanning
2. Links generate large sets of connected elements which may have long paths between them. Thus, to efficiently evaluate path expression queries (especially those with wildcards) with descendant-or-self axis ("//" axis) an appropriate index structure is needed.

In this paper a scalable path index is proposed called HID. It can efficiently address these problems and handle path queries with wildcards of type descendant-or-self or ancestors-or-self on complex XML documents with arbitrary (inter- und intra-document) links which is not considered before in the literatures.

The remainder of the paper is structured as follows. In the next section the state-of-the-art will be discussed. Section 3 introduces basic concepts, especially the underlying work of E. Cohen et al. on 2-hop covers. In section 4 the proposed scalable HID connection index will be introduced and discussed and section 5 concludes the paper.

2 State-of-the-Art

Indexing is essentially used to avoid exhaustive traversal of documents for query processing. Many of the proposed index structures focus on the simplified case of *tree-like* (rather than *graph-like*) data and simple path expressions (like, book/author/title ("/" parent-child axis). Queries containing branches have to be resolved into multiple sub-queries, each corresponding to a single branch in the original structure. The result of these sub-queries has then to be combined by expensive join operations to produce the final answer. These approaches are inefficient in handling *ancestors-descendants queries* with wildcard ("//") over arbitrary XML graphs with long paths.

Tree-Centric Nature: We discuss proposals that facilitate numbering and labeling schemes for the efficient evaluation of ancestors-descendants queries. [19] and [20] propose the use of an XML labeling scheme. By comparing labels assigned to the nodes of the XML tree it is possible to determine the relationship between any two nodes. [4] and [21] propose dynamic labeling schemes. Nodes inherit their parent labels as prefix to their own labels. The connectivity queries can simply be determined by examining whether the prefix relationship exists in the labels of the two nodes. [22] proposes a nested loop-join algorithm which requires B+ tree indexes on the input element sets. The proposal of [23] also relies on B+ trees, however, requires additionally that both element sets are sorted. Then those elements are eliminated that do not participate in the join. [4, 14] describe a labeling scheme for XML trees that supports efficient evaluation of ancestor queries as well as efficient insertion of new nodes. [2, 16] present a tree labeling scheme based on a two level partition of the tree.

Graph-Centric Nature: In the meantime also several approaches have been proposed that deal with graph-like XML-documents. For example, the APEX [10] index uses data mining algorithms to summarize paths that appear frequently in the query workload. Instead of keeping all paths starting from the root node, it maintains only paths of length two. Therefore, it performs poorly for *ancestor-descendant* and *descendant-or-self* queries (e.g. "author//Einstein" and "Einstein//" queries).

Many other approaches rely on a structural summary along with a stored mapping from these summary nodes to the data nodes. Such an index is used to evaluate path expressions directly by pruning the search space. DataGuide [8] is restricted to a simple label path and is not useful in complex path queries with several regular expressions. Index Fabric [1] is conceptually similar to the DataGuide in that it keeps all label paths starting from the root element. It encodes each label path to each XML element with a data value as a string and inserts the encoded label path and data value into an efficient index for strings. The index block and XML data are both stored in a relational database systems. Index Fabric losses all parent-child relationships, thus, is not efficient for processing partial matching queries. Similar to Index Fabric, the F+B Index [9] optimizes a set of branching queries. It is based on the Forward and Backward index (F&B index [28]) and suffers from the same problems than Index Fabric. Index schemes like 1-index [6], A(k)-index [12], and D(k)-index [5] are based on the concepts of similarity and bi-similarity of nodes. These indexes are suited for intra-document links (of type ID and IDREF(S)). But they ignore inter-document links (links by XLink and XPointer).

These approaches mostly focus on constructing index structures for paths without wildcard and exhibit poor performance for the ancestors- or descendants-or-self axes. Moreover, not much attention is paid to inter-document links. In our prior work [13] we dealt with this problem by introducing an index based on the concept of 2-hop cover [7]. With this solution the space and the time needed to build an index increases proportionally with the increase of the document sizes. R. Schenkel et al. [11] adapted and improved this idea with their HOPI index. It introduces a divide-and-conquer algorithm for index creation to improve the space and time characteristics of the 2-hop cover algorithm. However, still the size of the HOPI index can grow very large with large document sets. The time to build the HOPI index increases substantially with an increasing number of documents. Moreover, a HOPI index can not deal efficiently with highly linked XML collections like the Movies [26] or XMarch databases [27], especially since the partition problem that is the basis of the divide-and-conquer algorithm is known to be an NP-hard problem.

3 Basic Terms and Definitions

We consider a model of XML data in which we comments, processing instructions, and namespaces. Then, an XML document can be modeled as a directed graph $G_D=(V_{D,} E_D)$. V_D is a set of vertices (or nodes) that represent element names or attribute values. E_D is a set of edges, which indicate an element-sub-element or element-attribute edges. Each node in V_D is assigned a string label and has a unique idenifier The union of all XML document graphs G_1, …, G_n forms a large XML graph G=(V, E, EA, EL). V and E are defined as above. EA is the set of all directed edges where an edge represents the relationship between an element and a value expressed by an XML attribute. EL is the set of directed edges that represents the element-element relationships via IDREF and IDREFS attributes in the schema information (URI, XLink, XPointer).

Each node in the graph represents an element or attribute name or value. An element node is an object that contains additional information. Every element node has a label, the URL of the document, in which it occurs, and its identifier. Fig. 1 depicts an example XML data graph, showing four XML documents about Prof. Einstein (homepage (D1), publications (D3), research interest (D3) and research (D4)).

3.1 2-Hop Covers

The purpose of this paper is to introduce a new index structure that can be exploited by path expressions with wildcards that want to access the ancestors or descendants axes in arbitrary complex graphs that may contain arbitrary cycles. A straightforward approach would be to calculate the transitive closure of all descendants and ancestors for each node of the graph structure. Note, that graph structure here may mean a collection of linked XML-documents.

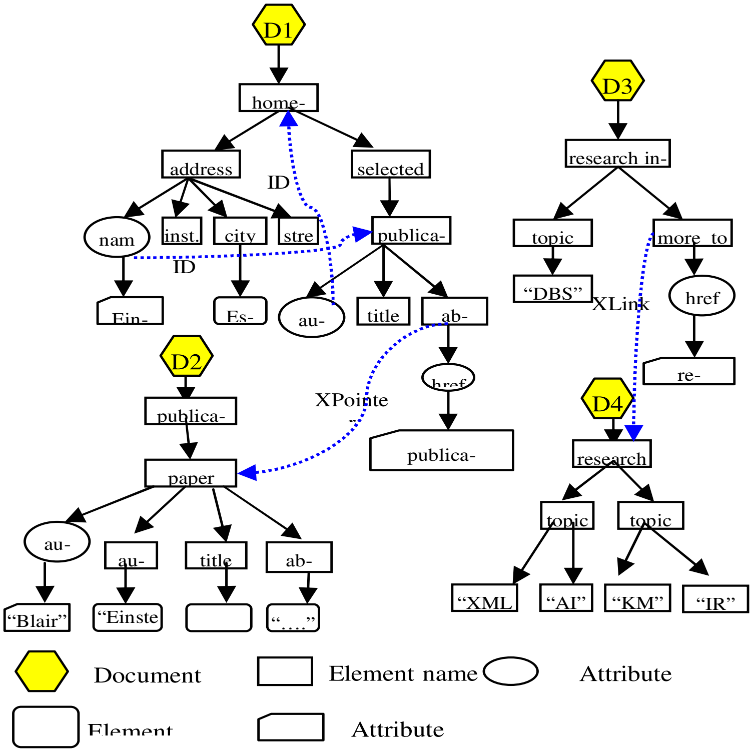


Fig. 1. XML documents graph

Edith Cohen et al. introduced the concept of a 2-hop cover [7]. It calculates and expresses the transitive closure in a much smarter way, thus being an order of magnitude more space-efficient and less time consuming in calculating the index. Instead of storing the two sets of all ancestors and all descendants of a node directly they propose to only store a significant smaller subset of each. Thus, for each node *u* two label sets $L_{IN}(u)$ and $L_{out}(u)$ are maintained. $L_{IN}(u)$ is a set of an arbitrary (however, in the ideal case minimal) number of ancestors. These are nodes that are (indirectly) connected to *u* by at least one (arbitrarily long) path. In the same way, $L_{out}(u)$ defines a set of "relevant" descendants of *u*. These are those nodes that can be reached from *u* by at least one (arbitrarily long) path. The calculation of these sets has to be done in a way that it is guaranteed that if there is a path from node *u* to *v* that this can be derived from the subsets of both nodes. Their idea is based on the concept of a "mediator" node. This is a node that is member of the descendant set of *u* as well as the ancestor set of *v* (in case of a descendant query, otherwise, the other way round), thus, is a node on the path from *u* to *v* (if there is such a path). Thus, when calculating the descendant property between *u* and *v* the descendants subset of u is intersected with the ancestors subset of v. If and only if there is an overlap between both sets (in the ideal case in one node only) a path between both nodes exists ($L_{out}(u) \cap L_{IN}(v) \neq \emptyset$). The name 2-hop cover stems from the fact that with this solution the first step is to get from u to the mediator node m and then, in a second step, from m to the final node v, thus resulting in a 2-hop proceeding. The challenge now is to keep the two subsets of descendants and ancestors as small as possible. Unfortunately, their calculation is an NP-hard problem. Cohen et al. call the mediator node m on a path from u to v the *center node* and add m to the set $L_{out}(u)$ of descendants of u and to the set $L_{IN}(v)$ of ancestors of v.

For example, consider figure 2 with the information of the 2-hop cover (each node has two sets, a set of its ancestors L_{IN} and a set of its descendants L_{out}. The node *abstract* that has the object identifier 6 is considered as the *center node* of the graph (the algorithm that construct the labeling choose abstract node because the center graph of this had the densest sub-graph among all center graphs). There is a connectivity between the two nodes *y=(3, selected pub.)* and *y1=(10, author)* because $L_{OUT}(y)$ intersect $L_{IN}(y1)$=(6) is not empty but there is no reachability between *z=(2, address)* and *y1=(10, author)* because $L_{OUT}(z) \cap L_{IN}(y1) = \emptyset$.

Since the original idea of 2-hop cover is more general, we adapted the definitions to match our purpose, especially by not considering more general concepts that are not needed here in our work.

Definition 1 (2-Hop Reachability Labeling)

A 2-hop reachability labeling for a directed graph G=(V, E) assigns to each vertex (node) $u \in$ V a set $L_{IN}(u)$ of ancestors and a set $L_{OUT}(u)$ of descendants: $L(u)=(L_{IN}(u), L_{OUT}(u))$, with $L_{IN}(u), L_{OUT}(u) \subseteq V$ and there is a path <*x*…..*u* > from each $x \in L_{IN}(u)$ to u and a path <*u*…..*x*> from u to each $x \in L_{OUT}(u)$. The node u is also added to both sets $L_{IN}(u)$ and $L_{OUT}(u)$.

Fig. 2 shows a part of the example scenario graph from Fig. 1 with 2-hop labeling added to each node in the graph.

The main idea of creating our scalable connection index using 2-hop labels is based on the following observation: Let a directed XML graph G=(V, E) be given. For each pair u, $v \in$ V of nodes with 2-hop labels $L(u)=(L_{OUT}(u), L_{IN}(u))$ and $L(v)=(L_{OUT}(v), L_{IN}(v))$ there is a path from u to v in G if there is a node $x \in$ V such that $x \in L_{OUT}(u) \cap L_{IN}(v)$.

As described in Definition 1, 2-hop reachability labeling of a given directed graph G=(V, E) assigns to each node two sets. A 2-hop cover of a directed graph G is a 2-hop labeling that covers all paths of the graph G.

Definition 2 (2-Hop Cover)

Let G=(V, E) be a directed graph with nodes V and edges E. A 2-hop cover is a 2-hop reachability labeling of G such that $L_{out}(u) \cap L_{IN}(v) \neq \emptyset$ iff there is at least one path from node an arbitrary node $u \in L_{IN}(u)$ to an arbitrary node $v \in L_{OUT}(u)$. The overall size of all labels for 2-hop covers is $\Sigma_{u \in V}(|L_{OUT}(u)|+|L_{IN}(u)|)$.

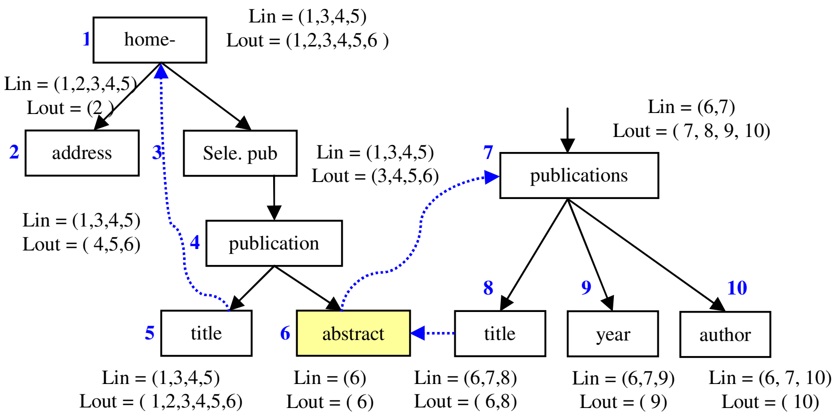


Fig. 2. Part of Fig1 (XML graph) with 2-hop reachability labeling

3.2 Efficient Evaluation of 2-Hop Cover

Given a directed graph G=(V, E), and the *transitive closure* G'=(V, E') as an input parameter which means that E' contains all combination of nodes between which an arbitrary long path exists. To find a minimum 2-hop cover for a directed graph is an NP-hard problem. E. Cohen et al. propose the following algorithm that needs polynomial time to compute the 2-hop cover for the graph G [7]:

Let node $x \in$ V be given. $C_{IN}(x)=\{v \in V|(v, x) \in E'\}$ is the set of all nodes for which there is a path from v to x in G (ancestors of the current node x). In the same way $C_{OUT}(x)=\{v \in V|(x, v) \in E'\}$ defines the set of all descendants of node x, such that there is a path from x to v in G. Let $S_x = S(C_{IN}(x), x, C_{OUT}(x))=\{(u, v) \in E' \mid u \in C_{IN}(x) \wedge v \in C_{OUT}(x)\}=\{(u, v) \in E'|(u, x) \in E' \wedge (x, v) \in E'\}$ denote the set of paths in G that contain x. Moreover, the node x is called the *center node* of S_x. The algorithm that is used to compute the optimal 2-hop cover takes in the beginning the set E' (by copying it to E'') and, step by step, chooses from it relevant connections. By this redundant connections are indirectly eliminated since they are not chosen. During the computation of the necessary connections E'' contains those connections that are not yet covered. In the beginning E'' = E' and the 2-hop labels for each $x \in$ G are empty. Now, node by node, the 2-hop labels are added. The set $S_x \cap E''$ contains all connections within G that contain x and are not yet covered. The relationship between the number of connections of x that are not yet covered and the total number of nodes is described by this ratio:

$$r(x)= \frac{\left|S_x \cap E''\right|}{\left|C_{IN}(x)+C_{OUT}(x)\right|}$$

In each iteration step the algorithm as proposed in [7] selects in a greedy manner the best x; that is the x with the highest value for r(x). This results in a small set of nodes. It covers as many not yet covered connections as possible. Thus, it increases the overall size of 2-hop labeling very little. More precisely for each $u \in C_{IN}(x) \wedge v \in C_{OUT}(x)$ $L_{OUT}(u)$ will be extended by x and for each $v \in C_{OUT}(x)$ $L_{IN}(v)$ will be extended by x. Then the set S_x will be removed from E''. The algorithm terminates when E'' is empty. This happens as soon as all connections in the transitive closure graph G' are covered with 2-hop covers. To compute the 2-hop covers for a given set E' the above algorithm requires exponential time because in each computation step for the algorithm there are an exponential number of subsets $C_{IN}(x)$, $C_{OUT}(x) \subseteq V$. We will now show how it can be improved to only use polynomial time to compute the 2-hop cover. This algorithm works on the basis of the center node concept and finds the densest sub-graph for a center node x.

Definition 3 (Construction of the Center-Graph):
The construction of the *center-graph* is to be performed as follows: Let G=(V, E) be given. E'' $\subseteq$ E' is the set of nodes that are not covered in G. For a node $x \in V$, the *center-graph* $G_x = (V_x, E_x)$ of *x (center node)* is the undirected graph with two node sets $V_{IN}(x)$ and $V_{OUT}(x)$. The vertices $V_x = V_{IN}(x) \cup V_{OUT}(x)$, where $V_{IN}(x) = \{u \mid u \in V: \exists\, v \in V: (u, v) \in E'' \wedge u \in C_{IN}(x) \wedge v \in C_{OUT}(x)\}$ contains all the nodes from which a path to x exists (not only incoming and outgoing edges but also the ancestors of x in G) and $V_{OUT}(x) = \{w \mid w \in V: \exists\, y \in V: (y, w) \in E'' \wedge y \in C_{IN}(x) \wedge w \in C_{OUT}(x)\}$ contains all the descendants of x in G. Fig. 3 shows the center-graph of Fig. 2 with the node *abstract* as *center node*.

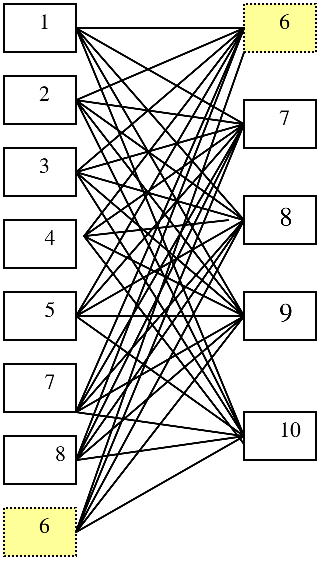


Fig. 3. Center-Graph of node 6 (*abstract)*

Definition 4 (Density of the Center-Graph):
The density of the center graph is defined as the number of edges divided by the number of nodes. In Figure 3 the density of the *center-graph* is 40(edges)/13(nodes)=3,08.

To compute the *densest sub-graphs* from the *center-graph*, Cohen [7] introduced the 2-approximation algorithm that needs linear time. It iteratively removes a vertex of minimum degree from the graph and computes the density of each resulting sub-graph. The result of the algorithm is a set of sub-graphs together with their density. The algorithm returns the sub-graph with the highest density.

Given an undirected *center-graph* $G_x = (V_x, E_x)$. The task is to find a subset $S \subseteq V_x$ for which the average degree in the sub-graph induced by S is maximized, i.e., a set that maximizes the ratio $|E(S)|/|S|$, where E(S) is the set of edges connecting two vertices of S (see Fig. 6). The refined algorithm for computing a 2-hop cover always chooses the best node *x* out of the remaining nodes in descending order of the density of the densest sub-graph. To do so it now needs polynomial time. To compute the transitive closure of a given graph G, E. Cohen uses the Foyd-Warshall algorithm [17]. It needs time $O(|V|^3)$. To compute the 2-hop cover from the transitive closure it needs time complexity $O(|V|^3)$.

Since the input parameter for the 2-hop cover is the pre-computed transitive closure of the original graph this algorithm can be very memory consuming if applied to a very large XML graph (e.g. DBLP with 5 million nodes or XMarch as a highly cyclic database). The HOPI [11] index solves this problem by proposing a divide-and-conquer algorithm that is based on partitioning the original graph. It computes the transitive closure and the 2-hop cover for each part. Then it merges the 2-hop covers of all partitions. The partition problem is an NP-hard problem. Thus, the optimal partition for the large XML graph cannot be computed. The objective of this paper is to introduce a scalable connection index that overcomes the partition graph problem.

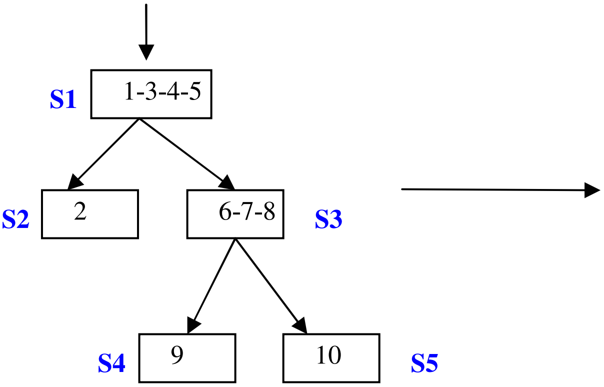
4 HID Path Index

The scalable HID connection index is based on the fact that if we shrink every strongly connected component (which is equivalent to a cycle) of the input XML graph G=(V, E) to a single node (which will be called *super-node*), then the resulting graph is a DAG (Directed Acyclic Graph) with a minimal number of nodes and edges. These strongly connected components are the most likely reason for redundant computations in the transitive closure algorithms since it does not detect and eliminate them. A strongly connected component (SCC) of a graph G is a maximal set of nodes or vertices $C \subseteq V$ such that for every pair of vertices u, $v \in C$ a path from *u* to *v* as well as vice versa (*v* to *u*) exists. The size of SCC is the number of the vertices it contains. Each SCC forms a *super-node*. A vertex which is not contained in any cycle is a SCC of size one and it also a *super-node*. Decomposing a directed graph into its strongly connected components using two depth-first search is described in [17]. For a given arbitrary XML graph G uses the transpose of G, which is also a graph $G^T = (V, E^T)$, where $E^T = \{(u, v): (v, u) \in E\}$. That is, E^T consists of the edges of G with their direction reversed. The time needed to create G^T is O(V+E) [17]. The following linear time algorithm from [17] can be used to compute the strongly connected component of a directed graph G. It uses two depth-first searches (DFS), one on G and one on G^T.

1. Call DFS(G) to compute finishing times f[u] for each vertex u.
2. compute G^T
3. Call DFS(G^T), but in the main loop of DFS, consider the vertices in order of decreasing f[u].
4. Output the vertices of each tree in the depth-first forest formed in line 3 as a separate strongly connected component.

When a node is visited for the first during the DFS it gets a *discovery time* (equivalent to an enumeration in sequential order). It gets a *finishing time* when its adjacency list has been examined completely (for more details see [17], section 22.3).

Example: Consider the XML directed graph (see Figure 2). It consists of ten nodes with labels {1, 2, 3, …, 10}. The resulting DAG from this graph (see Figure 4(a)) consists of five SCC: S1={1, 3, 4, 5}, S2={2}, S3={6, 7, 8}, S4={9}, and S5={10}. Using this technique the number of nodes is reduced to half. To efficiently evaluate the query, a table (see Fig. 4 (b)) is maintained that stores for each node the set to which it belongs. To answer reachability queries, for example *u//v,* it has to be checked first if *u* and *v* belong to the same set. If this is not the case then the HID index has to be used next.



Node	set
1	S1
2	S2
3	S1
4	S1
5	S1
6	S3
7	S3
8	S3
9	S4
10	S5

Fig. 4. (a) the result DAG of Fig. 2 (b) nodes table of Fig. 4 (a)

4.1 Efficient Comutation of the Transitive Closure Using Strongly Connected Components

Since the transitive closure is needed as input parameter for the 2-hop cover algorithm it needs to be materialized and stored in main memory during the construction time of the index. Thus, the question arises how the transitive closure of a linked forest of XML documents can be computed efficiently. The HOPI connection index [11] deals with this problem by proposing the divide-and-conquer algorithm. Since it partitions the original XML graph the transitive closure needs to be materialized for each partition separately. The partition graph problem is known to be NP-hard [11]. Thus it is difficult to find a good solution for a large XML graph in reasonable time. Moreover, two nodes that are connected in the original graph may be disconnected after the partitioning. This affects the efficiency of the query evaluation. Our scalable HID constructs the transitive closure from a directed acyclic graph (DAG) that has fewer edges and vertices than the original graph. Working with a DAG instead of the underlying XML graph has the following benefits:

1. Materializing the transitive closure of the computed DAG (see Fig. 6 (a), has five nodes) instead of the underlying large XML graph (see Fig. 2, has ten nodes) as input parameter to the 2-hop cover computation can avoid the problem of extensive space-consumption and, by this, may make a main memory-based computation of the cover feasible.
2. The reachability queries can be efficiently evaluated in O(1) if the two nodes are located in the same *super-node*.

E. Cohen [7] and R. Schenkel [11] use the Floyed Warshall [17] algorithm to compute the transitive closure for the input XML graph. It needs $O(|V|^3)$ time. In our work we implement the transitive algorithm introduced by Esko Nuutila [18] which is considered to be the best algorithm for the computation of the transitive closure of the directed graph based on the detection of strongly connected components. The main strength of this algorithm is that it scans the input graph only once without generating partial successor sets for each node. This algorithm therefore only needs $O(|V|^2)$ time.

Efficient Computation of the Densest Sub-graph Revisited

If a DAG G is given the 2-hop cover algorithm of E. Cohen requires building the transitive closure of G. For each *super-node x* in the DAG the center graph (see Definition 3) CG(*w*) is to be constructed. The algorithm proceeds in iterations, where the basic operation in each iteration is the selection of the densest sub-graph. The *super-node x* in the DAG, during the 2-hop labeling computation, this super-node is added to the two sets L_{IN} and L_{OUT} as explained in section 3.2. Figure 5 (a) shows the resulting DAG from the original XML graph (figure 2) using the SCC algorithm. It has five nodes instead of ten nodes.

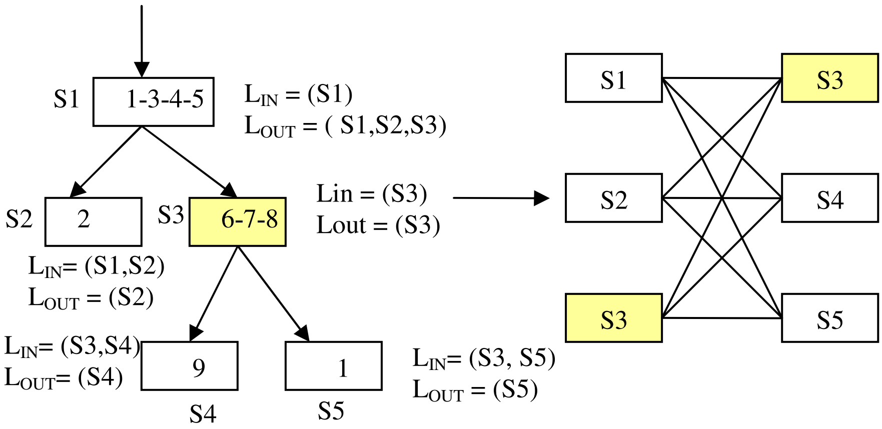


Fig. 5. (a) 2-hop labeling of DAG (b) the center graph of Fig. 5 (a)

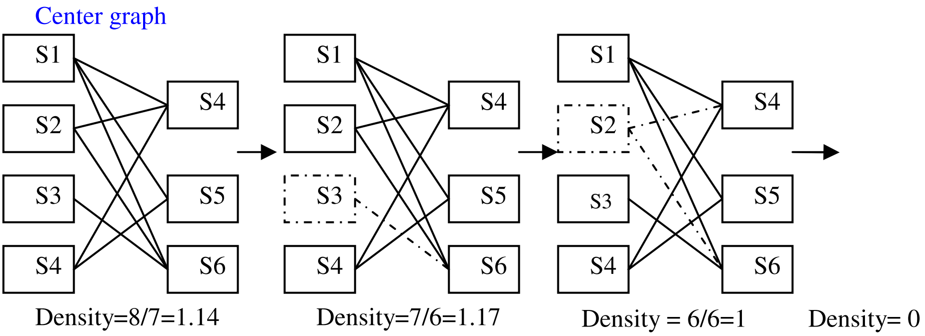


Fig. 6. Densest sub-graph of a given center graph

5 Implementation Study

5.1 Query Evaluation with the HID Approach

The objective of HID approach for indexing large-scale collections of interlinked XML documents is to optimize the required space and to reduce the time needed to evaluate reachability queries. To test the reachability between two nodes using the HID Index is to be done as follows:

1. For any two given nodes *u* and *v*, check the SCC table. If both nodes belong to the same super-node the reachability between *u* and *v* is already proved. A lookup at the table needs linear time. It is not necessary to use the HID index at all.
2. In case that the nodes *u* and *v* do not belong to the same super-node the HID index has to be used. All relevant information is stored in database tables. We will not introduce all tables here since we only want to give a rough overview of the algorithm and the used names are self-explaining. One table assigns the object identifiers to the nodes. Let us assume that these are oid1 and oid2 in our case. Now, the L_{OUT} set from the DESCENDANT table for oid1 and the L_{IN} set from ANCESTOR table for oid2 have to be read. Next, the intersection of the two sets is to be performed. If $L_{OUT}(oid1) \cap L_{IN}(oid2)$ is not empty than there is a reachability between the given two nodes. Otherwise the nodes are not connected.

6 Conclusion

In this paper we presented an efficient and scalable connection index called HID. It is an index structure for efficiently evaluating path queries of type ancestors-descendants or descendants-or-self axis (with wildcard "//") on very large XML data graphs with long paths. We presented an algorithm that allowed to efficiently construct a complete path index for a linked forest of XML-documents. It is based on strongly connected components. Our preliminary experiments with real-life XML data (e.g. movie database as a highly *cyclic* database and Mondial database as a database of highly *linked* documents) show that our scalable HID index can represent connections in highly interlinked XML collections very efficiently. It improves query processing costs and space requirements compared with the previous index structured significantly. Currently we are still performing our experiments in order to get a better understanding of the real performance of our approach and when it makes sense to deploy it. We also intend to compare our approach to the original approach and the HOPI approach. Moreover, we are working on the update problem. Since the construction of the index is quite complex its construction only makes sense if the index can be used for some time. However, this means that we have to deal with the problem of updates of the original XML-documents.

References

[1] Cooper, Brian F.; Sample, Neal; Franklin, Michael J.; Hjaltason; Gísli R.; Shadmon, Moshe: *A fast index for semistructured data*; VLDB 2001, Proceedings of 27th International Conference on Very Large Data Bases, September 11-14, 2001, Roma, Italy. Morgan Kaufmann 2001, ISBN 1-55860-804-4; 2001.

[2] H. Kaplan and T. Milo. *Short and simple lables schemes for small distances and other functions*. In 7th int. Workshop on Algorithms and Data Structures (WADS), pages 246-257 2001.

[3] D. Barashev and al. *Indexing XML to Support Path Expressions*. In 6th East-European Conference on advances in Databases and Infromation System (ADBIS), 2002.

[4] E. Cohen at al. *Labeling dynamic XML trees* . In Symposium on Principle of Databases (POSD), pages 271-281, 2002.

[5] C. Qun and al**.** *D(K)-Index: An adaptive Structural Summares for Graph-based Data*. In ACM SIGMOD Int. Conference on Mangement of Data, pages 134-144, 2003.

[6] T. Milo and D. Suciu. *Index Structures for path expressions*. In 7th International Conference on Database Theory (ICDT). pages 277-295, 1999.

[7] Cohen; Eran Halperin; Harin Kaplan and Uri Zwick: *Reachability and distance queries via 2-hop labels*. Proceedings Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms, pages 937–946. ACM Press, 2002.

[8] R. Goldman and J. Widom. *DataGuides: Enabling query formulation and optimization in semistructured databases*. In M. Jarke, M. J. Carey, K. R. Dittrich, F. H. Lochovsky, P. Loucopoulos, and M. A. Jeusfeld, editors, VLDB'97, Proceedings of 23rd International Conference on Very Large Data Bases, August 25-29, 1997, Athens, Greece, pages 436–445. Morgan Kaufmann, 1997.

[9] R. Kaushik et al. *Covering indexes for Branching path queri*es. In ACM SIGMOD int. Conference on Management of data, Pages 133-144, 2002.

[10] C.-W. Chung, J.-K. Min, and K. Shim: *APEX: An adaptive path index for XML data*. In Franklin et al. [6], pages 121–132.

[11] R. Schenkel et al. *HOPI: An Efficient Connection Index for Complex XML Document Collections*. In 9th Int. Conference on Extending Database Technology (EDBT), pages 237- 255, 2004.

[12] R. Kaushik et al. *Exploiting Local Similarity for Indexing Paths in Graph-Structured Data*. In 18th Int. Conference on Data Engineering (ICDE), 2002.

[13] Sayed and R. Unland. *Index-support on XML documents Containing Links*. IEEE Midwest Symposium on Circuits and System, 2003.

[14] H. Kaplan et al. *A Comparison of labeling schemes for ancestor queries*. In 13th ACM-SIAM Symposium on Discrete algorithms (SODA), Pages 954-963, 2002.

[15] The Mondial Database. http://dbis.informatik.uni-goettingen.de/Mondial/

[16] S. Abiteboul et al. *Compact labeling schemes for ancestor's queries*. In 12th ACM-SIAM Symposium on Discrete algorithms (SODA), Pages 547-556, 2001.

[17] T. H. Cormen et al. *Introduction to algorithms*. 2nd Edition, Chapters 22-23, 2001.

[18] Esko Nuutila and Soisalon-Soininen. *Efficient Transitive Closure Computation*. Technical Report TKO-B113, 1993.

[19] Q.Li and B. Moon**.** *Indexing and querying XML Data for Regular Path Expressions*. In 27th Int. Conference on Very Large Data Bases (VLDB), Pages 361-370, 2001.

[20] M. Yoshikawa and T. Amagasa. XRel: A Path-Index Based Approach to Storage and Retrivel XML Documents Using Relational Databases. In ACM Transactions on Internet Technology (TOIT), 2001.

[21] S.D. Tatarinov and C.Zhang. *Storing and Querying Ordered XML Using a Relational Database System*. In ACM SIGMOD Int. Conference on Management of Data, pages, 204-215 , 2002.

[22] C.Zhang, J.F. Naughton, D.J.DeWitt, Q. Luo, and G. Lohman. *On Supporting Containment Queries in Relational Database Mangement System*. In ACM SIGMOD Int. Conference on Management of Data, 2001.

[23] Shu-Yaho Chein et al. *Efficient Structural Joins on Indexed XML Documents*. In 28th Int. Conference on Very Large Data Bases (VLDB), 2002.

[24] XML Linking Language (XLink) Version 1.0, W3C Recommendation (27 June 2001), see http://www.W3.org/TR/xlink.

[25] XML Pointer Language (XPointer), W3C Working Draft (16 August 2002), see http://www.w3.org/TR/xptr.

[26] The Internet Movie Databse. http://www.imdb.com.

[27] The XML bechmark project. http://www.xml-benchmark.org

[28] S. Abiteboul, P. Bunmen, and D.Suciu. Data on the Web: from relations to semistructured data and XML. Morgan Kaufmann Publishers, Los Atlos, CA 94022, USA, 1999.

Integration of Virtual Reality and Database System Techniques

Elisa Bertino[1], Stefano Franzoni[2], Pietro Mazzoleni[2], and Stefano Valtolina[2]

[1] CS and ECE Departments, Purdue University,
West Lafayette (Indiana), USA
bertino@cs.purdue.edu
[2] DiCO, University of Milano, Milano, Italy
{franzoni, mazzolen, valtolin}@dico.unimi.it

Abstract. In this paper we discuss issues concerning the development of interactive virtual reality (VR) environments. We argue that the integration of such type of environments with database technology has the potential of providing on one side much flexibility and on the other hand of resulting in enhanced interfaces for accessing contents from digital archives. The paper also describes a project dealing with the dissemination of cultural heritage contents. Within the project an integrated framework has been developed that enhances conventional VR environments with database interactions.

1 Introduction

In the last ten years, virtual reality (VR) technologies have been the focus of intense developments, also because of the increased availability and accessibility of dedicated hardware platforms and of fast advances in display technologies. VR techniques have also been successfully applied to different domains, such medicine, architecture, chemistry, education, entertainment. One of the most useful characteristics of a VR-based approach, shared by all its applications, is the possibility it offers to look at the objects of interest from multiple viewpoints, usually not available in normal conditions. For example, an architect can visit the building he/she has designed before the construction; a pilot can be trained through a simulation without actually flying.

Research opportunities related to the VR world are steadily growing in number. Recently, VR has attracted the interest of researchers as a new and powerful tool for enhancing human interactions with databases and digital archives. In particular, the application of such technology is very promising in making the access to digital archives easier and more attractive and the visualization of their contents more effective. Even though current database systems typically provide a large number of different interfaces, ranging from the direct use of SQL language to form-based interfaces and to application-specific interfaces, interacting with a database requires some skills in using computer systems. We are still far from being able to provide users with tools which are natural for them to use. However, the main drawback is that those interfaces are not particular attractive. This is an obstacle to a wider use of database

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 92 – 103, 2005.

systems and digital archives as sources of information for a large variety of users, from children to senior citizens. Moreover also the interpretation of query results can be problematic, especially for the less skilled users. We believe that results should be displayed in terms of presentation objects that are related to the semantics of retrieved information. We need very much tools able to provide the equivalent of the desktop metaphor for database systems and digital archives. A main requirement characterizing database systems and digital archives is however that the metaphor should involve objects semantically related to the actual information contents. Thus, metaphoric presentations should be dynamically generated based on query results.

We believe that VR is an important element in achieving the above goals. It can be applied to create environments that, by hiding the complexity of the underlying databases and archives, are more comprehensible to users and make easier for them to interact with the system. Not only a VR approach can simulate the real world to which the stored information are related, it also offers possibilities to customize contents that are very difficult to achieve in the physical world. For example, if a database contains information concerning a digital library, we can think of developing a virtual library showing the books a user is interested in. In such a case, the user can easily browse the information, because she/he can interact with objects that belong to his/her personal experience and which are characterized by well-known properties and behavior. We can think that a customized library can be built for each user, based on his/her interests, the tasks he/she is involved in, the past access history. We believe that by mapping the information retrieved from a database onto objects of the virtual world according to a well-understood metaphor one can dramatically enhance the process of acquiring new information, especially for the non-expert users.

As we already mentioned, VR techniques can be very useful also in the information visualization process. A virtual environment makes it possible to take advantage of the spatial orientation ability, innate in every human being, to browse the database in a very natural way. Moreover, one can quickly adjust the level of detail with which the information is examined by users by simply changing the point of view in the virtual space. Information visualization is also a powerful instrument for highlighting the relations linking the data. Every three dimensional object is endowed with six degrees of freedom, consisting of the three spatial coordinates and its orientation. These coordinates, combined with the objects' shape and color, can help identifying possible hidden relations and trends correlating the data. Another important property of information visualization through a VR approach is that it allows one to quickly and effectively analyze large amounts of data. This characteristic is very valuable in enhancing traditional data mining techniques. Another very promising application of VR techniques is related to the dissemination of cultural heritage (CH) contents, especially when dealing with ancient manuscripts and artifacts that need to be carefully preserved and cannot thus be made available to the general public or with environments not any longer existing.

In this paper we discuss issues related to dynamic and interactive VR environments and outlines open research issues. We focus on the use of VR and its integration with database system techniques in the framework of an application for the exploration and visualization of cultural heritage contents, developed as part the DHX (Digital Ecological and Artistic Heritage Exchange) project [34]. The project has developed a number of innovative multi-user interaction techniques and integrated a multimedia database within a VR environment to enhance the amount and quality of information provided to users during their virtual visits.

The remainder of this paper is organized as follows. Section 2 briefly summarizes current efforts dealing with the use of VR for the visualization of cultural heritage. Sections 3 and 4 discuss issues related the development of VR presentations for the domain of cultural heritage and new interaction instruments, respectively. Section 5 describes the DHX project by focusing on the approaches adopted to achieve high interactivity while ensuring reasonable performance and on the integration of a VR environment with a multimedia database system. Section 6 concludes the paper and outlines future work.

2 Visualization of Cultural Heritage – Current Research Efforts

Virtual reconstructions of archaeological excavations, museums, cultural environments have recently been the focus of many projects supported by the European Union that has devoted a lot of resources to this sector. An analysis of virtual reality systems adopted in these projects shows a large variety of adopted technological solutions, such immersive VR, CAVE, Augmented reality (AR) [7-13]. In particular, many projects are devoted to the 3D reconstruction of historical buildings, archeological sites, or other settlements of past cultures [1-6]. Other research efforts combine the real world with specific 3D graphic elements by mean of AR platforms [7-13].

Those virtual cultural heritage (VCH) environments have an increasingly important role in the conservation and interpretation of past culture; however, they are merely designed to visualize historical objects rather than to help the visitor to immerse him/herself in the virtual world to stand face to face with past cultures [14]. Historical human everyday life is as important as the level of details of the 3D model since visitors are fascinated by aspects of social life with which they can relate to and interact with. These are typically features of popular computer games [15]. In fact, some surveys have been directed to investigate the possible application of game theory to VCH environments [16-18]; a main issue here is to avoid that the user looks away from the cultural sense of the framework in order to win the game. Based on these investigations, several projects have developed VR worlds in which the visitor is immersed in a crowd simulation of some life activities [19-21], by means of virtual population, with the aim of increasing the reality of the reconstructed scene. However, results from the application of the game theory show that the system without adequate interactions or engaging storytelling can bore the visitors. For this reason, some interaction devices have been integrated in virtual reality system in order to make the scene less inert and users more active [22-26].

To summarize, we can say that the main technical drawbacks of current projects are the low interactivity and the lack of customization and adaptivity to a large variety of user classes. It is important that those systems be able to cater to a diverse, broad audience with different information needs, and also to support customized visits, possibly extended in time, on a per-user basis. From the contents point of view, VCH environments should be much richer, and provide many more details and more accurate reconstructions. Also, which content is presented should depend on the user specific information needs and context.

3 Issues in the Development of VCH Applications

The design and deployment of effective VCH applications requires first of all interdisciplinary development teams, which should typically include domain specialists whose main task is to gather and validate historical data, graphic designers and modelers to develop textures and 3D models, computer science specialists to optimize the data management process, and communication experts for the dissemination aspects. By exploiting such a large interdisciplinary body of expertise it is possible to develop virtual reconstructions with scientific foundations and suitable for both specialist and educational uses. The work of such teams needs however adequate support in terms of tools and repositories to enhance the development process.

Moreover it is important to complement the virtual world with a database containing information about geometries, textures, 3D objects, but also connected to the reconstructed environment. For example, it is very useful to know how a particular element was reconstructed and by whom, the historical sources from which information about the element has been acquired, and the hypothesis made during the reconstruction. It is therefore is important to enhance the virtual environment with an information repository containing all information pertaining to the VR objects and the reconstruction processes.

Several European projects have investigated approaches to represent the semantic historical and cultural contexts of a virtual environment; this is the case of the ARCO and the SCULPTEUR projects. These projects have developed ontologies for cultural assets, in order to define metadata able to describe 3D objects and their digital representations. These sets of metadata are based on different international standards, such as CIDOC and DUBLIN CORE. However, additional elements must be defined in order to describe new data types used in 3D environments.

Besides creating a VR environment accurate from a historical point of view, it is important to be able to generate and then to dynamically modify the virtual world according to the user needs and wishes. In several projects, such as CINECA, 3DMurale, ARCO, such processes are made possible by connecting the 3D world to a relational database. Unfortunately, the variety and the personalization of the possible queries are limited by VR environments that are not dynamically modifiable in their geometric structure. This problem could be solved by making parametric the environment itself, so that it can be modified according to the number and the types of the three-dimensional elements to visualize. Such a system would support a rapid prototyping of VCH applications. This is the approach taken by the CiVeDi system [35]. Such a system includes a repository of multimedia objects, implemented by a relational database; the VCH designer specifies through queries the objects to be visualized. The system automatically generates a VR environment including only these objects. The designer can also specify clustering criteria according to which objects are placed in the virtual world.

4 New Interaction Instruments

For a more effective dissemination of VCH environments it is important to develop new interaction devices enabling the users to query the environment without leaving the immersive context. The use of devices like pc or palmtop is likely to distract the user, bringing back her/him to the reality. Alternative or complementary elements, like replications of theatrical machines or pointing devices masked by magic wands, can represent more natural instruments to use. Moreover the integration of vocal recognition equipment would represent an important enhancement to VR environments. By using those devices users would be able to directly communicate with the virtual guide, for retrieving additional information or customizing the VR presentations.

Besides investigating innovative interaction devices it is also important to introduce the concept of "storytelling". As the computer game literature teaches, in order to develop applications that do not bore the player, it is necessary to conceive engaging and highly interactive scenarios. Several investigations have been carried out in order to integrate computer game techniques in cultural contexts. Everyday life moments, reconstructed historical events, environments that remember the passage and the choices made by users, the adoption of alternative pointing devices are only a few examples of how to create engaging environments that correct from a historical point of view.

Obviously the use of interactive VR systems and the adoption of innovative interaction devices require addressing the performance problems related to real time generation of immersive scenarios. Performance issues include: the development of approaches that are a compromise between high precision models, but computationally very expensive, and models with a less detailed geometry, but that can be easily and efficiently integrated in virtual environments; the development of efficient techniques for illumination, an extremely important element for a realistic rendering of the environment, but at the same time particularly onerous in terms of real time computation; the development of articulated and yet efficient user interaction techniques, supporting for example object selection and positioning by users, changes to textures, and adjustments to the lights in real time.

5 The DHX Project - A 3D Virtual Theatre

The goal of the DHX project is the development of an infrastructure for promoting the sharing and the use of the European cultural heritage, through the virtual reconstruction of sites relevant from an artistic or naturalistic point of view. As part of the DHX project, supported by the EU, we have developed an interactive digital narrative and real-time visualization of an Italian theatre during the 19th century. This project illustrates how to integrate the traditional concepts of cultural heritage with VR and database system technologies. Novel multimedia interaction devices and digital narrative representations combined with historical and architectural environments scientifically accurate offer users a real-time immersive visualization where to live experiences of the past.

One of the most characteristic aspects of the Italian social and cultural life during the 19th century was an intense and widely spread theatrical activity. During this period, in Italy as well as in the rest of Europe, going to theatre was one of the most common amusements for both the aristocracy and the common people. Only in the city of Milan, there were about 20 theatres of different sizes, where more than 1000 companies used to play. However, whereas the amount of information and documents concerning such theatrical activities is quite relevant, most of the buildings are no longer available and the ones still in place have undergone extensive changes over the years. This is the typical situation in which VR techniques can be effectively applied to propose contents otherwise impossible to access. In particular, our application aims at reconstructing an Italian theatre of that period, thus making it possible to appreciate again the original structure and appearance of such theatre. It is important to remark that our theatre is not the reconstruction of a real building, but an historically correct model, obtained by combining features of several Italian theatres. In particular, we set the reconstruction in the middle of the 19th century, and we paid particular attention to choose only those elements compatible with that time. Since technological, artistic and social changes were at that time much slower than nowadays, in some cases elements of the first part of the century can coexist with features of the second half of the century.

5.1 The 3D Model

For our model, we adopted the plan of the Canobbiana theatre, which share, with the other theatres of the period, the "all'italiana" structure. By using a 3D commercial software, "ArchiCAD", and based on the Canobbiana planimetries and elevations, we reproduced an early structural component of the scenes. The ArchiCAD model, although correct from a spatial and dimensional point of view, was imprecise with respect to the polygons arrangement. For example, we had overlapping walls, uncompleted architectural structures, and a very large number of polygons. To address this problem, almost all of the structures and architectural components were manually re-modelled in "Alias Maya"[27] using the space-dimensional references derived from the ArchiCAD measurements. Maya offers a fully integrated solution to address a complete VR system and provides some common methods to keep the application frame rate appropriate for a real-time interactive simulation.

Because our model was to be integrated in a VR environment, we were particularly careful with respect to performance in order to assure a pleasant visualization for users. The complexity of the model geometry, as well as the texture sizes and the amount of scene-specific objects are critical elements in determining the time required to render a scene. To maintain the appropriate frame rates and real-time interactions, it is best to use polygonal geometry to build the models. This technique requires less data than the data required by the use of NURB (Non-Uniform, Rational, B-spline) geometry. A NURB surface is a smooth interpolation surface that allows one to define curved forms that are not faceted, but "really" curved. However, a polygonal model supports a faster rendering calculation, since it is much less complex than a model build with NURB techniques. Furthermore, the use of primitive shapes as basis for building new elements and the cutting of unnecessary faces are useful practices to decrease the number of polygons and the time required to evaluate a surface during the rendering phase. Other methods for the development of the 3D models for real time visualization are the division of the scene into several overlapping modules and the use of the softening edge technique. The first strategy allows one to export single modules into different files, and to load them only when needed, thus achieving a better frame rate. The second technique allows one to show adjacent flat faces smoother along their surfaces during the computation of the shading representation, in order to use geometrical elements with fewer polygons. Colors, materials and textures covering the model were realized with the goal of creating a photo realistic effect. For example, we used the light mapping technique to calculate the correct lighting of every part of a scene; we then converted it into simple textures and applied it to the geometry. By following those techniques, we were able to develop a model compatible with the real time generation of immersive VR scenarios.

5.2 The Virtual Reality Framework

The main purpose of a VR system is to immerse the user in an artificial environment generating the illusion to be into a real world environment. In such a semi-immersive system, users are placed in front of a large projection screen equipped with a stereo surround system [28-31]. By wearing a pair of stereo glasses, users have the feeling of being immersed in a virtual environment. Our application can be displayed on a semi-immersive visualization system, composed of a couple of projectors, a rigid screen for back projection (size of 2,80 x 2,00 meters), a graphic workstation and a pc/touchscreen for managing the user interactions.

The application is implemented with Avango [32-33], an object-oriented framework for the development of distributed, interactive virtual reality systems. Avango is based on SGI Performer, the programming interface oriented to the development of real-time visual applications. All the advanced rendering tasks like culling, level-of-detail switching and communication with the graphics hardware, are handled by Performer. Whenever the underlying hardware allows, Performer utilizes multiple processors and multiple graphic pipelines. New functions can be implemented by subclassing and extending the existing Avango classes, which are written in the C++ programming language. In addition to the C++ API, Avango is characterized by a language binding to the interpreted language Scheme, a general purpose programming language deriving from Lisp. All high-level Avango objects can be created and manipulated from Scheme. An application is then just a collection of Scheme scripts which instantiate the desired Avango objects, call methods on them, set their field values and define relationships between them. The Avango objects are included into the nodes of a tree graph named scene-graph, which is the Perfomer data structure maintaining the information that defines a virtual world. A central concept in Avango is the concept of shared scene-graph, which is accessible from all processes composing a distributed application. Each process owns a local copy of the scene graph and the contained state information which are transparently synchronized with respect to the user. The distribution system is based on the use of a memory segment shared among the processes taking part in the application. Objects allocated into such a shared memory segment become visible to all participating processes.

5.3 The Virtual Tour

The main goal of our application is to make it possible for end-users to acquire new knowledge concerning the Italian drama theatre in the 19th century through the exploration of the virtual reconstruction of a typical theatre of the time. As we already mentioned, in order to make the exploration of the virtual world more engaging for the users, it is useful to organize the visit around a story. In our case, the story is centered on the interaction between the user and a virtual character met when the visit begins. The virtual character, symbolizing the spirit of the theatre, is looking for a talisman he lost somewhere in the building; the talisman controls the magic responsible for the success of the performed plays. The user, going along with the virtual character in the search, has the opportunity to learn the story and the most interesting features of the theatre. The virtual character performs the role of a guide, describing the environment and giving additional information if required. He also gives the user the opportunity to make choices concerning the contents of the visit. For example, in the central box, the virtual character asks the user to choose among the available plays to see one played on the stage.

The tour begins in the theatre's hall, where the user encounters the virtual character. Then they visit the stalls, the central box and, if the user is interested, a common box. The visit continues on the stage; here the user can interact with the theatrical machineries. In the end, the virtual character finds the talisman and the tour ends. The virtual character resembles Meneghino, a typical mask of the Commedia dell'Arte, the Italian ancient masked comedy born in the Renaissance and performed until the first half of the eighteenth century. Its model, developed in Maya, has the appearance of a puppet in order to accentuate its fantastic nature. The character can perform a set of gestures, developed as Maya animations and loaded into Avango. Lip synchronization with the speech is also supported.

5.4 The Interaction Framework

The core of our system is the interaction framework which includes a database system. Our VR environment makes available to users various methods to retrieve information from the database. The main method is the dialogue with the virtual character. The tour is organized according to specific steps defined by the underlying storyline, in which the guide has a crucial role. During the visit, the guide can establish a dialogue with the end-user in order to satisfy her/his curiosity. This communication method is controlled by a message board, called InteractionBoard. The InteractionBoard is accessed by means of the touch screen of a PC installed in the VR platform. The message board is triggered depending on the position of the visitor. Following a predefined dialogue structure, the guide can deal with a particular theme, either in details or according to an introductory style. The choice is up to the visitor who, by clicking on the screen, can use the message board to specify her/his preferences.

The InteractionBoard is also used to select the plays in the central box and to retrieve information about some objects placed in the environment. When the guide describes an artifact, further information is visualized on the InteractionBoard. According to the same approach, objects that are not directly connected with the guide talk can trigger an information box on the PC screen when the visitor gets closer to them. The interaction framework is complemented by special devices reproducing theatrical machineries of the 19th century. Starting from the historical information gathered from the Piccolo Teatro di Milano, which still preserves real examples of those devices, virtual machineries have been built and integrated into the theatre. For example when the user visits the backstage, she/he has the possibility to activate a wind machine to learn how, during the 19th century, the noise of the wind was simulated using this device. In order to interact with the virtual reconstruction, a real machinery, with the same size of the original one, has been built. When the user starts turning the real handle, the virtual machine turns as well. Such machinery is designed to offer to the users a new, more intuitive tool to interact with the environment.

5.5 The Multimedia Database System

To be able to answer user requests for additional information during the visit, we have included in our system a component supporting user queries against a multimedia database (see Figure 1). The core of such component is a relational database built on Oracle technology. It is articulated into four main content components related to various aspects of the theatrical activity in Milan during the 19th century: theatres, theatrical companies, performances, and documents published in the newspapers. The database also contains a variety of multimedia data, such as images, theatre maps, scene costumes and audio/video reproductions of parts of the performances.

The historical data have been collected by researchers from the Department of Performing Arts of the University of Milan. However, the data collection process is

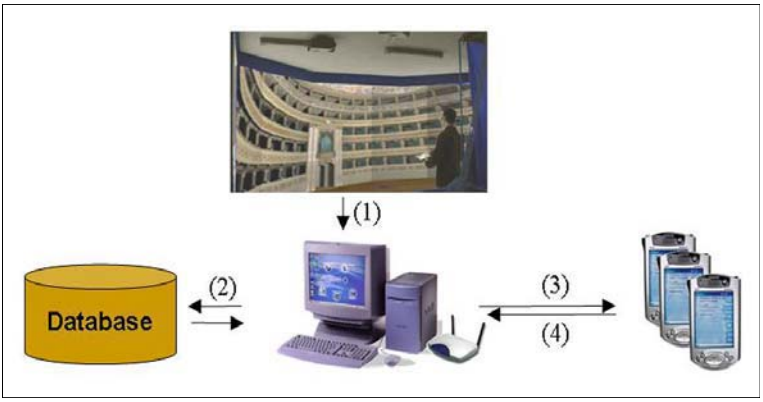


Fig. 1. DB and VR interaction

still going on. Visual and textual data are constantly growing in number because of new discoveries of theatre images, actors' and playwrights' portraits, play-bills and, in particularly interesting cases, letters, contracts or any other documents. The use of a multimedia database as the core of our interaction system makes however possible extending the database contents without having to modify the VR environment. The most difficult problem is represented by the collected images, since they are not usually in a good preservation state. As far as audio data are concerned, obviously there are no tracks from the past, so that it is necessary to mix past and present: our solution is to let young actors read the best passages from the once most famous, but nowadays often completely forgotten, plays.

Such a large amount of information must be smoothly integrated in the VR environment. A solution we have adopted is based on the use of a palmtop or of the InteractionBoard to allow users to formulate database queries in order to obtain further information about the environment. When a user is closer to an important object or the virtual guide is describing a particular place, information is loaded into the interaction-board or into the palmtop. In addition by using these devices the user has the possibility to explore the database in order to retrieve additional interesting information; for example in some places of the theatre she/he can listen or watch to several audios or videos reproducing pieces of dramas or operas from the 19th century.

6 Conclusions and Future Work

In this paper we have discussed issues concerning dynamic VR environments for the dissemination of cultural heritage content. We have also briefly described an application related to the presentation of a typical Italian drama theatre of the 19th century.

Future developments are addressed to offer an improved integration of the database system in the VR environment. In our suggested solution the 3D virtual tour is integrated with an additional tool: a database application used to obtain further information about what the user can see during his virtual reality experience. But, probably this method spoils the user's feeling of being immersed in the reconstructed environment because his attention is shifted to the database application placed either on the palmtop or on the InteractionBoard. For this, we are investigating other techniques for supporting database queries in VR environments.

To make the visit even more engaging we are also considering the introduction of new theatrical machineries and the extension of the application to support multi-user interactions. Multi-user interactions are addressed to create a shared virtual world where remote co-players can meet and interact in the same virtual, three-dimensional environment. The DHX consortium is evaluating and testing multiple techniques to integrate multi-user features in the AVANGO framework. Finally, we are investigating the extension of the InteractionBoard with voice-input to allow the visitor to control the dialogue by directly speaking to the virtual guide.

Acknowledgments. The work reported in this paper supported by the EU under the Project DHX (Project n. IST-2001-33476) and under the DELOS Network of Excellence.

References

1. BBC web site.http://www.bbc.co.uk/education/history/3d.shtml
2. Virtual Heritage Network. http://www.virtualheritage.net
3. Learning Sites Inc. http://www.learningsites.com/Frame_layout01.htm
4. CHARISMATIC IST project.http://www.charismatic-project.com/athens.html
5. THE THEATRON project: http://www.theatron.co.uk
6. N.U.M.E. project.http://www.storiaeinformatica.it/nume/english/ntitolo_eng.html
7. N. Magnenat-Thalmann, G. Papagiannakis,A. Foni,M. Arevalo,N. Cadi-Yazli, "Simulating life in ancient sites using mixed reality technology", *CEIG04*, Seville, May 2004.
8. G. Papagiannakis, S. Schertenleib, M. Ponder, M.Arévalo, N. Magnenat-Thalmann, D. Thalmann, "Real-Time Virtual Humans in AR Sites", *IEE Visual Media Production* (CVMP), London UK, March 2004, pp. 273-276.
9. L. Vacchetti, V. Lepetit, G. Papagiannakis, M. Ponder, P. Fua, N. Magnenat-Thalmann, and D. Thalmann, Stable Real-Time Interaction Between Virtual Humans and Real Scenes", *International Conference on 3-D Digital Imaging and Modeling*, Banff, October 2003.
10. G. Papagiannakis, M. Ponder, T. Molet, S. Kshirsagar, F. Cordier, N. Magnenat-Thalmann, D. Thalmann, LIFEPLUS: Revival of life in ancient Pompeii, *Virtual Systems and Multimedia, VSMM 2002 Virtual*, October 2002.
11. R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier and B MacIntyre, "Recent advances in augmented reality", *Computer Graphics and Applications*, IEEE Computer Society Press, Volume 21, No. 6, Nov/Dec 2001, pp. 34-47.
12. Y. S. Kim, T. Kesavadas, S. Paley and D. Sanders, "Real-time Animation of King Ashurnasir-pal II (883-859 BC) in the Virtual Recreated Northwest Palace", *7th International Symposium on Virtual Systems and Multimedia*, IEEE Computer Society Press, Berkeley USA, 2001, pp 128-136.
13. THE ARCHAEOGUIDE Project. http://archeoguide.intranet.gr/project.htm
14. Lidunn Mosaker, "Visualising historical knowledge using virtual reality technology", *Digital Creativity*, Volume 12, No. 1, 2001, pp. 15–25.
15. Yu Suzuki, author of "Shenmue", An interactive film for SEGA Dreamcast.
16. E. Champion, "Heritage Role Playing-History as an Interactive Digital Game", *IE2004 Australian Workshop on Interactive Entertainment*, Sydney, February 2004.
17. E. Champion, "Online Exploration of Mayan Culture", *VSMM2003 Hybrid Reality*, Montreal, October 2003.
18. E. Champion, "Applying Game Design Theory to Virtual Heritage Environments", *Graphite Annual Conference*, Melbourne, February 2003.
19. G. Papagiannakis, A. Foni, N. Magnenat-Thalmann, "Real-Time recreated ceremonies in VR restituted cultural heritage sites", *CIPA 19th International Symposium*, September 2003, pp.235-240.
20. B. Ulicny and D. Thalmann, "Crowd simulation for virtual heritage", *Proc. First International Workshop on 3d Virtual Heritage*, Geneva, 2002, pp. 28-32.
21. The CAHRISMA Project. http://www.miralab.unige.ch//3research/research_project.cfm?projectid=CAHRISMA
22. J. Ciger, M. Gutierrez, F. Vexo and D.Thalmann, "The Magic Wand", *Proceedings of SCCG '2003*, Budmerice 2003, pp. 132-138.
23. P. Jablonka, S. Kirchner and J. Serangeli, "TroiaVR: A Virtual Reality Model of Troy and the Troad", *CAA 2002 The Digital Heritage of Archaeology*, Heraklion CreteGreece, April 2002. http://www.uni-tuebingen.de/troia

24. S. Hynst, M. Gervautz, M. Grabner and K. Schindler, "A work-flow and data model for reconstruction, management, and visualization of archaeological sites", *ACM Siggraph Symposium on Virtual Reality, Archaeology and Cultural Heritage 2001*, Glyfada, November 2001.
25. J. Cosmas, T. Itegaki, D. Green, E. Grabczewski, F. Weimer, L. Van Gool, A. Zalesny, D. Vanrintel, F. Leberl, M. Grabner, K. Schindler, K. Karner, M. Gervautz, S. Hynst, M. Waelkens, M. Pollefeys, R. DeGeest, R. Sablatnig and M. Kampel, "3d MURALE: A Multimedia System for Archaeology", *ACM Siggraph Symposium on Virtual Reality, Archaeology and Cultural Heritage 2001*, Glyfada, November 2001.
26. GRAPHISOFT'S ARCHICAD is an integrated, object-oriented 2D drafting and 3d architectural design software. http://www.intcad.com/archicad.html
27. MAYA 5: 3d animation and visual effects software. www.alias.com
28. P.A. Howarth and P.J. Costello, "The Nauseogenicity of Using a Head-Mounted Display, Configured as a Personal Viewing System, for an Hour", *Proceedings of the Second FIVE International Conference*, Pisa, December 1996. http://www.lboro.ac.uk/departments/hu/groups/ viserg/pisa.htm
29. Computer Engineering Research Group, Electrical and Computer Engineering Department, University of Toronto. http://wearcam.org/head-mounted-displays.html
30. Kaiser Electro-Optics, Inc. http://www.keo.com/
31. NVIS: New Virtual Imaging System Inc. http://www.nvis.com/
32. H. Tramberend, "AVANGO: A Distributed Virtual Reality Framework", *IEEE Virtual Reality '99*, JW Marriott Hotel, Houston, March 1999.
33. AVANGO: Object-oriented framework for Virtual Reality applications. www.avango.com
34. DHX (Digital Artistic and Ecological Heritage). EU project, framework of the IST 2001-33476. http://www.eurasian-dhx.org/
35. P. Mazzoleni, E. Bertino, E. Ferrari, S. Valtolina, CiVeDi: A Customized Virtual Environment for Database Interaction, *ACM SIGMOD RECORD*, Vol.33, No.3, September 2004.

From MultiMedia to UniversalMedia

Masahito Hirakawa

Interdisciplinary Faculty of Science and Engineering, Shimane University,
Nishikawatsu 1060, Matsue 690-8504, Japan
hirakawa@cis.shimane-u.ac.jp

Abstract. Multimedia is in common use nowadays. However it is just a collection of multiple media which include text, picture, video, and audio, and they are not really integrated. In this paper, we present a notion of UniversalMedia. It is not just text, or picture, or video, or audio. Rather it is an all-in-one active object. In addition, some trials the author has carried out as a step toward realization of such active media are explained.

1 Introduction

Multimedia initiatives began more than 15 years ago. Nowadays this word is in common use. Now then, what is multimedia? For example, in [1], multimedia is defined by “the use of computers to present text, graphics, video, animation, and sound in an integrated way.” Integration of multiple media - that’s fine. However, I would say that *integrated* is recognized in most cases as *synchronized* in multimedia community.

For example, consider the Synchronized Multimedia Integration Language (SMIL) [2]. It is an XML-based language, and used for rich-media presentations which integrate streaming audio and video with image, text or any other media type. It enables authors to specify what and when should be presented; that is, enables them to control the precise time that a sentence is spoken and make it coincide with the display of a given image appearing on the screen.

We are also faced with a serious issue that there are many different hardware and formats for each medium. With regard to this, Golshani brought up a criticism as in the following [3]:

Why should consumers have to purchase a different device for every type of multimedia file they want to play? As long as a processing unit has adequate memory, a reader (perhaps optical), a visual display unit, and an audio output device, shouldn’t the consumer be able to enjoy a movie, listen to a song, and view pictures? When will we be freed from vendor-dependent formats and technologies?

Such technology-centered approach doesn’t help us to solve the problem. Meanwhile, graphic/web/multimedia designers have experienced for years how the information should be organized and presented. But they are interested in utilizing media which are available. We would say that we, as engineers, have responsibility of preparing well-formed media to attain further success in media research.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 104 – 114, 2005.

Here, video and animation are powerful media in the sense that they include motion pictures, audios, and texts. The number of media being adopted is, however, not a main concern. What is more important is how natural and fit the media are. The term *multi*media may mislead us into useless research. We thus propose in this paper using *UniversalMedia* as a replacement of multimedia.

The same wording “universal media” is used as a Web3D consortium working group solution that increases the realism of on-line Web3D worlds (VRML, Java 3D and other on-line 3D technologies) [4]. It aims at allowing content authors to create media-rich worlds that can be instantly loaded over even the slowest dial-up modem Internet connections by providing a small, cross-platform library of locally resident media elements and a uniform resource name mechanism. To tell our idea from this Web3D consortium working group solution, we capitalize the first character of the words and express it by UniversalMedia.

In the rest of the paper, what we intend by UniversalMedia is discussed, followed by some works the author has carried out as a step toward such goal. Specifically, in Section 2, a basic idea of UniversalMedia is presented. Our experimental works will be explained extensively in Section 3. Finally, in Section 4, a concluding remark is given.

2 UniversalMedia

When we talk about multimedia, one may refer to data entities such as text, image, audio, and video. Meanwhile, multimedia systems take advantage of human senses to facilitate our communication with the computer. Interaction between human and computer can be improved by the use of multiple sensing channels such as gesture, voice, gaze, and facial expression. In this case, the word *modality* is preferably used rather than media. A multimodal interface exploits different sensations toward realization of advanced human-computer interface which is natural and expressive to people.

Jain who was an editor-in-chief of the IEEE MultiMedia magazine quoted the fable “Six Blind Men and an Elephant” and mentioned multimedia as in the following [5]:

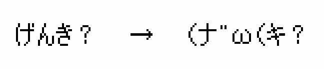
> *Each blind man perceived something completely different – the nose as a snake, the leg as a tree trunk, the tail as a rope – and non comprehended the whole. Likewise, a narrow perspective leads most people to consider as multimedia only limited aspects of it. Multimedia is not just JPEG, or networking, or authoring tools, or virtual reality, or information systems. Multimedia is computers using multiple media to let people deal with information naturally.*

Go and Carroll also referred to the same story in [6] as a metaphor for understanding different views of scenario-based system design.

Meanwhile, when people talk about multimedia (as data entities), they might have a common understanding that more is better - videos are better than pictures, and pictures are better than texts. I would say this is not correct. Videos are not the ultimate media. It is argued in [7] that sometimes a few words are worth more than any number of pictures.

Is it really essential for us to separate media into texts, pictures, videos, voices, and so forth? Let us give you an interesting example.

Mobile phones are no longer a tool used only for talking. They are used mostly for sending text messages to others. Interestingly, some - mostly young girls - in Japan recognize Hiragana, the Japanese cursive syllabary, characters as images and form a sentence by a combination of completely irrelevant characters. Those characters or expressions are called Gal-Moji (characters of young girls) or Heta-Moji (awkward characters). Figure 1 shows an example.



げんき？ → (ナ゛ω(キ？

Fig. 1. Example of Gal-Moji. Left-hand side is a Hiragana expression saying "How are you?" and right-hand side is its corresponding Gal-Moji expression

Those who use Gal-Moji are not peculiar people any longer. In fact, a Karaoke company has started a service of attaching Gal-Moji to songs.

Gal-Moji users don't care about the type of media. What they are interested in is how much the expressiveness of media is. This shows a necessity of investigating a new type of media that is All-in-One. UniversalMedia we present in this paper is such media. Let us modify the Jain's message and say that "UniversalMedia is not just texts, or pictures, or videos, or sounds. UniversalMedia is computers using all-in-one media to let people deal with information naturally."

Specifically, UniversalMedia should support the following features.

1. Multiple media expressiveness

This is of course a must. However it is noted again that the information could be in text, picture, video, or any other forms, but its meaning may not be captured properly just by looking at the corresponding bit sequence. In other words, it is needed to provide a media conversion facility.

2. Interactiveness

It is obvious that the user communicates with the computer continuously by changing his/her query to get a reasonable solution or placing commands to complete a task. Sophisticated interactive capabilities are essential.

3. Invisible/harmonized computing

As noted, UniversalMedia is an all-in-one medium. The word all-in-one, however, doesn't imply just a combination of individual media entities. Those entities should be harmonized so that people feel the medium is one coherent object in its expression and don't need to aware what entity they are manipulating. In other words, this idea is a media version of ubiquitous/invisible computing proposed by Weiser [8]. Media which are a vehicle for communicating thought or the information in general vanish into the background.

4. Context/situation-sensitivity

Context- or situation-sensitive computing is one of the active topics in the computer community. The meaning of UniversalMedia changes depending on the context to which the media is situated. This implies that UniversalMedia are personalized. Personalization is the process of tailoring the media information to individual users' characteristics or preferences by the use of information either previously obtained (i.e., personal interaction history) or provided in real-time about the user. This helps the user to get a solution satisfying his/her needs more effectively and efficiently, interact with the computer faster and easier and, consequently, have better satisfaction.

5. Knowledgeable
UniversalMedia are exchanged among people to have better effect. To support this, like the semantic web [9], UniversalMedia should be knowledgeable about themselves and self-explanatory.

6. Lifelong capturing
To make the context sensitivity higher, the system should capture the lifelong users' activities. Making records of human life itself is of great importance in UniversalMedia. [10] mentioned that reviewing recorded, everyday personal interactions, family images, or other information items, such as notes, at any time might lend itself to a human experience with greater intensity and enjoyment. Digitally logging every moment and element of their lives often results in the unanticipated capture of valuable moments. Furthermore, medium is changed - maintained and evolved. It is requested for UniversalMedia to support automated techniques to assist the maintaining, understanding and restructuring of very large and complex media data.

In the next section we will explain some of the trials we have done with relation to UniversalMedia.

3 Trials Toward Realization of UniversalMedia

3.1 Situation-Sensitive, Lifelong Information Management

3.1.1 Situated Computing

When the desktop metaphor was presented three decades ago, the computer was considered to be a tool for specific tasks such as writing a document, drawing an image and calculating budgets, even though those applications could be run in parallel. This design was reasonable when the user's position was fixed inside the office and the computers had very limited capabilities. With the technological advancement both in hardware and software, it has become possible to develop very advanced tiny mobile/wearable computers. Those computers can be connected to powerful mobile peripheral devices such as a video camera and a GPS receiver, and they also can possess the online wireless connection to the network. However, the interaction with computers are still designed based on the desktop metaphor.

The main objective in situated computing in [11] was to enhance the human-computer interaction through simplified interfacing based on the user's real world situation. Here the system organization and data management play an important roll in the success of situated computing as well as they influence on the design of user-interaction. The situation metaphor, which was proposed as the foundation for situated computing, goes beyond its semantic level in achieving the requirements of situated computing. It is not only an interactive metaphor for visual and non-visual interaction but also a design metaphor which influences on the functional and algorithmic design of applications.

The development methodology consists of three layers. The first level gives the theoretical design adopted in the declaration of situation metaphor. The SIFF (Situated Information Filing and Filtering) framework providing the software abstraction as the core foundation for all services/applications is on the second level. At the third level, tools and applications are organized based on the SIFF framework. We have implemented four example applications which include situation-dependent browser, augmented album, pattern browser, and situation-dependent chat system. In the following we explain two of them, augment album and pattern browser.

3.1.2 Augment Album

Multimedia database is one of the highly discussed areas in the multimedia community. In such research area, most of the studies have focused towards the content-based retrieval. It is a very costly task as it is necessary to carry out a lot of processing to extract features contained in multimedia data. The complex unstructured nature of multimedia data makes it very difficult to understand their content without direct human participation. In [12] we proposed the use of contextual information to enrich the meaning of multimedia data, which is the complement to content-based approach.

Here let us consider the following two queries for a personal video/image collection; “Find pictures in which a dog is running” and “Find pictures of X’mas party last year at my uncle’s house”. The difference is that the latter one is much more context-oriented and hence the content-based approach doesn’t work well. Frankly speaking it is impossible to find such pictures in the existing content-retrieval technologies (consider the difficulty in finding uncle’s face or house, for example).

Advantage of the context-based approach could also be explained differently as in the following. Filing of data is usually considered a separate independent activity from finding of data, but a proper filing is the key to easy finding. If filing is done improperly, a collection of data becomes almost meaningless even though it has essentially lots of useful information. The inadequacy of keyword-oriented indexing for multimedia database retrieval shows this fact. The context-based retrieval for personal video/image database is considered a proposal of connecting finding with filing, and thus allows the user to retrieve the database effectively. Ideally, we say that queries can be formed according to the way the user *remembers* about images/video clips in the database.

Though the content of a picture itself represents a situation or situations, our approach was not focused on content-based analysis of the picture. We assume that contextual information is gathered in other ways. In our trial, three contextual parameters are considered: geographical location, time and the corresponding events which are made the user to take pictures.

Three components, Map Component, Time Frame Component and Events Component, which correspond to parameters in interpreting the user-situation, are used in designing graphical interface that provides the facilities to browse, query, and modify this collection (see Fig. 2).

By capturing the contextual information of multimedia data sources like digital videos and images when they are produced, it becomes possible to enrich the semantic of those data sources and some complex content-based analysis can be simplified. For example, in order to identify photographs taken at the same location, we do not need to use complex content analysis technique. The context-based approach is also promising to provide better human-computer interaction.

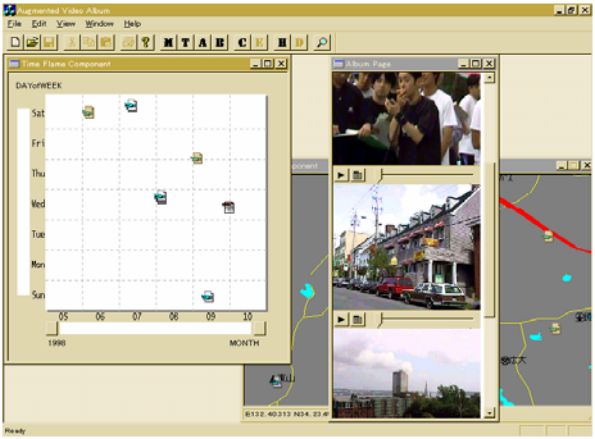


Fig. 2. By specifying a condition with Map, Time Frame, or Events components, the system retrieves the matched video clips. Video clips can be played by clicking the corresponding one on the screen

3.1.3 Pattern Browser

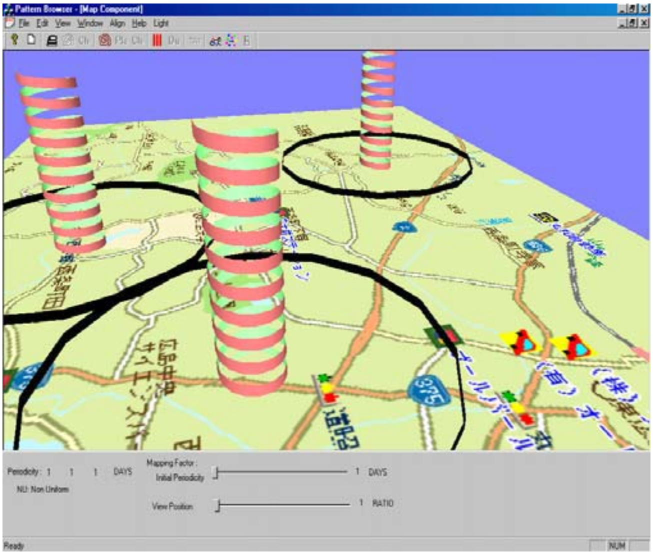
Pattern Browser is an interactive visualization tool and aims at helping a user to find valuable spatiotemporal patterns in lifelong data [13].

Pattern Browser has mainly two windows allowing the user to specify conditions for the visualization and also to observe patterns from different angles. One is a map window and the other a spiral window.

Map window displays a geographical map in a three dimensional visual space. When the user double clicks a certain location on the map, a spiral appears with the boundary indicating the geographical area it covers, as shown in Fig. 3. Here spirals are used to represent the occurrence of temporal events. This spiral can be moved and its boundary can also be adjusted by dragging it. In this way, the user specifies any number of spirals, as he/she likes. Of course, zoom-in and -out operations are provided so as to be able to change the scale of the map.

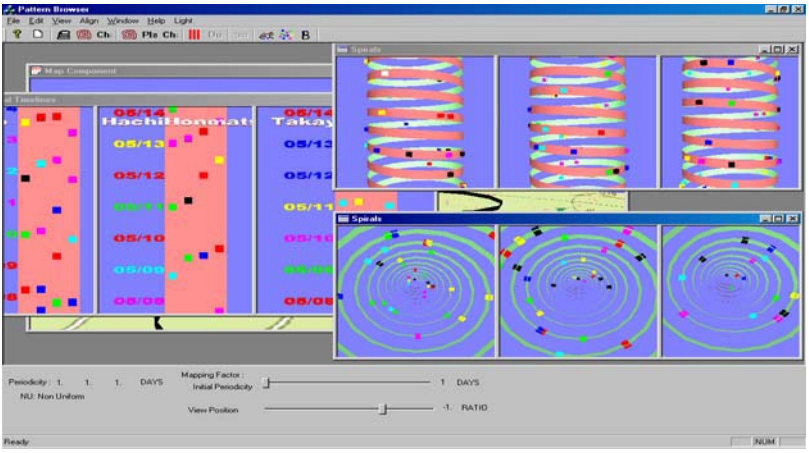
A simple linear timeline which is used commonly doesn't support viewing periodic aspects of the data, though the periodicity is one of the important factors in recognizing patterns. Hence we adopted the spiral expression for the timeline.

In general, a spiral can take two forms depending on the dimensions in which it is visualized. The 2D form of a spiral, known as planar spiral, shows a continuous path of a point in a plane moving around a central point. The 3D form of a spiral is a helix, which is a coil formed by a wire around a uniform tube. Although these two spirals are conceptually described as distinct objects, they can be presented as two different views of a single object in the visualization process. The top view of a helix is presented as its planar spiral in the visualization.


Pattern Browser - [Map Component]
File Edit View Window Align Help Light
Ready
NUM

Periodicity : 1 1 1 DAYS
NU: Non Uniform
Mapping Factor :
Initial Periodicity 1 DAYS
View Position 1 RATIO

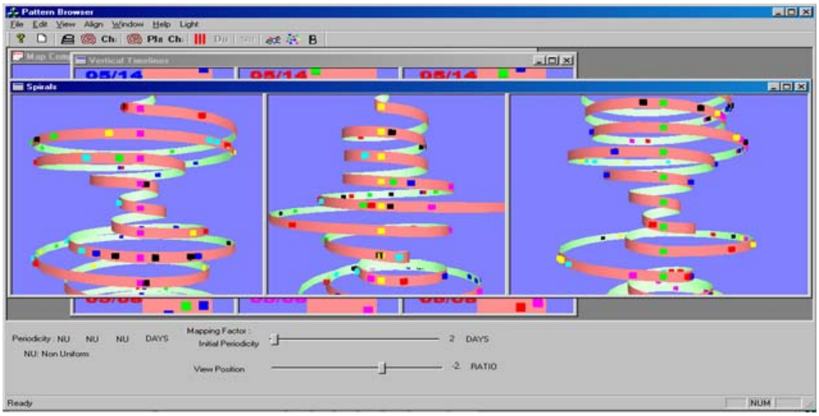
Fig. 3. Map window displays a geographical map with spirals



Periodicity : 1. 1. 1. DAYS
NU: Non Uniform
Mapping Factor :
Initial Periodicity 1 DAYS
View Position -1. RATIO

Fig. 4. Spiral views and linear view of timelines in spiral window. Initially, the radius of spirals proportionate to a single day. This can be modified by changing the visual radius of the spiral or changing the time scaling factor. Furthermore th`e helix spiral view can be changed to planar view or linear view

Spirals with colored square icons which represent events are displayed on the spiral window, as shown in Fig. 4. The user can navigate through this space by changing the viewpoint, rotating spirals, or adjusting the visual radius of spirals (i.e., scale of time periodicity). Furthermore, target events belonging to a certain category can be aligned to help the user find a temporal pattern, resulting in a non-uniform spiral, as shown in Fig. 5.



Periodicity : NU NU NU DAYS
NU: Non Uniform
Mapping Factor :
Initial Periodicity 2 DAYS
View Position -2 RATIO

Fig. 5. Snapshot of irregular spirals when icons having yellow color (e.g., sale of product A) are aligned. These patterns can be read in two different ways. First each individual pattern is read from top to bottom analyzing its shape and icons. The user does the horizontal reading while comparing three patterns together. The first pattern says that the sale of product A increases and then decreases repeatedly. On the other hand, the second pattern shows a rather uniform sale of the product A but sometimes a sudden decrease. The third pattern is similar to the first one. In addition, it would be said that the sale decreases at almost the same time with the first pattern

3.2 Transparent Interface Media

Media research covers human-computer interaction. One interesting approach from UniversalMedia point of view would be augmented reality, or mixed reality in a broader sense, where computer-generated digital information is mixed with real world objects and they are perceptually assimilated. In other words, we can feel that objects in the real world are augmented in their properties and functions with the help of computers. In addition computers are invisible to us. We don't need to be aware of the computer and learn how to operate it. What to do for us is just behaves as usual as they do in the real world - for example, placing a book on a desk, opening it, and pointing at an object on the page.

There are several different approaches to integrate the digital information with real world objects. Utilization of a head-mounted display (HMD) is the most common approach. Though HMDs are powerful and advantageous as a tool to immerse the user in a smartly integrated environment, the user is forced to wear an HMD. This is bothersome for the user and impediment to his/her work.

As another approach, we assumed a projector-based one where digital information is overlaid onto real world objects through a video projector [14], [15]. Unlike other existing projector-based systems, we proposed to use a transparent, thin film-type screen which is placed between the user and the real world objects. The user can see those objects through the display. When the user selects an object by pointing at it with his/her finger, the associated computer-generated information is presented on the display.

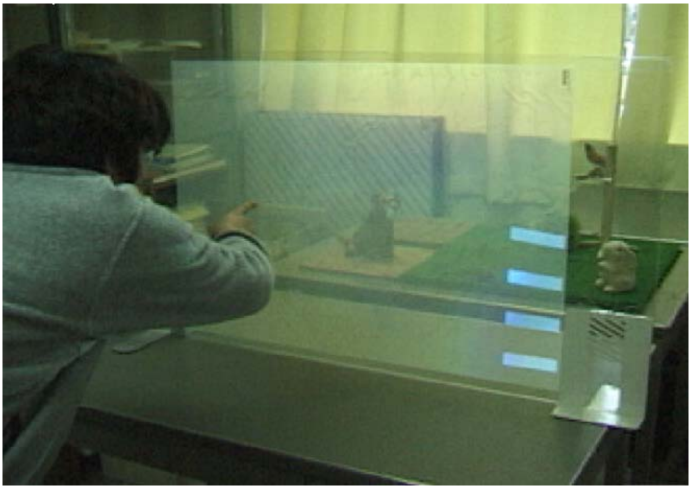


Fig. 6. The user sees and points at an object through the display. The associated information is presented on the display. Position of the information on the display changes in accordance with the movement of the user. The display between the user and objects in the real world blocks his/her manipulation of those objects with hands. However, in such applications as museum and zoo, visitors are not allowed to touch exhibits and animals for protection, security, and safety

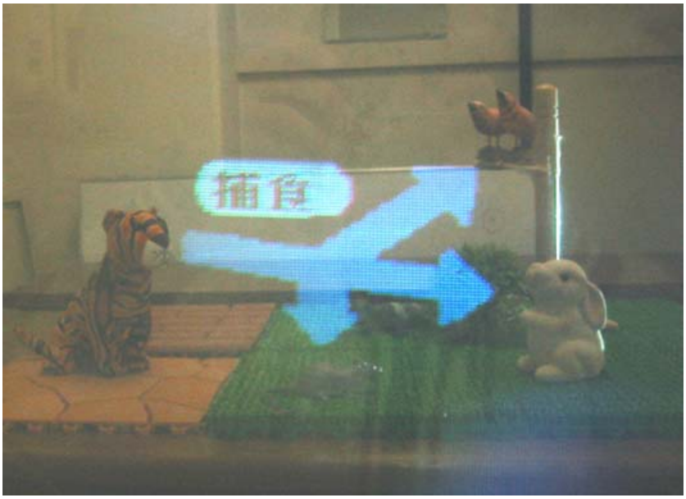


Fig. 7. The system can present a relationship among objects as well. When the user points at a tiger, a food chain relationship can be presented with an arrow, explaining that rabbits, mice, and birds are eaten by tiger

Interestingly, the display is almost transparent and doesn't have any screen frame which separates the virtual world from the real world. The real and virtual worlds can be integrated seamlessly, resulting in a possibility of truly smart media space.

We have demonstrated the applicability of the system to several application domains which include zoo/museum and library [14]. Figures 6 and 7 show snapshots of the system.

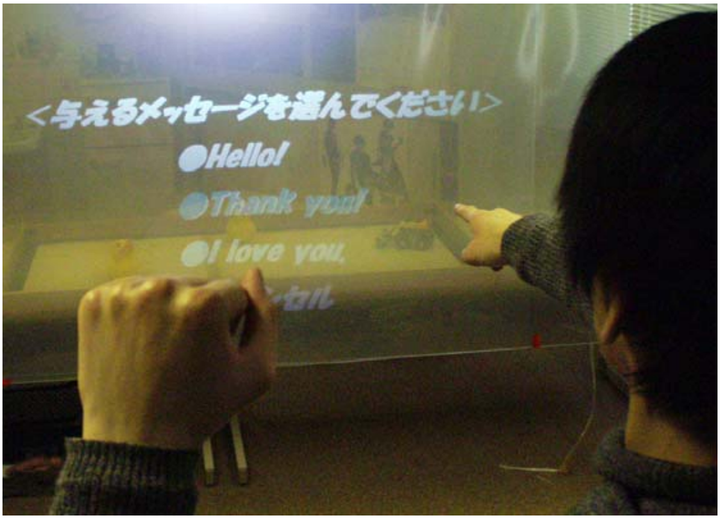


Fig. 8. A real world object can work as a career of a message. The message is kept within the object and transferred to someone who is interested in the object. In the current implementation, several possible messages are provided in advance and the user selects one of them. The message can be recalled by choosing the corresponding object later. Incorporation of a voice input/output facility enhances the usefulness of the application

Furthermore, additional facilities allowing multiple users to exchange their ideas and/or keep pace each other via objects in the real world have been implemented so that the system can be used in a collaborative working environment [15]. Real world objects serve as a mediator among the users, as well as a media to be augmented in its value by the computer. A snapshot of the system is given in Fig. 8.

4 Concluding Remark

Media research is crucial for the success in next-generation computing. Considering that the word multimedia is misleading, we presented in this paper a concept of UniversalMedia to make the point clearer. Simply, UniversalMedia is not just texts, or pictures, or videos, or sounds. It is all-in-one media to let people deal with information naturally. In addition, UniversalMedia is not passive in the sense that it is capable of sensing others and changing its role with the correspondence to others' properties.

Moreover, it keeps a lifelong history of target occurrences and events, resulting in achieving media evolution.

We also presented some trials toward realization of such media. However they are considered just in the initial stage. Further studies must follow.

References

1. http://www.webopedia.com
2. http://www.w3.org/AudioVideo/
3. Golshani, F.: Pushing Forward with Interoperability. IEEE MultiMedia, Vol. 12, No. 1 (2005) c3
4. Walsh, A.E.: Universal Media: Media-Rich Content for Bandwidth-Starved Devices. ACM SIGGRAPH Computer Graphics, Vol. 34 (2000) 37-41
5. Jain, R.: Multimedia Computing. IEEE MultiMedia, Vol. 1, No.1 (1994) 3-4
6. Go, K. and Carroll, J.M.: The Blind Men and the Elephant: Views of Scenario-Based System Design. ACM interactions, Vol. XI.6 (2004) 44-53
7. Johnson, J.: Textual Bloopers. ACM interactions, Vol. VII.5 (2000) 28-48
8. Weiser, M.: The Computer for the 21st Century. Scientific American, Vol. 265, No. 3 (1991) 94-104
9. Berners-Lee, T., Hendler, J., and Lassila, O.: The Semantic Web. Scientific American, Vol. 284, No. 5 (2001) 34-43
10. Nack, F.: You Must Remember This. IEEE MultiMedia, Vol. 12, No. 1 (2005) 4-7
11. Hirakawa, M. and Hewagamage, K.P.: Situated Computing: A Paradigm for the Mobile User Interaction with Multimedia Sources. Annals of Software Engineering, Vol. 12, No. 1 (2001) 213-239
12. Hewagamage, K.P. and Hirakawa, M.: Augmented Album: Situation-dependent System for a Personal Digital Video/Image Collection. Proc., IEEE International Conference on Multimedia and Expo (2000) 323-326
13. Hewagamage, K.P. and Hirakawa, M.: An Interactive Visual Language for Spatiotemporal Patterns. Journal of Visual Languages and Computing, Vol.12, No.3 (2001) 325-349
14. Hirakawa, M. Kojima, Y. and Yoshitaka, A.: Transparent Interface: A Seamless Media Space Integrating the Real and Virtual Worlds. Proc., IEEE Symposium on Human Centric Computing Languages and Environments (2003) 120-122
15. Hirakawa, M. and Koike, S.: A Collaborative Augmented Reality System using Transparent Display. Proc., International Workshop on Multimedia and Web Design (2004) 410-416

The Multimedia Semantic Web

William I. Grosky

Department of Computer and Information Science,
University of Michigan-Dearborn, Dearborn, MI 48128
wgrosky@umich.edu

Abstract. This talk discusses our ideas on the importation of multimedia information into the semantic web paradigm and how the nature of multimedia data should cause this paradigm to undergo some interesting transformations.

1 Introduction

Our approach to the semantic web is broader than just simply importing multimedia data types into current semantic web approaches. It is well known that interpretation depends on context, whether for a work of art, a piece of literature, or a natural language utterance. Our work has addressed the dynamic context of a collection of linked multimedia documents, of which the web is a perfect example. Contextual document semantics emerge through identification of various users' browsing paths though this multimedia collection. Our work has devised techniques that use multimedia information as part of this determination. Some implications of our approach are that the author of a web page cannot completely define that document's semantics and that semantics emerge through use.

With the advent of search engines, traditional web browsing, consisting of following the links of a page, has been replaced by search and navigate. In a search engine, a user enters a query string and the search engine retrieves a list of URLs that match the query in the order of relevance. The search engine also provides a brief summary of the links. The user browses the web starting at the search-result page. In order to bridge the semantic gap, it is essential to understand the user’s needs. This can be accomplished by tracing the web pages he traversed. From a web perspective, we argue that an author publishes a document and provides links to other documents that he thinks are relevant to the topic of that page. We classify the process of analyzing documents using such links provided by the author as *static analysis*. Such analysis can only derive the page’s default semantics: that conceived by the author. But, the actual semantics of the page, we believe, is that which emerges over a period of time, when several users browse through the interconnected web.

Short sub-paths of a user’s browsing path, bounded by so-called *semantic breakpoints,* exhibit uniform semantics, and these semantics can be captured, easily represented, and used to our advantage. The browsing path of a user contributes to the semantics of the pages traversed. The sequence of pages browsed by the user helps determine the semantics of the browsing path and this, in turn, helps determine the semantics of each page in the browsing path. We call this process *dynamic analysis,*

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 115 – 118, 2005.

and the derived semantics will be emergent semantics [1]. The path traversed by a user is not limited to the links within a page. It could be something a user types in or he clicks on URLs delivered to him by other means such as instant messaging or emails. Deriving the emergent semantics of documents will enable us to derive the semantics of the browsing path and eventually detecting the user's intent or the general trend.

In our work, user browsing paths are used to derive emergent semantics of web pages. We include textual features and image features of the web pages along paths in order to build a feature-path matrix. We then discuss the emerging MPEG-7 standard in the context of our work and the semantic web paradigm.

2 Multimedia Semantics

There is currently no separation between content and presentation in most web pages. The use of XML and XSLT is limited to a small subset of the entire World Wide web. The process of extracting relevant keywords from the URLs that comprise a path involves parsing the web page, eliminating all the presentation information, advertisements, insignificant graphics, and other navigational aids. After removing the presentation details we have a web document that is similar to any other textual document. A path is made up of a multiple of such web pages. The various keywords in each web page in the path are counted to generate the textual portion of the feature-path matrix. The list of unique keywords is generated over a period of time as each web page is processed.

Similar to text, we extract visual keywords from JPEG images appearing on web pages. These visual keywords are features that can be extracted and counted. These types of images are the predominant images appearing on the web, and we process these images in the compressed domain. The JPEG standard is color blind and does not specify any particular color space. The existing de-facto JPEG file format standards specify YC_bC_r, since this permits greater compression. The luminance component (Y) contains the grayscale and the chrominance components (C_r and C_b) contain the color information. The DC $(0,0)^{th}$ coefficient of the cosine transform of each 8 × 8 block represents that block's average intensity value. We use the DC coefficients of all the three Y, C_b and C_r components as the basis for our features.

As part of our experiments in content-based image retrieval using JPEG images, we have discovered that the use of the average deviation of the DC coefficients of each DCT block from its 8 neighbors to be very effective. Our approach takes care of the blocks around the edges where there may not be 8 neighbors. The number of neighbors considered can be expanded to simulate the variable-size sliding window as in the WALRUS system [2]. The benefit of our approach is that we are able to extract the features directly by reading the compressed domain image file without any additional wavelet computational overhead. Also, the use of the deviation helps in identifying a uniform region or an edge.

Using all these features, we applied the standard information retrieval techniques concerning stop features, weighting policies, normalization, and dimensional reduction in the construction of our path-feature matrix [3], which is based on aggregating the various feature counts along semantically uniform subpaths.

3 Coherent Semantic Subpaths

In our work, browsing path history is used to derive web page semantics. Isolating and processing just a page will not infer the actual semantics, but an ordered sequence of pages, called the browsing path, will. The browsing path of a user is broken into contiguous sub-paths. The semantic break points are identified along the user's browsing path where the semantics change appreciably. The sequence of pages that exhibit similar semantics is clustered as a subpath. The semantics of a single page is actually the special case where the length of the subpath is 1. We contend that the semantics of a web page is derived from its context. The location of a web page in any path contributes to its emergent semantics. Due to this dependency, the identification of a semantic breakpoint in a long browsing path (with multiple semantics) is crucial [4]. Starting with a set of users' browsing paths, the first task is to break long browsing paths into sub-paths that exhibit uniform semantics. A long browsing path is usually composed of multiple semantics corresponding to the browsing activity of a typical user. It is also possible that the entire path could exhibit uniform semantics. Having subpaths with uniform semantics provides a starting point to derive the semantics of each of the web pages in the sub-paths.

4 Some Typical Experiments

We collected web pages by simulating user browsing through a collection of newsweb sites like Google news, New York Times and Columbia News Blaster. The test data set for a typical proof-of-concept experiment consisted of 16 paths with a maximum of 17 URLs per path. The images from each of these pages were processed separately. The weighted and normalized textual keywords were then combined with the separately weighted and normalized visual keywords to build the keyword-path matrix.

We conducted several types of experiments:

- Using only paths with uniform semantics
- Using only paths with multiple semantics
- Using only individual web pages

We experimented with textual keywords as well as both textual and visual keywords. We experimented with path queries of length 1 and of lengths greater than 1. Queries consisted solely of textual keywords. The visual keywords were used to aid the textual keywords during the latent semantic analysis phase [5], and were not used in the query processing.

The following general conclusions resulted from our experiments:

- Queries using semantically uniform paths of length greater than 1 outperformed queries using semantically uniform paths of length 1. The longer the path the better the performance.
- The use of visual keywords had mixed results. When the images associated with a topic are coherent, then the visual keywords aid the textual keywords.
- The use of visual keywords helped discriminate between topics when textual keywords cannot.

4 MPEG-7

Until recently, managing multimedia information has been quite ad-hoc. Tools that were developed by one research group could not easily be repurposed and used as components in the design of more complex systems by other research groups. This had been caused, largely, by the complexity of devising transformations between the different data and metadata representation schemes used by these different groups. The growing popularity of MPEG-7 has changed this state of affairs. Allowing for the representation of a multiplicity of multimedia features in a uniform fashion, it is becoming much easier to transform from one representation scheme to another in a more automatic and transparent fashion than was previously possible. Even though MPEG-7 must further evolve if it is to transparently interoperate with other metadata standards, we have come a long way from the early days of digital multimedia.

MPEG-7 is a first attempt towards creation of a generic standard for multimedia content description. Since February 2002, it has been accepted as an International Standard, a content description normative for multimedia information. This standard defines the common interface for describing multimedia metadata, and is achieved by specifying a set of Descriptors (D), Description Schemes (DSs), and the Description Definition language (DDL). A Descriptor represents a feature that characterizes the multimedia content. A description scheme details the relationships between descriptors and other description schemes. The description definition language is used to specify the description schemes and descriptors, allowing the extension and modification of existing Description Schemes.

We have previously shown the efficacy of using MPEG-7 descriptors in conjunction with XML database technology in a synergistic fashion for region labeling and retrieval [6]. The application environment, aerial images, was quite interesting in that is exhibits dynamically emerging semantics which could be identified through our approach. We are currently studying how to embed the present work in an MPEG-7 environment.

References

1. Santini, S., Gupta, A., Jain, R.: Emergent Semantics Through Interaction in Image Databases, IEEE Transactions on Knowledge and Data Engineering (2001) 337-351
2. Natsev, A., Rastogi, R., Shim, K.: Walrus: A Similarity retrieval Algorithm for Image Databases, Proceedings of the ACM SIGMOD International Conference on Management of Data (1999) 395-406
3. Sreenath, D.V., Grosky, W.I., Fotouhi, F.: Using Coherent Semantic Subpaths to Derive Emergent Semantics, Proceedings of the Eighth International conference on Knowledge-Based Intelligent Information and Engineering Systems, Lecture Notes in Computer Science, Volume 3215, Wellington, New Zealand (2004), 173-179
4. Grosky, W.I., Sreenath, D.V.: Emergent Document Semantics, Proceedings of the Third IEEE International Symposium on Image and Signal Processing and Analysis, Rome, Italy (2003), 266-271
5. Zhao, R., Grosky, W.I.: Narrowing the Semantic Gap – Improved Text-Based Web Document Retrieval Using Visual Features, IEEE Transactions on Multimedia (2002), 189-200
6. Grosky, W.I., Patel, N., Li, X., Fotouhi, F.: Dynamically Emerging Semantics in an MPEG-7 Image Database, Submitted for publication

Storing and Retrieving Multimedia Web Notes

Paolo Bottoni, Roberta Civica, Stefano Levialdi,
Laura Orso, Emanuele Panizzi, and Rosa Trinchese

Department of Computer Science – University of Rome “La Sapienza”,
Via Salaria 113, 00198, Rome, Italy
{bottoni, levialdi, panizzi, trinchese}@di.uniroma1.it
rcivica@libero.it, laurorso@tin.it

Abstract. Digital annotation of multimedia documents adds personal information to a document (e.g. a web page) or parts of it (a multimedia object such as an image or a video stream contained in the document). Digital annotations can be kept private or shared among different users over the internet, allowing discussions and cooperative work. We discuss some architectural solutions and storage schemas, to the problem of annotating multimedia documents with objects which are in turn of multimedial nature. Annotations can refer to whole documents or single portions thereof, as usual, but also to multi-objects, i.e. groups of objects contained in a single document. We present a new digital annotation system organized in a client-server architecture, where the client is a plug-in for a standard web browser and the servers are repositories of annotations to which different clients can login. Annotations can be retrieved and filtered, based on their metadata descriptors, and possibly on their content, and one can choose different annotation servers for a document. An implementation integrated in a widely used Web browser is discussed.

1 Introduction

As the amount of multimedia material stored on the Web, or held in repositories of corporate organizations, keeps increasing, the problem arises of devising efficient methods for retrieving and employing it in the context of complex activities. Examples of such activities might be performing collaborative work on line, where one might want to use images or videos to illustrate some ideas, discussing features of objects of common interest in a community of practice, exchanging comments on video material in a virtual community.

From an abstract point of view, all of these might be seen as forms of annotation of digital content, where an annotation refers either to the whole document at hand, or to segments of it, which must be clearly identified.

The possibility of referring to concrete objects provides a better grounding of such discussions with respect to forums, or newsgroups, where discussions are very soon led astray due to the lack of a focusing object. The possibility of sharing annotations is of obvious advantage as a support for cooperative work, where forms of discussion on shared objects are already in use [13].

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 119–137, 2005.

With respect to the traditional notion of annotation, referring to an activity where a physical (paper) document was modified or associated with other documents of the same nature, to be circulated within a restricted organization, or held privately by its owner, the possibility of digitally annotating digital documents introduces three important novelties:

1. The original document can be left unaltered and the content of the annotation can be stored in a different document, referring back to the original one, which can also be used independently.
2. The document(s) resulting from the annotation activity can still be held privately or with restricted circulation within an organization, but can also enjoy different levels of publicity, up to global availability over the WWW, always allowing access to the original annotated document.
3. The content of the annotation is not restricted to text, but can include any form of multimedia material, thus becoming a new multimedia document, subject in turn to the annotation process.

The traditional activities of annotating paper-based material and consulting the annotations are normally performed without disrupting the main working activity, but, when users consult digital documents, they find themselves in a different situation. Users follow links, read, observe, or hear content, occasionally fill in forms (for example to perform search), and if something interesting comes in their way, they add URLs to their bookmarks. No real support is given by the current web browsers to annotate documents, nor to place the content of the navigation in the context of the user activity, knowledge, purpose in retrieving content.

Current systems for producing and retrieving annotations in digital form are instead mainly based on dedicated environments, a pioneering initiative under this respect being Annotea [1]. Hence, users must either redefine their established ways of working and learn to integrate their work in new environments, or alternate between their traditional work environment and the annotation one, in order to take advantage of the possibilities of an annotation technology.

We feel, on the contrary, that annotation will become a common activity in future styles of navigation, generating a wealth of new content. For this reason, tools supporting annotation must be smoothly integrated into existing browsers, allowing a user to effortlessly switch between the *navigating* and *annotating* interaction modalities of with the Web content.

We designed and developed a new digital annotation system, called MADCOW (from the acronym of Multimedia Annotation of Digital Content Over the Web), which is currently available for free download[1]. MADCOW is organized in a client-server architecture, where the client is a plug-in for a standard web browser and the servers are repositories of annotations to which clients can login.

The theoretical base for our approach has been presented in [10]. In our view, the annotated document is a multimedia one: a web page including text, images,

[1] http://hci.uniroma1.it/madcowhome/

audio and video files; moreover, the web note itself can be a multimedia object as well, being composed, for example, by text and images; since it can be displayed as a web page, it can be annotated in turn. Finally, web notes can be private or public, i.e. shared. As the design of MADCOW separates platform-independent from platform specific components, it can be implemented as an integration to different browsers (on the client side) and different web servers (on the server one). We started by implementing a MSIE plug-in as client and a PHP/MySQL application for the server side.

In the following sections we describe the background in Section 2, the formal model in Section 3 and the corresponding XML schema in Section 4. The client-server architecture of MADCOW is described in Section 5; Section 6 describes its usage, while Section 7 draws conclusions.

2 Related Work

In this section we synthetically report on the current efforts to allow interactive production and retrieval of annotations over the Web. We shortly describe their features, and summarize the differences with our approach.

- `AnnoteImage` is an authoring tool which allows users to create and publish on the Web own atlases about annotated medical images [15, 11]. In this Java application, annotation data is stored in text files using a LISP-like syntax, and atlases are acceded over the Web via CGI scripts.
- `PAIS` improves on AnnoteImage by using Java applets rather than CGI scripts [16, 14]. Besides, annotation data is stored using the Image Markup Language 1.0 [17], an XML Schema which describes image metadata and annotations.
- `I2Cnet` is a server network containing medical annotated images [12]. It allows doctors to create, read, search and locally save annotations on images, as well as communicate them via e-mail to other doctors in read-only HTML format or as "annotation objects" readable in the I2Cnet environment. Annotation objects, exported in HTML format, can also be posted on a I2Cnet server after a moderator's review, but, after the upload, they can no longer be modified, not even by their author. The user interface is a Java applet connected to an I2Cnet daemon. Annotations data are stored in an ASCII file.
- `Inote` is a standalone Java application that allows users to create annotations on images and saving annotation data on XML files [8].
- `Vannotea` is a prototype system whose major components are search and retrieval database, annotation database, application server and MPEG-2 streaming [20]. The model used for storing metadata is a simple application profile which combines Dublin Core [2] and MPEG-7 [3] standards. Furthermore, Vannotea is based on Annotea [1] so it uses an RDF-based annotation schema and Xpointer to link the annotations to the document.
- `MRAS` [7], the Microsoft Research Annotation System, is a system composed by a server and a client that communicate via HTTP. Moreover, it

is possible to compose outgoing e-mails from annotation data and to process incoming e-mail from the server.
- `VideoAnnEx` [4] has been released by IBM in July 2002. This tool allows the user to annotate MPEG-1 or MPEG-2 video files. Annotations are stored using MPEG-7 descriptions in XML files. The tool can also open MPEG-7 files in order to display the annotation for the corresponding video file.
- `Madeus` [19] is an authoring environment for multimedia documents developed to analyze video structures. It saves information in an XML format, using a proprietary DTD, and composes videos with other document elements.

The tools described here are not integrated in existing browsers, but are standalone applications, dedicated browsers or authoring and viewing tools for creating, uploading and downloading annotations on remote repositories. Thus, when users need to produce annotations on an image or video contained in a Web page, the multimedia object cannot be directly selected from the Web page, but its URI has to be specified. Hence, contrary to what is made possible by our implementation as a browser toolbar, the generated annotation is unrelated to the original web page that contained the media object, possibly creating a loss of context for subsequent users of the annotation. Moreover, the literature concerning these tools, while employing external servers, does not mention the possibility of multiple servers hosting annotations, nor the management of distributed annotations concerning the same document, all features which are offered by our approach. Finally, these tools are generally devoted to a single specific type of document, eg. a video, an image, a text, whereas MADCOW allows a fully integrated management of multimedia documents.

3 The Formal Model

Annotations occur on digital objects, where a digital object o is a typed tuple of attribute-value pairs: $o = typeName((attr_1;\, val_1),\, \ldots,\, (attr_n;\, val_n))$. The *type name* denotes the category an object belongs to: it is a string such as *program*, *file*, *image*, *annotation*, and the like.

A document can thus be made up of several objects specified by a *content* attribute. Examples of content are *text*, *hyperlinks*, *table structures*, *form fields*, *images*, *audio* and *video* files. Formally, the actual content of a digital object is specified by a possibly empty function $f : S \rightarrow V$. The *support* S of the object is a subset of a Cartesian product formed with initial segments of the integer numbers; the vocabulary V, on which the value of **contents** is formed, is a finite set of symbols. The content can in general be considered as some sentence from a language, and it can be formed according to some syntax or semantics. The vocabulary can consist of different types of symbols according to the object type: characters for text, colors for images, image sequences for movie clips, and so on. A vocabulary may mix symbols of different kinds (as in multimedia objects), or may include symbols which are interpreted as references to other objects (e.g. a web page may contain links to image, sound, or movie objects).

In this paper we only refer to web-based annotation, so that the content of a document is considered as a web page identified by a URL.

A *digital annotation* (or simply annotation) is an object a of type *annotation*, related to (portions of) the content of one or more objects. Each portion is defined as a *structure*, i.e. a restriction of the f function defining the content to a subset $E \subseteq S$ of its support. E is called the *support* of the structure. As for any other object, an annotation can be defined by functions on different supports and vocabularies, i.e. it can be an image, or text, etc, so that an annotation can be annotated in its turn.

By viewing a multimedia object as a collection of digital objects connected by some synchronization constraint on their rendering, we allow the construction of annotations on simultaneous views of the related objects. Formally, we introduce the notion of *multistructure* (**MS**) on a *multiobject* (**MO**), where a multiobject MO is an indexed set of objects $\{o_1, \ldots, o_n\}$. A **MS** is an ordered sequence of **cs**s belonging to one or more objects, i.e. an indexed family of function restrictions $\{f'_{1,1}, \ldots, f'_{m,n}\}$, where $f'_{j,i}$ is the j-th restriction of the content function of the i-th object.

4 An XML Schema for Multimedia Annotation

The formal model above is reflected in the XML Schema that we have devised for creating and storing annotations in MADCOW A general view of the schema is presented in Figure 1.

In the figures, we have used the conventions of XMLSpy, an XML editor integrated with Visual Studio .NET. Here complex types are connected to their components through nodes whose iconic aspect denotes the composition construct used. In particular dots indicate a sequence of elements and the open switch indicates selection from a set of choices. Areas enclosed within a dashed line indicate that their presence is optional (i.e. the minimum number of occurrences is 0, the maximum 1).

Under this schema a digital annotation consists of two main components: the *metadata* and the *annotationBody.*

The metadata define information, which is either automatically created or introduced by the user, or selected from a list of options relating to the annotation as a document (i.e. author, title, dates of creation and last modification, type of annotation and visibility), together with a reference, held in the tag `sourceHttp`, to the original document being annotated. The two metadata `type` and `visibility` are peculiar to our approach and can be described as follows:

- The type of the annotation is one from an enumeration of values which describe the different functional roles that the annotation plays with respect to the annotated object. We follow here the definitions of the Rhetorical Structure Theory (RST) [18], a descriptive theory of the organization of natural language texts, which identifies in each text some nucleus (defining the main concept in the text) and some satellites which do not have semantic autonomy, but can be understood only with reference to the nucleus and

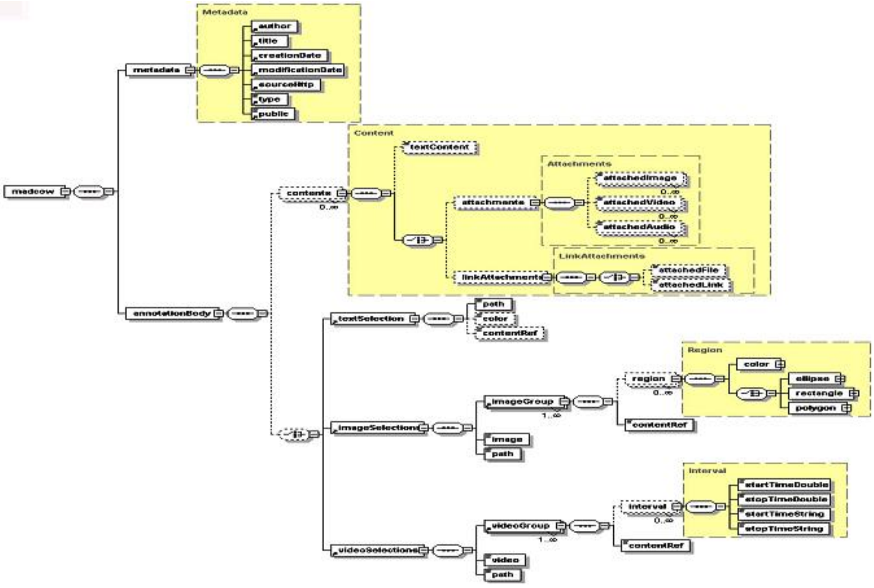


Fig. 1. The general XML Schema for MADCOW annotations

clarify its meaning. When applied to the domain of annotations, it is easy to consider annotations as some form of satellite. In particular, we have identified the following: *explanation*, *comment*, *question*, *integration*, *example*, *summary*, *solution*, *announcement* and *memorandum*, as the possible values of the `Type` attribute.

– Concerning `visibility`, an annotation can be either public or private. The latter are for personal use, while the former are shareable among different users. Public annotations are stored remotely and can be downloaded by anyone having the proper access rights. The annotation author has higher privileges over a public annotation, being the only one allowed to modify or delete it. Forms of group ownership are also devised, so that modifications of existing documents are permitted only to members related to the document author, allowing members to cooperate on the production of an annotation.

These pieces of information are maintained according to standard definitions, taken from the Dublin Core or from the XSL native types. In particular, the author, title, dates and source attributes take values in the domains defined by the Dublin Core elements *creator*, *title*, *date*, and *source*, respectively.

The annotation body associates the annotation content with the indication of the exact location of the (multi)structure for which the annotation was created. It is therefore structured into two elements:

1. the content, which is in turn composed of the text introduced by the annotator together with (optionally) either an unbounded list of links to URIs for multimedia documents (`attachedLink`), or the URI of some local resource (`attachedFile`), and

2. the selection, i.e. the actual multistructure the annotation is associated with. This selection can be a portion of text, a group of regions (structures) in an image, or a group of intervals in a video (or audio).

The multimedia documents which can be attached to an annotation are images, video or audio files. These will be presented in a HTML page whenever the annotation they are contained in is activated by the user (see Section 5). On the contrary, the link attachments do not open an annotation file, but define an active zone in the document. If such a zone is selected, the document indicated by the URI for the `attachedFile` or `attachedLink` elements is opened in a new suitable window.

Selections are described according to their type.

- A text region is defined by the path identifying it (using a string which complies to the `XPointer` standard, so that it can be used for HTML files), plus the colour for its highlighting and a string which is a reference to the attached element possibly indicated in the content. The reference is empty if the annotation does not have associated content, as is the case for simple selection highlighting, or for private marks.
- An image region is interactively drawn, so that the associated information is its perimeter and the colour in which this has been drawn. According to the performed interaction, the actual geometry of the selection is given in different ways. As shown in Figure 2, if the selection is drawn as a rectangle or an ellipse, we use a type `regular`, thus specifying the (x, y) coordinates of the upper left corner, together with the height and width of the enclosing rectangle. All of these data are mandatory and are of type float. If the selection is drawn as a polygon, then it is stored as a list of (x, y) float coordinates. Moreover, each region group is possibly associated with some attached document, activated when the user selects the corresponding region.
- A video (or audio) interval is defined by its start and end time (two values of type `double`) with respect to the start time of the video and by two strings which describe the same values (these are used to produce a SAMI[2]-compliant description of the interval) (see Figure 3). Again, each interval group is associated with some attached document, activated when the video stream has reached the starting time of the interval and deactivated at the ending time.

5 The MADCOW System

MADCOW is based on a distributed architecture, in which document servers and annotation servers are present, and independent of one another. A document server is indeed any website able to deliver content identified by a URI.

[2] SAMI is an XML-based, SMILE-like, Microsoft proprietary language for the description of multimedia content

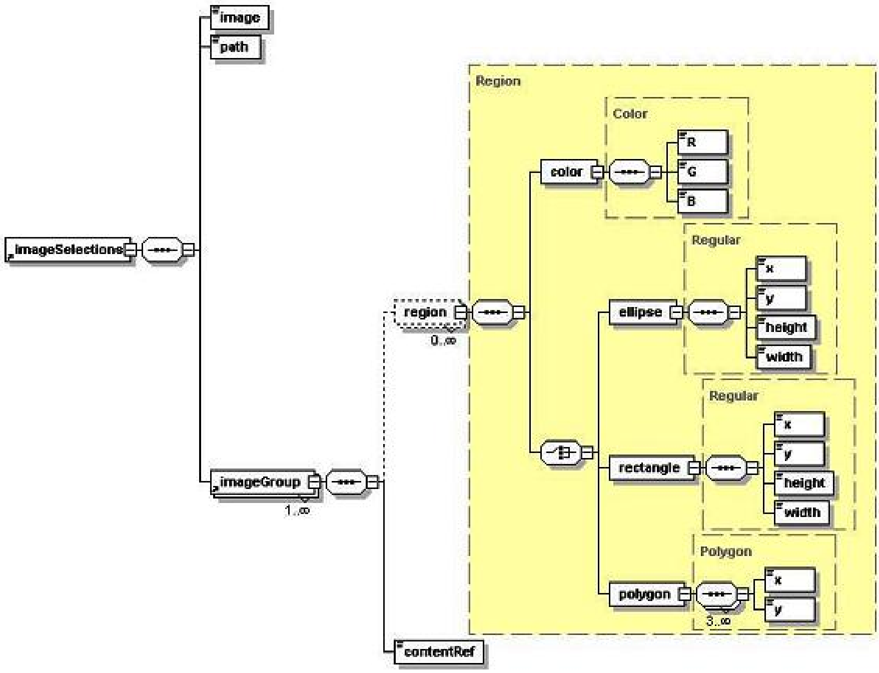


Fig. 2. A detail of the XML Schema for image annotation

MADCOW has been designed as consisting of a server side and a client side. The former contains a database of web notes related to documents; the latter allows downloading of web notes from the servers, exploration of their content and production and storage of new annotations. The user can decide to operate on a particular server or on all servers containing annotations.

The annotation servers are accessible via the HTTP protocol. The client connects to the server in order to retrieve and download annotations as well as to upload newly created ones. The server can ask for a login, and can provide only those annotations that the user is enabled to access according to a privilege policy.

5.1 Server Architecture

A MADCOW annotation server is identified by a URI, and is composed by a set of scripts (eg. JSP or PHP or ASP scripts) that offer several annotation-related services via the HTTP GET and POST methods. The annotation server uses a database to store annotations and to retrieve them based on the URL of the document they refer to, or based on a search on any other attribute (annotation metadata), e.g. the type of annotation. The database schema is defined upon the Annotation XML Schema.

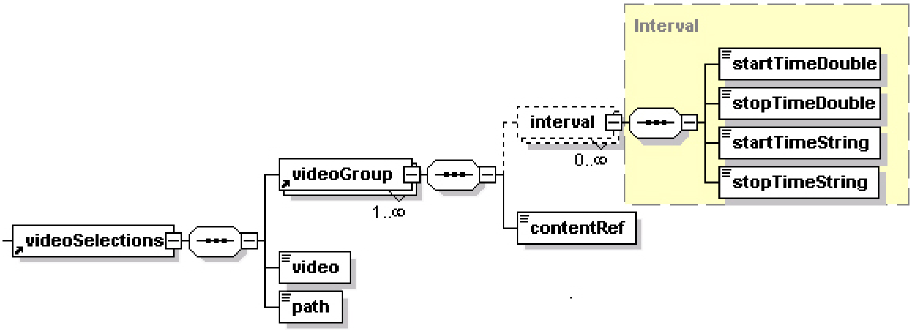


Fig. 3. A detail of the XML Schema for video annotation

Each annotation in the server is uniquely identified by an annotation ID and is directly accessible using a unique URI. For each annotation, the three fields representing the ID, the type and the annotation location within the document are called *annotation description*, and constitute the necessary information for MADCOW to display placeholders, i.e. icons that are used as clickable access points to the annotations. The annotation location refers to the original web page exploiting the XML Pointer Language (XPointer) [6], which allows the identification of fragments for any URI reference that locates a resource. It is based on XPath [5], an XML Language that models a HTML document as a tree of nodes, and allows addressing of portions of documents.

The set of services that a MADCOW annotation server provides is the following:

1. storage of a newly created annotation
2. update of an existing annotation
3. retrieval of all the annotation descriptions related to a specific document
4. retrieval of all the descriptions of the annotations that match one or more search criteria
5. retrieval of the URIs of all the documents to which the annotations in the server are related

When the user produces a new annotation, the MADCOW client formats the metadata and content in an XML file according to the described XMLSchema, and sends it to the annotation server using the HTTP POST method. The server saves the annotation in its database.

Annotation Server Network. MADCOW can use different servers to store and retrieve web notes. In fact, it manages a list of server URIs, which comes filled with a starting set of servers, and can be extended both manually and automatically.

In fact, users may know about the existence of annotation servers that are publicized on web sites or in other ways, so that they may add the respective URIs to the server list. Moreover, MADCOW can find out a new annotation server

URI and add it automatically to its list. It could discover a new annotation server in two ways:

- by reading the default annotation server suggested by a document: this can be optionally specified by the document author in a HTML META tag.
- by using a search engine for retrieving URIs of servers that hold annotations about the current document.

Search engines can access annotation servers when crawling the web as they do for ordinary web sites, and can use the appropriate service (see above) to retrieve URIs of documents annotated in each annotation server.

5.2 The MADCOW Client

The MADCOW client is designed as a plug-in for existing browsers (see Figure 4), so that it can be implemented in different versions, in order to support different communities of users. When the user loads a web page in the browser, the MADCOW client looks for web notes related to that page in known servers as well as in its private annotation storage. Each server responds with an XML file containing annotation descriptions. The client uses them in order to display the placeholders in the document.

If the user clicks on a placeholder icon, MADCOW downloads the corresponding web note from the server, by using the ID contained in the description. The multimedia content of the web note is loaded and shown in a new browser window. As the web note can contain audio, video, image or text documents, suitable browser plugins are automatically activated to reproduce the multimedia content of the annotation, as well as to show its attributes.

The MADCOW client also allows users to create new annotations by selecting the object(s) to be annotated and choosing the proper entry in the system menu. In this case, a new window opens, whereby the annotation author can add textual content, images, video and audio files. The user is asked to decide whether the annotation should be private (accessible only to him or her) or public.

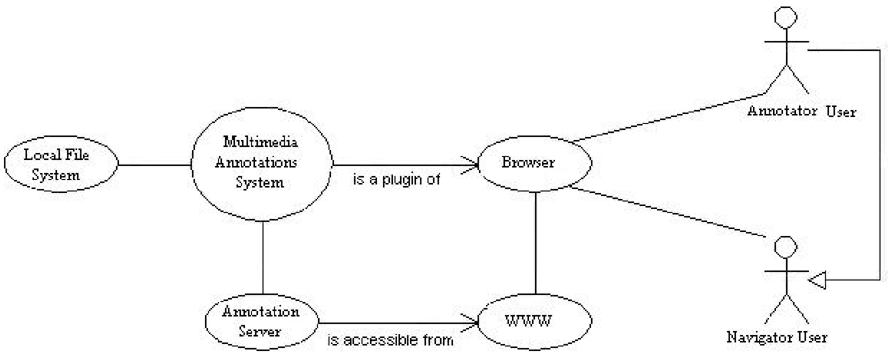


Fig. 4. MADCOW context diagram

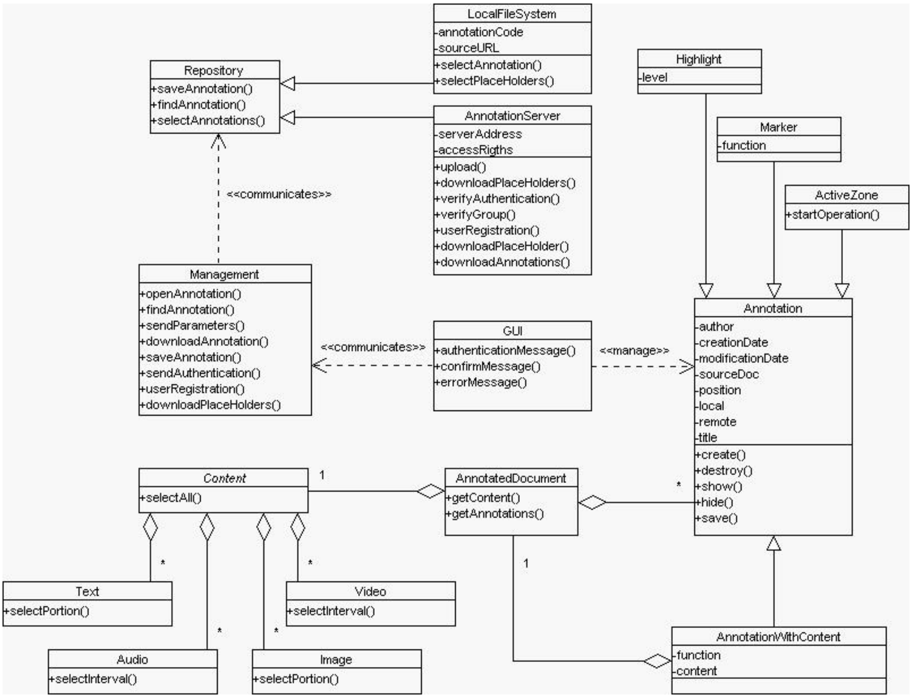


Fig. 5. The conceptual architecture of MADCOW

Classes. MADCOW conceptual architecture is described by the Class Diagram of Figure 5.

The GUI class provides the user with an access to the Annotation and Management classes.

The first contains the (metadata) attributes, as well as the methods that manage the annotation life cycle. Annotation is an abstract class to be specialized in one of its concrete types: Highlight, Mark, Active Zone and Annotation with Content. `Annotation` provides the basic structure for the management of the location attribute, in the form of a list of XPointers. `Highlight` provides support for the choice of the style of highlighting. `Mark` additionally allows the selection of the type of mark, which can also be of user's own creation, not related to any specific type from the RST. The `ActiveZone` class allows activities to be started when the user interacts with structures defined by some annotation actions. Finally, `AnnotationWithContent` allows an annotation to embody a document that can be annotated in turn. The content of an annotation being a digital document, it consists of Text, Video, Image, Audio or any combination of these.

The `Management` class contains the client side methods that allow retrieving (saving) of annotations from (to) the repository. The storage can be a local directory on the user's computer, in which case it is managed by the `LocalFileSystem` class, or it can be any server from the client's list, and managed by the `AnnotationServer`.

6 The MADCOW Implementation and Usage

The current implementation of the MADCOW client is a plug-in for the browser Internet Explorer; the MADCOW Annotation Toolbar, visible under the browser address bar, allows the user to manage annotations. In the Annotation Toolbar a menu and ten buttons are present (see Figure 6). The menu, marked with the MADCOW name and the small arrow, contains configuration and personalization functions. The first button starting from the left allow users to create annotations with content. The next four buttons allow the user to create a marker (an annotation without any content but the title), an active zone (an annotation that consists in a hyperlink pointing to another resource), a multistructure[3] (a normal annotation that refer to more than one selection in the document) and a highlight annotation (a background colour to highlight a portion of text) respectively. The sixth and seventh allow users to modify and delete an annotation. The eighth displays all the placeholders available for the current web page. Finally the last two buttons open a search and a filter annotation window respectively.

In the following subsections we will describe the activities needed to create and access annotations.

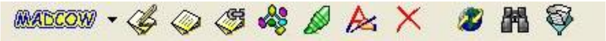


Fig. 6. Buttons of the MADCOWToolbar

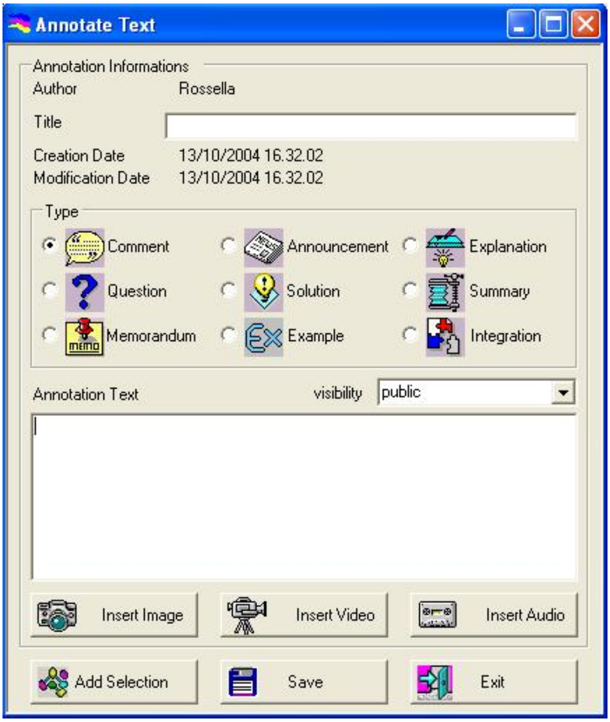
6.1 Create a Web Note

When users browse the web and want to create an annotation, they first select a region in the web page and then click on one of the first four buttons in the toolbar. The MADCOW client detects the selection format (text, image or audio/video) as well as the selected type (annotation with content, mark, active zone or highlight) and opens up a dialog window with the proper creation interface.

Annotate Text. The annotate text dialog window (see Figure 7) shows up when the user selects a portion of text and clicks the first button of the annotation toolbar. The user specifies all the metadata that describe the annotation (title of annotation, RST type, public or private visibility) and annotation content: text and optional image or video/audio files. Some fields (such as creation date, modification date and author name) are automatically filled in by the system. When the user saves the annotation, the icon corresponding to the selected type is inserted as a placeholder in the web page, as shown in Figure 12.

Annotate Image. The dialog window of Annotate Image (see Figure 8) allows the user to annotate the selected image. The image annotation window is similar to the textual annotation one in the right side, while, in the left side, it contains

[3] Not supported by the current Madcow client implementation.



Annotate Text

Annotation Informations

Author Rossella

Title

Creation Date 13/10/2004 16.32.02

Modification Date 13/10/2004 16.32.02

Type

● Comment ○ Announcement ○ Explanation

○ Question ○ Solution ○ Summary

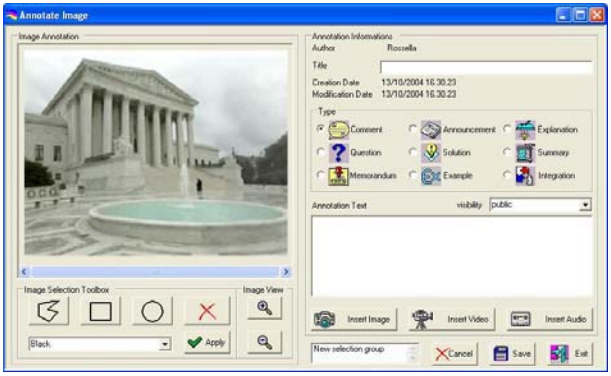
○ Memorandum ○ Example ○ Integration

Annotation Text visibility public

Fig. 7. Annotate Text dialog window

the image the user wants to annotate, on the top, and the toolbox to select different portions of the image, on the bottom. With the toolbox the user can select a shape (e.g. circle, rectangle or polygon), as well as the line colour of the shape. The user can also modify or delete the region selected and zoom in or out. On the right side, differently from the text annotation window, there is the list of all regions defined by the annotation. In order to annotate an image, the user selects it on the original document, clicks on the create button, fills in the attributes in the window right side, and selects some regions on the image. For any region, users can insert some text, attach files and click on the Apply button. These operations can be repeated to define and annotate other regions until the user saves the annotation.

Annotate Video/Audio. The video annotation dialog window contains in the left side the video being annotated and the tools to select different video intervals (see figure 9). In this case, users can play the video and click on the start and stop buttons to select an interval: the video is paused and users can write a comment related to that interval and attach any multimedia file. By clicking on the apply button, the comment is inserted into the listbox and video playing is resumed. Users can repeat these operations to select other intervals and, at the end, save the annotation.



Annotate Image

Annotation Informations

Author: Rossella

Title:

Creation Date: 13/10/2004 16.30.23

Modification Date: 13/10/2004 16.30.23

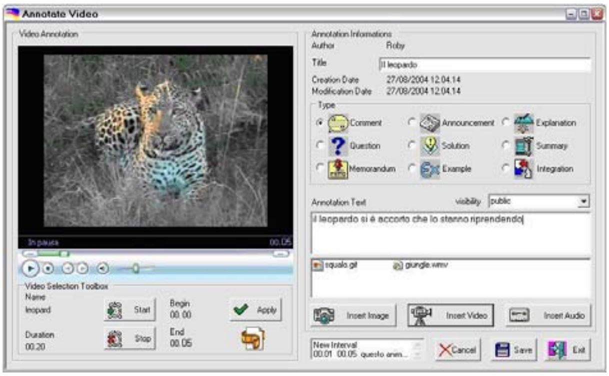
Type: ● Comment

visibility: public

Black

New selection group

Fig. 8. Annotate Image dialog window



Annotate Video

Annotation Informations

Author: Roby

Title: Il leopardo

Creation Date: 27/08/2004 12.04.14

Modification Date: 27/08/2004 12.04.14

Type: ● Comment

visibility: public

Annotation Text: il leopardo si è accorto che lo stanno riprendendo

squalo.gif giungle.wmv

In pausa 00.05

Name: leopard

Duration: 00.20

Begin: 00.00

End: 00.05

New Interval
00.01 00.05 questo anim...

Fig. 9. Annotation on the video

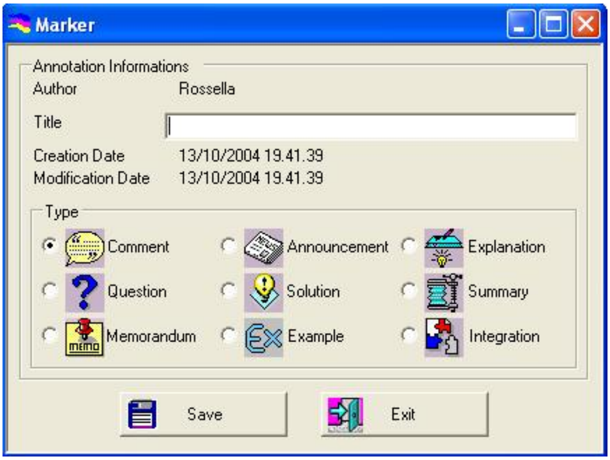
Marker. The marker annotation consists only of a graphical icon inserted into the annotated document. The user can insert only title and type (see figure 10).

Active Zone. This kind of annotation allows users to insert a link to an existing web page or to a local file that is then uploaded to the annotation server (see figure 11).

Highlight. To create a highlight annotation, users simply select a portion of text and click on the highlight button. No metadata specification is required. Highlights are always private.

6.2 Web Notes Access

MADCOW users can access any public annotation in different ways: they can have all annotation placeholders for the current web page displayed or filtered to display only a subset, open a single annotation, and search for annotations that satisfy some parameters.



Marker

Annotation Informations

Author Rossella

Title

Creation Date 13/10/2004 19.41.39

Modification Date 13/10/2004 19.41.39

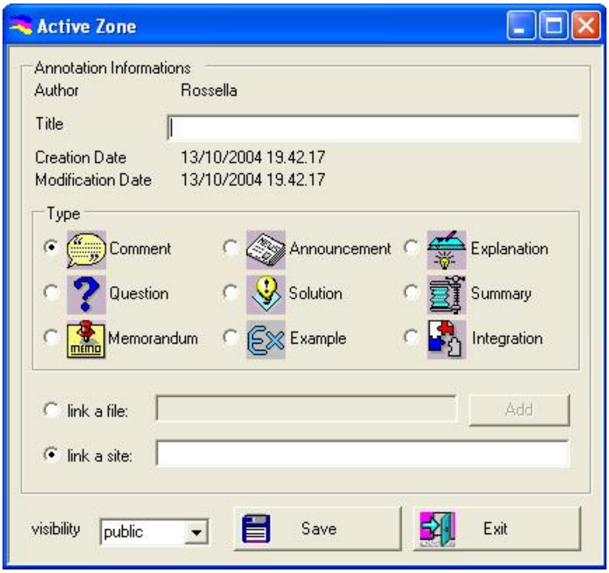
Type

● Comment ○ Announcement ○ Explanation

○ Question ○ Solution ○ Summary

○ Memorandum ○ Example ○ Integration

Fig. 10. Marker dialog window



Active Zone

Annotation Informations

Author Rossella

Title

Creation Date 13/10/2004 19.42.17

Modification Date 13/10/2004 19.42.17

Type

● Comment ○ Announcement ○ Explanation

○ Question ○ Solution ○ Summary

○ Memorandum ○ Example ○ Integration

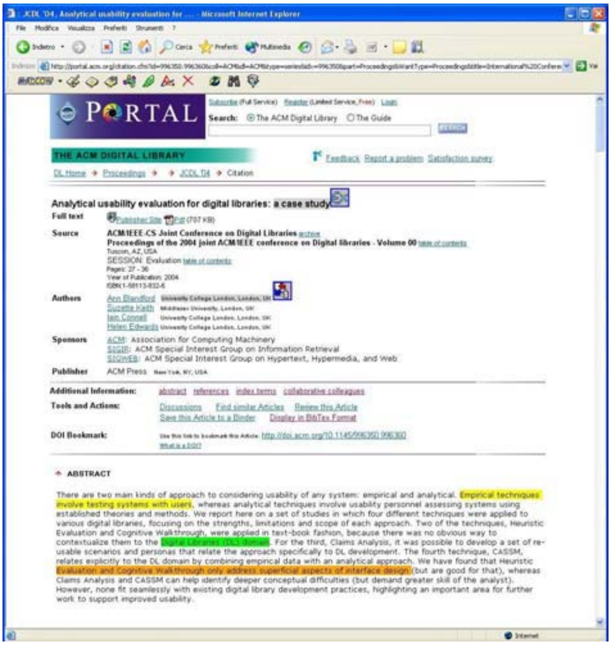
○ link a file:

● link a site:

visibility public

Fig. 11. Active Zone dialog window

Placeholders. Upon opening a web page, users can visualize all annotations by simply clicking on the download placeholder button of the annotation toolbar. Each placeholder is represented by an icon corresponding to the type of the annotation (see figure 12). Moving the mouse over a placeholder results in showing a tooltip describing author and title of the annotation. Clicking on the placeholder, on the other hand, has different results according to the kind of annotation: normal annotations are opened in a new browser window; active zones show in a new window the linked page or the user file; finally markers' placeholders do not allow clicking. A MADCOW option allows the automatical download of placeholders of the browsed Web pages.


: JCDL '04, Analytical usability evaluation for ... - Microsoft Internet Explorer
File Modifica Visualizza Preferiti Strumenti ?
Indietro
Cerca
Preferiti
Multimedia
Indirizzo
http://portal.acm.org/citation.cfm?id=996350.996360&coll=ACM&dl=ACM&type=series&idx=996350&part=Proceedings&WantType=Proceedings&title=International%20Confere
Vai
MADCOW
PORTAL
Subscribe (Full Service) Register (Limited Service, Free) Login
Search: ◉ The ACM Digital Library ○ The Guide
SEARCH
THE ACM DIGITAL LIBRARY
Feedback Report a problem Satisfaction survey
DL Home → Proceedings → → JCDL '04 → Citation
Internet

Analytical usability evaluation for digital libraries: a case study

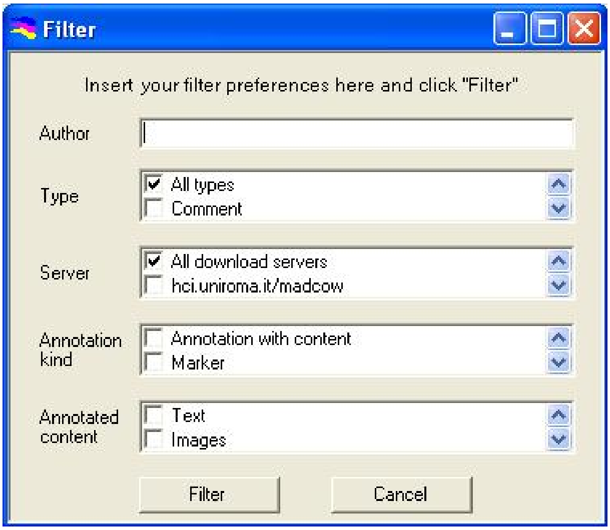
Full text	Publisher Site Pdf (707 KB)
Source	ACM/IEEE-CS Joint Conference on Digital Libraries archive Proceedings of the 2004 joint ACM/IEEE conference on Digital libraries - Volume 00 table of contents Tuscon, AZ, USA SESSION: Evaluation table of contents Pages: 27 - 36 Year of Publication: 2004 ISBN:1-58113-832-6
Authors	Ann Blandford University College London, London, UK Suzette Keith Middlesex University, London, UK Iain Connell University College London, London, UK Helen Edwards University College London, London, UK
Sponsors	ACM: Association for Computing Machinery SIGIR: ACM Special Interest Group on Information Retrieval SIGWEB: ACM Special Interest Group on Hypertext, Hypermedia, and Web
Publisher	ACM Press New York, NY, USA

Additional Information:	abstract references index terms collaborative colleagues
Tools and Actions:	Discussions Find similar Articles Review this Article Save this Article to a Binder Display in BibTex Format
DOI Bookmark:	Use this link to bookmark this Article: http://doi.acm.org/10.1145/996350.996360 What is a DOI?

ABSTRACT

There are two main kinds of approach to considering usability of any system: empirical and analytical. Empirical techniques involve testing systems with users, whereas analytical techniques involve usability personnel assessing systems using established theories and methods. We report here on a set of studies in which four different techniques were applied to various digital libraries, focusing on the strengths, limitations and scope of each approach. Two of the techniques, Heuristic Evaluation and Cognitive Walkthrough, were applied in text-book fashion, because there was no obvious way to contextualize them to the Digital Libraries (DL) domain. For the third, Claims Analysis, it was possible to develop a set of re-usable scenarios and personas that relate the approach specifically to DL development. The fourth technique, CASSM, relates explicitly to the DL domain by combining empirical data with an analytical approach. We have found that Heuristic Evaluation and Cognitive Walkthrough only address superficial aspects of interface design (but are good for that), whereas Claims Analysis and CASSM can help identify deeper conceptual difficulties (but demand greater skill of the analyst). However, none fit seamlessly with existing digital library development practices, highlighting an important area for further work to support improved usability.

Fig. 12. A page with placeholders deriving from annotation



Insert your filter preferences here and click "Filter"

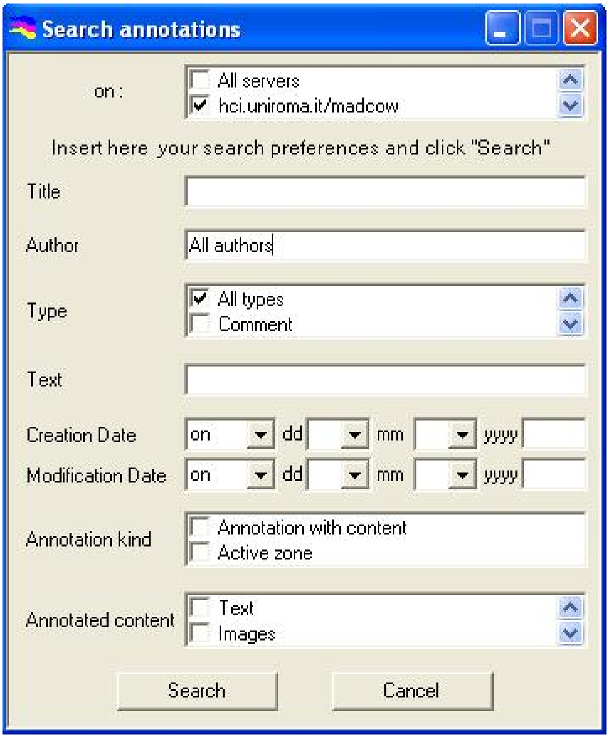
Author	
Type	☑ All types □ Comment
Server	☑ All download servers □ hci.uniroma.it/madcow
Annotation kind	□ Annotation with content □ Marker
Annotated content	□ Text □ Images

Fig. 13. Filter Annotation dialog window

Filter Annotation. Users can filter annotations out by clicking on the filter button and choosing appropriate criteria (see Figure 13). Placeholders of filtered-out annotations will disappear.

Open Annotation. Users can open an annotation by simply clicking on the corresponding placeholder; a new browser window will open up containing the annotation web page. This page is composed of two parts:

- the first part contains all the annotation metadata;
- in the second part, the annotation content is presented as follows:
  1. for textual annotations, the comment of the user and the attached files are presented;
  2. for image annotations, the image is shown with the drawn regions superimposed; these regions are clickable; when the user clicks on a region a comment and the attached files are displayed below the image;
  3. for video annotations, the video appears together with the clickable list of intervals; when the video is started, the comments are shown during the associated intervals; if the user clicks on an item in the interval listbox, the video player moves to the selected interval and the corresponding comment is shown;



Search annotations

on : ☐ All servers
☑ hci.uniroma.it/madcow

Insert here your search preferences and click "Search"

Title

Author: All authors

Type: ☑ All types
☐ Comment

Text

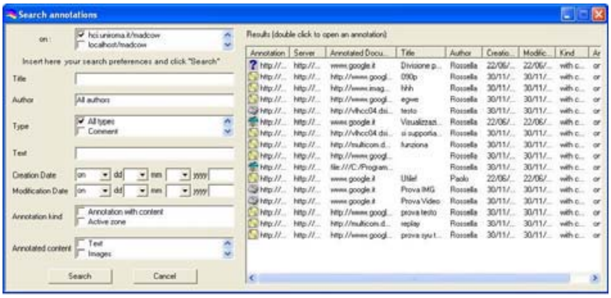
Creation Date: on dd mm yyyy

Modification Date: on dd mm yyyy

Annotation kind: ☐ Annotation with content
☐ Active zone

Annotated content: ☐ Text
☐ Images

Fig. 14. Search Annotation dialog window



on: ☑ hci.uniroma.it/madcow ☐ localhost/madcow

Insert here your search preferences and click "Search"

Title

Author: All authors

Type: ☑ All types ☐ Comment

Text

Creation Date: on dd mm yyyy

Modification Date: on dd mm yyyy

Annotation kind: ☐ Annotation with content ☐ Active zone

Annotated content: ☐ Text ☐ Images

Results (double click to open an annotation)

Annotation	Server	Annotated Docu...	Title	Author	Creatio...	Modific...	Kind	Ar
http://...	http://...	www.google.it	Divisione p...	Rossella	22/06/...	22/06/...	with c...	or
http://...	http://...	http://www.googl...	090p	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	http://www.imag...	hhh	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	http://www.googl...	egwe	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	http://vlhcc04.dsi...	testo	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	www.google.it	Visualizzazi...	Rossella	22/06/...	22/06/...	with c...	or
http://...	http://...	http://vlhcc04.dsi...	si supporta...	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	http://multicom.d...	funziona	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	http://www.googl...		Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	file:///C:/Program...		Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	www.google.it	Utile!	Paolo	22/06/...	22/06/...	with c...	or
http://...	http://...	www.google.it	Prova IMG	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	www.google.it	Prova Video	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	http://www.googl...	prova testo	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	http://multicom.d...	replay	Rossella	30/11/...	30/11/...	with c...	or
http://...	http://...	http://www.googl...	prova syu t...	Rossella	30/11/...	30/11/...	with c...	or

Fig. 15. Search Annotation dialog window result

Search Annotation. MADCOW annotations allows users to search any public annotation created by any other user. In the search annotation dialog window (see Figure 14) some parameters from the metadata appear as selection criteria, e.g. server name, annotation type, annotation kind, etc. When the user clicks on the search button, the window is enlarged with the list of results that satisfy the parameters (see Figure 15). Users can refine the search or open an annotation by double-clicking its entry in the list.

7 Conclusions

The problem of making annotations readily producible and available to users when consulting a Web page included calls for efficient storage, retrieval and interaction management. This is particularly demanding when both the consulted page and the created annotation may include multimedia content, such as images or video. We have defined a notion of digital object which is the basis both for the definition of the content to be annotated and of the content of the annotation. The different types of objects support different types of interactions, but all the portions to be annotated can be described in terms of *regions*, so that a uniform model of annotation can be devised, as the association of digital objects to regions. As different types of digital objects support different definitions of region, specific forms of interaction on the client side have been realized. The resulting web notes are transmitted to and retrieved from the server in the form of XML documents, complying to an XML Schema defining both the metadata to be associated with the annotation and the associations between regions and contents. A distributed network of annotation servers can be realized, in order to separate the management of annotations from that of the annotated documents.

Acknowledgements

This work is funded by the WEB-MINDS project supported by the Italian MIUR under the FIRB program.

References

1. Annotea. http://www.w3.org/2001/Annotea.
2. Dublin core. http://dublincore.org.
3. Mpeg. http://www.mpeg.org.
4. Videoannex annotation tool. http://www.research.ibm.com/VideoAnnEx/.
5. Xpath. http://www.w3c.org/TR/xpath.
6. Xpointer. http://www.w3c.org/XML/Linking.
7. D. Bargeron, A. Gupta, J. Grudin, and E. Sanocki. Annotations for streaming video on the web: System design and usage studies. Technical report, Microsoft Research, 1999. http:// research.microsoft.com/research/coet/MRAS/www8/paper.htm.
8. R. Bingler. Inote image annotation in java. Technical report, University of Virginia, 1998. http://jefferson.village.virginia.edu/inote/.
9. F. Bonifazi, S. Levialdi, P. Rizzo, and R. Trinchese. A web-based annotation tool supporting e-learning. In Proc. AVI '02, pages 123–128. ACM Press, 2002.
10. P. Bottoni, S. Levialdi, and P. Rizzo. An analysis and case study of digital annotation. In N. Bianchi-Berthouze, editor, Proc. DNIS-2003, pages 216–230. Springer, 2003.
11. J. Brinkley, R. Jakobovits, and C. Rosse. An online image management system for anatomy teaching. In Proceedings of the AMIA 2002 Annual Symposium, 2002.
12. C. Chronaki, X. Zabulis, and S. Orphanoudakis. I2cnet medical image annotation service. In Med In-form (Lond), 1997.
13. D. Cox and S. Greenberg. Supporting collaborative interpretation in distributed groupware. In Proc. CSCW 2000, pages 289–298. ACM Press, 2000.
14. W. Lober. Personal annotated image server (pais). Technical report, Washington University, 2000.
15. W. Lober and J. Brinkley. A portable image annotation tool for web-based anatomy atlases. In Proceedings of the AMIA 1999 Annual Symposium, 1999.
16. W. Lober and J. Brinkley. A personal annotated image server. Fellowship F38, LM00086, NLM, 2000.
17. W. Lober, L. Trigg, D. Bliss, and J. Brinkley. Iml: An image markup language. Fellowship F38, LM00086, NLM, 2000.
18. W. Mann and S. Thompson. Rhetorical structure theory: A framework for the analysis of texts. IPRA Papers in Pragmatics, 1:1–21, 1987.
19. C. Roisin, T. Tran Thuong, and L. Villard. Integration of structured video in a multimedia authoring system. In Proceedings of the Eurographics Multimedia '99 Workshop, pages 133–142. Springer Computer Science, 1999. http://opera.inrialpes.fr/people/Tien.Tran-Thuong/Welcome.html Demos and Publication section.
20. R. Schroeter, J. Hunter, and D. Kosovic. Vannotea - a collaborative video indexing, annotation and discussion system for broadband networks. In K-CAP 2003 Workshop on "Knowledge Markup and Semantic Annotation", 2003.
21. P. Bottoni, R. Civica, S. Levialdi, L. Orso, E. Panizzi, and R. Trinchese. MADCOW: a Multimedia Digital Annotation System In AVI 2004 Working Conference on Advanced Visual Interfaces, 2004.

A Searching Method Based on Problem Description and Algorithmic Features

Yutaka Watanobe, Rentaro Yoshioka, and Nikolay N. Mirenkov

University of Aizu, Aizu-Wakamatsu, 965-8580, Japan

Abstract. A searching method related to "what" and "how" problem descriptions within a special software component library is presented. The "what" problem description is based on a high level representation of general features of initial and final data the problem can operate and produce. The "how" problem description is based on another high level representation of computational features of algorithm used for the problem solution. To realize the searching method, a special library in which each item is stored as a cyberFilm is considered. CyberFilm formats are used for representing data/knowledge units as self-explanatory components, and a special double ID is assigned to each component. The double ID is a pair of a URL-like address and a classification code. A cyberFilm represents a software component as a set of algorithmic features that provide a high-level and yet precise description of computation as well as semantics of problem that should be solved by the corresponding algorithm. These features are the basis of the classification code and the searching method.

In this paper, a basic idea of the "what" problem description for the searching method, and an overview of the features for the "how" problem description, are considered. A possible library structure that uses the classification codes to support various searching goals is also described.

1 Introduction

In spite of the many difficulties of programming, a greater population of people is required to be involved in the information technology and the development of information resources for supporting the future growth of our society. Software component libraries have become an important part of software creation. It is extremely rare that we create programs completely from scratch. In the simplest form, we use code fragments created by others as includes and imports, or simply make calls to standard libraries of the programming languages. In a more sophisticated form, we program within the framework of some component-based technologies, such as Microsoft's COM, JavaBeans, OSGI, etc. [5]. These frameworks help in reducing the developer's load related to the tasks of accessing, composing, and executing components. In either case, building new solutions by combining existing components improves quality and supports rapid development. However, it is very difficult for users with limited knowledge to understand and reuse these libraries. Such libraries include a large variety of algorithms. For example, Java class libraries and Standard Template Library [9] provide a wide range of functions from basic data structures and GUI gadgets to high performance algorithms. Library based programming tools, such as LEDA [10] also provides a lot of useful algorithms. Many

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 138–149, 2005.

algorithms have been implemented for special purposes in different fields. Currently, it is up to the user to find an appropriate class or component and discover how to use them. The most common approach in performing this search is reading text and diagrams of thick documents. It is possible to say that we are still facing the productivity problems in programming [6-7]. One possible method for the user to acquire the right components corresponding to his/her needs is retrievals based on a description of problem that they want to solve. In many cases, the users know what to do, but do not know how to do it. This means, the searching method should support problem solving through extracting or predicting algorithmic features from the problem description. Searching operations based on algorithmic features are also important for users to get their target that is more suitable for their problem/application.

In this paper, a searching method based on "what" and "how" problem descriptions within a special software component library is presented. The "what" problem description is based on a high level representation of general features of the initial data the problem can operate on and final data it can produce. The "how" problem description is based on another high level representation of computational features of algorithm used for the problem solution. To realize the searching method, a special library in which each item is stored as a cyberFilm [13] is considered. CyberFilm formats are used for representing data/knowledge units as self-explanatory components [11-13], and a special double ID is assigned to each component. The double ID is a pair of a URL-like address and a classification code. A cyberFilm represents a software component as a set of algorithmic features that provide a high-level and yet precise description of computation as well as semantics of problem that should be solved by the corresponding algorithm. The algorithmic features of a cyberFilm component includes multimedia algorithmic skeletons (AS), variables and arithmetic/logic or multimedia expressions, input/output operations, and integrated view of these three. The semantic features of a cyberFilm component include space and time shapes, multimedia types, and mathematical and/or physical relations of initial and final data. The first four types of features represent the "how" problem description, and the fifth type represents the "what" problem description. These features are a basis for the creation of the classification code and the searching method. Our goal is to provide a set of modules for people that enable them understand, specify their search targets effectively. Currently, the set of modules for computer is based on conventional database system. We don't use abstract model that has limited number of operations and symbols, but use multimedia language that includes open set of icons. SQL-like queries that are implicitly generated by people based on the multimedia language, are send to conventional database system, then it returns result set correspond to the queries. In this paper, a basic idea of the "what" problem description for the searching method, and an overview of the features for the "how" problem description, are considered. A possible library structure that use the classification codes to support various searching goals is also described.

2 Related Works

In addition to papers from the introduction, we would like to mention some other works related to software component search. SPARS-J (Java Software Product Archiving, analyzing and Retrieving System) [3] is a component search system where source codes of classes or interfaces are parsed and ranking methods based on frequency of words to discover relations between various components are used. Agora [4] is also a component search system that shows a useful integration of Web search engines and component introspection. However, these systems work only with the source code format. They provide effective searching of components such as Java classes and JavaBeans. As a rule, many object-oriented language researchers focus on fixing Java's trouble spots or extending the language itself [1]. However, it is difficult to extract additional high-level descriptive information, such as algorithmic features, from them.

We should consider about problem description methods. A catalog of algorithmic problems in Algorithm Design Manual [8] explain each problem with a pair of graphics representing the problem instance or input on the left and the result of solving the problem on this instance on the right. It enables us to understand problem more than just definitions by stylized images and examples. "AlgoVista [2]" is a web-based search engine that assists programmers to find algorithms and implementations that solve specific problems. It is based on input/output samples that describe the behavior of their needed algorithm. It provides attractive searching method with drawing pictures, but well-defined data models for input and output that describe problem description can not be specified.

SUMS [16] that provides a multimedia access to world cultural heritage, is regarding the "what" and the "how" with some strategies. However, the system does not go deep enough for the "what" and "how" semantic understanding, and is not suitable for software component search.

3 The "What" Problem Description

In this section, a basic idea of the searching method based on the "what" problem description, is considered. First, to illustrate the "what" and "how" problem descriptions, let us consider how a problem can be described. Suppose there is a list of customer information that must be sorted according to some criteria. In this case, the "what" part consists of the description of the input data, the format and size of customer information, the criteria, format for ordering the sorted list, and so on. On the other hand, the "how" part consists of the description of the computational method, the algorithm, used to sort the list.

In the "what" problem description, a special multimedia language is used to describe general features of initial data which can be accessible as problem input and general features of result data which can be produced as the problem output. These features include space and time shapes of the data, multimedia types of data units and mathematical and/or physical (application) relations of the data. It is important to note that they can include the representation of relations between data sub-sets of the input set or between data sub-sets of output data set, as well as between sub sets from different sets. The language constructs used for representing these features are employed as "keysymbols" in searching operation for the software components. For example, "sphere" can be a symbol of spatial shapes of initial (result) data, "color" can present values of data units on sphere points, "contour" can show some neighborhood, "gradient" of colors within neighborhood can demonstrate a physical relation, and "pipeline" can define temporal shape of data input (output). Such searching method will enable users to get algorithmic solutions that work or produce data with specific features. It will especially help end-users to start solving their problems. When they need a solution to solve their problems, they might not know any ideas of how to solve them. But they can know some semantic features of their problems. The library support such searching method efficiently, because each library component includes these semantic features. "Keysymbols" presenting these features are also used for generating some sections of the classification code attached to each library component.

4 The "How" Problem Description

In this section, the details of the "how" problem description is considered. It is based on high level representation of computational features of algorithm used for the problem solution. A computational feature can be considered as a set of attributes (multimedia symbols), and we present two main sets of attributes important for the searching operations. The first group is attributes mainly to identify each component and to summarize its algorithmic features. The second group is attributes to describe, how it is related to other components and how it has been used by other users. In other words, the idea is to describe the component from internal and external perspectives to allow a variety of searching ways, thus increasing the possibility of finding a suitable component from a large library for various users. Each attribute takes a part in the creation of a classification code. In this paper, we do not mention how the codes are created, but we focus on the extraction of the internal/external attributes from the algorithmic features, and in next section, we considere a possible library structure that makes efficient use of these attributes and applicability of them.

4.1 Internal Attributes

The first group is a set of internal attributes to uniquely identify and summarize the algorithmic features of a component. The first column of Table 1 depicts the possible list of internal attributes. The attributes are defined by the user through special editors to specify algorithmic features. These attributes are used to produce different sections of the classification code. In this section, details of these internal attributes and how they are implicitly defined by the user in a language of micro-icons, are described.

- **Structure.** A structure or a combination of structures is one of the most interesting and the most likely to be required attribute for searching computational algorithms as well as algorithmic skeletons. The structure is a space shape of computation, and can determines a model of corresponding applications and problems. Figure 1 (a) depicts examples of micro-icons to specify basic structures such as 2-D grids, 3-D grids, trees, graphs, particles in 3D space, and so on. In many cases, these basic icons will be used to specify structures, but also additional icons that represent more specific features of a structure such as additional properties or different layouts, can be used for a variety of applications and user's intentions. Figure 1 (b) shows examples of additional micro-icons to specify constrained structures for more special types of computation, such as circular trees, cone trees, bipartite graphs, complete graphs, and balanced trees.

Table 1. Table for Internal Attributes

Internal Attributes
Structures
Schemes
Masks
Dynamic irregularities
Type of compositions
Mathematical notations
Variables
Stencils
Input/Output structures
Input/Output communication types
Complexity based on AS
Complexity based on full features
Complexity defined by authors
Component title
Author
Date
ID

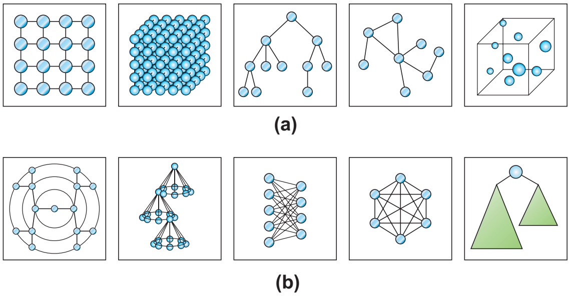


Fig. 1. Examples of micro-icons for structures

- **Mask.** A mask specifies static irregularity on regular structures. In other words, a mask modifies the space structures so that the user can characterize a computational shape in more detail. Figure 2 represents examples of micro-icons for the masks related to 2-D grids and 3-D grids. Both simple and complex geometrical configurations and patterns of irregularity can be represented.
- **Scheme.** A scheme defines computational flow on a structure. It is represented by a partial order of node scanning on the corresponding structure. Figure 3 shows examples of micro-icons for possible computational schemes. This attribute allows to specify sequential or parallel processing, directional characteristics and patterns of computational flows.
- **Dynamic irregularity.** A dynamic irregularity is a set of special masks that predicate masks and add additional levels of irregularity to the computational schemes. In fact,

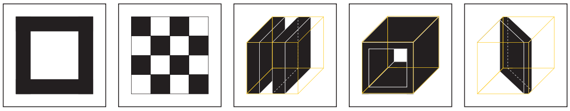


Fig. 2. Examples of micro-icons for masks

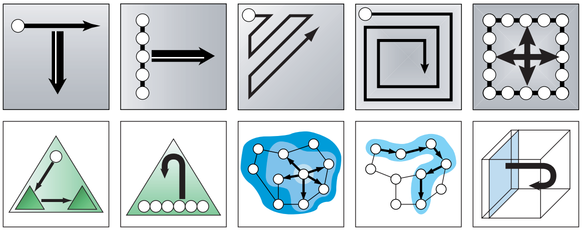


Fig. 3. Examples of micro-icons for computational schemes

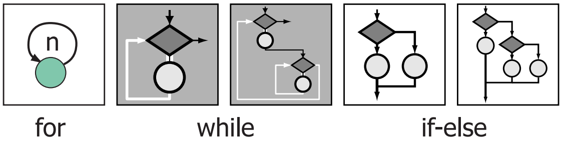


Fig. 4. Examples of Temporal Composition

these masks are applied not to space structures but to computational schemes on the structures. A mask can change a scheme depending on some conditions.

- **Type of Composition and Levels of Hierarchy.** For composite components, a great variety of spatial (for example, shown in Figure 1), temporal, and hierarchical structures, and a combination of spatial and hierarchical structures, are supported. Also, in hierarchical compositions, **levels of the hierarchy** can be important attributes to characterize algorithmic feature of corresponding components. In temporal compositions, special structures for iterative and branching constructs are considered as shown in Figure 4.
- **Mathematical Notations/Functions.** Mathematical notations/functions characterize corresponding software component. There are many possible notations/functions for different kinds of computation, and they can be attributes of the component. Figure 5 shows examples of micro-icons for the mathematical notations/functions.
- **Variable.** A variable is a list of (structure, data type) pairs that declared in the component. For instance, integer, float, string, are typical data type, and the pairs of such data types and structure (shape of the data) can be significant attributes of a software component.
- **Stencil.** A stencil is used as a high-level construct for specifying indexes in formulas [11]. Figure 6 shows examples of micro-icons for stencils. Some time, hints of algorithmic solution can be extracted from this attribute.

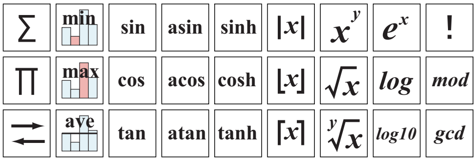


Fig. 5. Examples of micro-icons for mathematical notations/functions

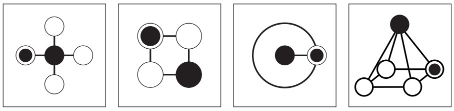


Fig. 6. Examples of micro-icons for stencils

- **Input/Output Structures and Their Data Types.** Input/output structures include 1-D/2-D/3-D grids, trees, pyramid, and other structures. This attribute is a set of pairs (a structure and a data type such as integer, double precision integer, float, double precision float, character, string, etc.). Kinds of I/O structures and data types occurred before, after, and during the process of computation, can be specified by low-level micro icons [14]. In fact, these attributes are related to technical issue for implementation of corresponding component.
- **Complexity Based on Algorithmic Skeletons.** Complexity of algorithmic skeletons can be estimated from a combination of schemes and structure sizes. This complexity is based on the number of computation steps represented by the algorithmic skeleton and does not consider complexity of operations performed in the structure nodes.
- **Complexity Based on Full Features.** It is like in previous case with additional taking into account complexity of operations in the structure nodes.
- **Complexity Defined by Component Designers.** Some times, it is difficult or unsafe to derive exact complexities of corresponding components automatically. As an attribute, the author can evaluate the complexity by himself.
- **Component Title.** A title is a conventional name which can be used as a keyword for searching operations.
- **Author.** This attribute includes a personal name, as well as a name of company, university, or research organization. For example, a search based on this attribute can provide a list of components registered by a particular group of people or organizations.
- **Date of Creation.** Currently, date is automatically set to the registration data. Also, this date implies latest modification date.
- **ID.** When a component is registered, a URL-like address is attached to the component. In this address, the component is available in the library component. It means that a direct observation of the component feature can be used for searching operations.

Above mentioned attributes are recorded for each library component. On assumption, each component can be applied to various application fields. It will be interesting and useful to search components by application fields as well. However, depending on creators and users, definitions of such attributes may be different. People in different fields have different knowledge and use different terminology. So, clusters of library components depending on people's knowledge or professional orientation should also be prepared in the library.

4.2 External Attributes

The second group is attributes to describe how the component is related to other components and how it has been used by other users. Relations between components and statistical attributes such as frequency of usage, can be attractive search keys. After the registration or specification, any activity with a component in the library will be recorded and used to update the statistics and links between the components. Table 2 shows the possible set of external attributes. In this section details of the attributes are considered.

- **Latest Referred Components.** A set of components that have been used recently (for example, this month) should be supported. This value can be used to derive activity rates as below.
- **Number of Usage.** This value can comprise famousness of corresponding components. Also, this value can be used to derive the activity rates.
- **Activity Rate.** This value computed from the latest referred components and the number of usage. This attribute is important not only for searching, but also for library arrangement. For example, passive knowledge such as components that are used seldom for a long time, will be removed to a special section of the library by the system automatically. Also, in some search interfaces, multiple search results are ordered based on the activity rate. That is to say, the most active component will appear first in a list of candidates.
- **Number of Updates.** This value is incremented every time when the component is modified. In some sense, it is like a version number. Definitely, a variety of components improves the library quality.
- **Parent and List of Children.** As mentioned previously, a library component can be derived from existing components by editing/composing operations. A parent as well

Table 2. Table for External Attributes

External Attributes
Latest referred date
Number of usage
Activity rate
Number of updates
Parent
List of children
Number of inheritance
List of source links
List of target links
Frequency of usage based on user's activity

as a list of children provide genetic (historical) information of the components. These attributes enable search based on inheritance relationship between components.

- **List of Links.** The links associate composite components. There are two types of links: for target and for source components.

Here, it is important to mention that a library component has an attribute to record the frequency of usage. It would be useful to know which components were frequently used by users from a certain field or profession. The library maintains tables to provide mapping between component IDs and frequencies for each different groups.

5 Possible Library Structure

In previous sections, many kinds of attributes are presented. These attributes are well classified to provide high-level searching operations. In this section, a possible library structure that makes efficient use of the attributes is considered. A main infrastructure is that each item in the library is stored as a special software component called cyberFilm. A cyberFilm represents a software component as a set of algorithmic features that provides a high-level and yet precise description of computation as well as semantics of problem that should be solved by the corresponding algorithm. So, it is easy to extract well classified attributes from them. These attributes are used in special modules of the library to perform searching operations. We do not mention about concrete search algorithms and their efficiency, but we focus on kinds of search methods applicable with the attributes through a special set of library interface panels. Technically, the interfaces are to implicitly create SQL-like queries that are sent to conventional database system to get corresponding target components. Our goal is to provide search interfaces that enable users to specify high level representation of semantic of problem and computational features of algorithm, that is "what" and "how" problem description. Of course, the interface panels can be combined to provide more sophisticated searching operations.

Figure 7 shows an example of an interface for searching by icon language with a narrow down technique. Users can specify preferred values for several attributes to find matching components. The specification of the attributes is based on a set of icons. Figure 7 shows that our focus is on searching with attributes related to semantic features of problem input/output and algorithmic features of computational methods that can solve the problem. In this example, in top-left panel, problem input and output are specified by icons. In each icon, we can see a cube with color that represents values of data unit on cube points. Additionally, the user can specify mathematical/physical relation between these semantic features by additional micro-icons. At the below of the semantic feature panel, we can specify icons related to algorithmic features, or system shows such icons that provide problem solution automatically.

Figure 8 shows an example of an interface for search by specifying possible ranges of attributes values. Many kinds of numerical values can be extracted from attributes related to both "what" and "how" problem description. This interface shows components that match several attributes simultaneously in a two or three dimensional surface. It has vertical and horizontal groups of axes to specify value ranges of several attributes. The number of axes and corresponding attributes are changeable. Operations such as "union," "in-

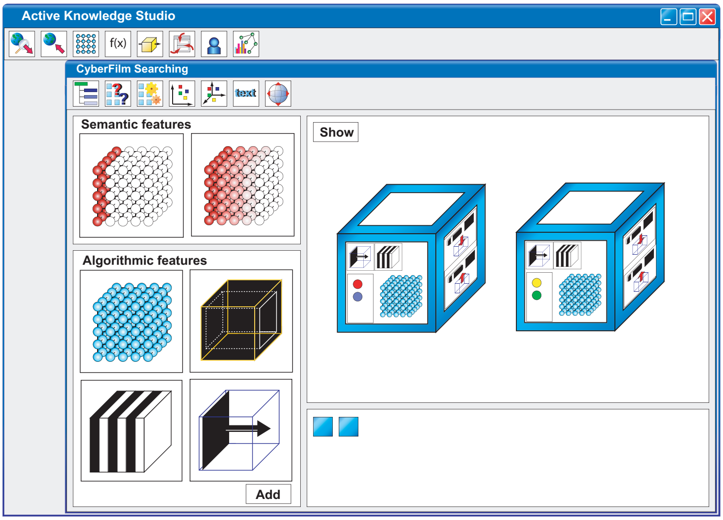


Fig. 7. An interface for search on queries

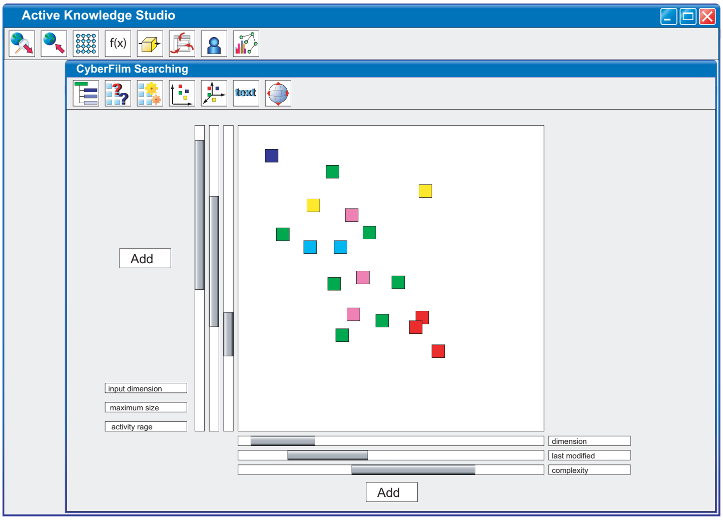


Fig. 8. An interface for search on values of attributes

tersection," "difference," and "complement" of these attributes are supported. Selected components are presented by colored squares according to their features.

Figure 9 shows an example of an interface for searching on the interrelationships between components. These searching operations are based on traversing library components connected by links on a three dimensional spherical surface. Such information visualization technique is presented in [15]. A main point of this searching method is considering different kinds of interrelationships. Some of them are related to the embed-

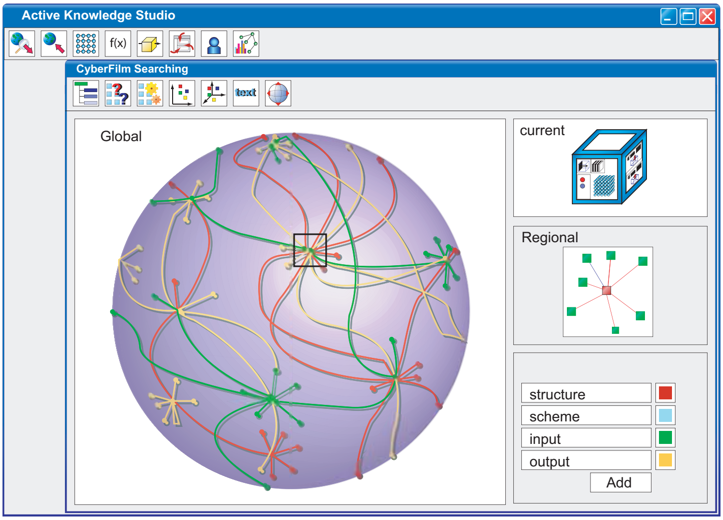


Fig. 9. An interface for search on interrelationships

ded inheritance, others are created as a result of the component and library use. Basically, a set of links implies relations between components, so they are changeable according to users purposes. One of useful relationship is similarity. The proximity of any two components are derived by similarity of problem descriptions and how algorithmically (based on the attributes) similar they are. As a rule, lengths of the links depict distances between components. Consequently, components that have similar attributes are gathered, while less similar components will appear distantly from one another. The distances between two components and their locations are computed based on the classification codes. The inheritance relationship uses the list of parents and children. Also, links to other components can be considered. A component can include other component. This link corresponds to such a relationship. These searching operations enable the users not only to search target components, but also to discover new knowledge and algorithms for them. The user can start searching from any component and then traverse in a direction based on his/her intentions. Also, the interface will support searching on multiple attributes. In such case, the attributes are distinguished by using different colors of links, and intensities of relevance ratios are represented by thickness of links.

6 Conclusion

A searching method within a special library related to "what" and "how" problem descriptions has been presented. The "what" problem description is based on a high level representation of general features of initial and result data the problem can operate on and produce. The "how" problem description is based on another high level representation of computational features of algorithm used for the problem solution. To realize the searching method, an infrastructure with a special library (in future, a database) in which each item is stored as a cyberFilm has been considered. A cyberFilm represents a software component as a set of algorithmic features that provide a high-level and yet precise description of computation as well as semantics of problem that should be solved by the corresponding algorithm. These features are a basis for creation of the classification code and the searching method. A basic idea of the "what" problem description for the searching method is to use generalized input/output specifications of semantic types. The "how" problem description is based on sets of attributes representing algorithmic features and allow a search from multiple points of view. Some interface panels that are based on the attributes have been introduced. These searching operations and their combinations use many attributes, so we will analyze and evaluate our searching process in future works.

References

1. A. P. Black, Post-Javaism, IEEE Internet Computing, Vol.8, No.1, 2004, 96, 93–95.
2. Christian S. Collberg, A Fuzzy Visual Query Language for a Domain-Specific Web Search Engine, Proceedings of the Second International Conference on Diagrammatic Representation and Inference, 2002, 175–190.
3. T. Yamamoto, Overview of Component Search System SPARS-J, International Workshop on Community-Driven Evolution of Knowledge Artifacts, University of California at Irvine, 2003.
4. R. C. Seacord, S. A. Hissam, and K. C. Wallnau, Agora: A Search Engine for Software Components, IEEE Internet Computing, Vol.2, No.6, 1998, 62–70.
5. G. T. Heineman, and W. T. Council, Component-based Software Engineering: putting the pieces together, Addision-Wesley, NHH(2001). New York, 1998.
6. F. P. Brooks, No Silver Bullet: Essence and Accidents of Software Engineering, *Computer*, Vol.20, No.4, 1987, 10–19.
7. S. Hallem, D. Park, and D. Engler, Uprooting Software Defects at the Source, Queue, Vol.1, No.8, 2004, 65–71.
8. Steven S. Skiena, The Algorithm Design Manual, TEROS, 1998.
9. Nicolai M. Josuttis The C++ Standard Library A Tutorial and Reference, Addision-Wesley, 2001.
10. LEDA, http://www.algorithmic-solutions.com/
11. R. Yoshioka, N. Mirenkov, Visual Computing within Environment of Self-explanatory Components, Soft Computing Journal, SpringerVerlag, Vol.7, No.1, 2002, 20–32.
12. N. Mirenkov, A. Vazhenin, R. Yoshioka, T. Ebihara, T. Hirotomi, T. Mirenkova, Self-explanatory components: a new programming paradigm. International Journal of Software Engineering and Knowledge Engineering, world scientific, Vol.11, No.1, 2000, 5–36.
13. R. Yoshioka, N. Mirenkov, Y. Tsuchida, Y. Watanobe, Visual Notation of Film Language System, Proceedings of 2002 International Conference of Distributed Multimedia Systems, San Francisco, California, USA, 2002, 648–655.
14. R. Roxas and N. Mirenkov, Visualizing Input/Output Specification Proceedings of 2002 International Conference of Distributed Multimedia Systems, San Francisco, California, USA, 2002, 660–663.
15. S. K. Card, J. D. Mackinlay, and B. Shneiderman, Information Visualization, Morgan Kaufmann Publishers, 1999.
16. System for Universal Media Searching(SUMS), http://sumscorp.com/develop

Augmenting the Power of the Various Versions of LSI Used in Document Retrieval

Hua Yan[1], William I. Grosky[2], and Farshad Fotouhi[1]

[1] Computer Science Department, Wayne State University, Detroit, MI 48202
{hyan, fotouhi}@cs.wayne.edu
[2] Department of Computer and Information Science,
University of Michigan-Dearborn, Dearborn, MI 48128
wgrosky@umich.edu

Abstract. This paper clarifies the apparent randomness and confusion in the choice of different query methods using LSI to be found in the current text and image retrieval literature. We also propose and test some modified query approaches using the novel technique of singular value rescaling (SVR). Experiments on standardized TREC data set confirmed the effectiveness of SVR, showing an improvement ratio of 5.6% over the best conventional LSI query approach.

1 Introduction

Much research [1], [2], [3] has been done on Latent Semantic Indexing (LSI) since it was first proposed in 1989 by Deerwester, Dumais, Furnas, et. al. [4]. Usually, LSI utilizes the singular value decomposition (SVD) model. For a collection of d documents containing t distinctive terms, a term-by-document matrix, A_0, is constructed, where $A_0(i,j)$ is the frequency with which term i occurs in document j, after any necessary term weighting has been performed [1]. Subjected to the SVD process, A_0 is transformed to $U_0\ S_0\ V_0^T$, where U_0 is a $t\times r$ orthogonal matrix, S_0 is an $r\times r$ diagonal matrix bearing singular values $s_1 \geq s_2 \geq, \ldots, \geq s_r$, and V_0^T is an $r\times d$ orthogonal matrix. Note that in the text retrieval literature, document vectors can refer either to the column vectors $A_{0\ (:,\ i)}$ or to the column vectors $V_0^T{}_{(:,\ i)}$, and term vectors can refer either to the row vectors $A_{0\ (j,\ :)}$ or to the row vectors $U_{0\ (j\ ,:)}$. The same nomenclature also applies to the dimensionally reduced model of $A = U\ S\ V^T$, where U consists of the first k columns of U_0, V^T consists of the first k rows of V_0^T, and S stores the largest k singular values of S_0. Presumably, the new model, A, better captures the *hidden* (latent) semantic structure of relations between terms and documents than does the original model A_0.

The remainder of this paper is organized as follows. In Section 2, we present three versions of the standard query method, clarifying the different approaches in the literature. These versions are compared in Section 3. Our new approach is discussed in Section 4, along with our experiments. Finally, we present our conclusions in Section 5.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 150 – 158, 2005.

2 Standard Query Method: Three Philosophies, Three Versions

There exist three philosophies on how to conduct a query q in the dimensionally reduced LSI model, giving rise to the following three versions of the standard query method:

2.1 Version A

Underlying Philosophy. Column vectors $(V^T_{(:,\ 1)}, ..., V^T_{(:,\ d)})$ in matrix V^T are k-dimensional document vectors, their dimension having been reduced from t. Dimensionally-reduced $V^T_{(:,\ i)}$ should carry some kind of latent semantic information captured from the original model and may be used for querying purposes. However, since $V^T_{(:,\ i)}$ is k-dimensional while q is t-dimensional, we need to translate q into some proper form in order to compare it with $V^T_{(:,\ i)}$. Observing that $A=(A_{(:,\ 1)}, ..., A_{(:,\ d)})$ and $V^T = (V^T_{(:,\ 1)}, ..., V^T_{(:,\ d)})$, equation $A = USV^T$ leads to $(A_{(:,\ 1)}, ..., A_{(:,\ d)}) = U\ S\ (V^T_{(:,\ 1)}, ...,$ $V^T_{(:,d)})$. Thus, for any individual column vector in A, we have, $A_{(:,\ i)} = USV^T_{(:,\ i)}$, for $1 \leq i \leq d$, which implies that $V^T_{(:,\ i)} = S^{-1}\ U^T A_{(:,\ i)}$, for $1 \leq i \leq d$. Treating q just like a normal document vector $A_{(:,\ i)}$, we transform q to $qa = S^{-1}\ U^T\ q$. Now, qa has the same dimension as $V^T_{(:,\ i)}$. Some papers which have used this approach are [5], [6].

Query method. First, use formula $qa = S^{-1}\ U^T\ q$ to translate the original query q into a form comparable with any column vector $V^T_{(:,\ i)}$ in matrix V^T. Then compute the cosine between qa and each $V^T_{(:,\ i)}$, for $1 \leq i \leq d$.

2.2 Version B

Underlying Philosophy. As mentioned earlier, document vectors can mean two different things: either column vectors $(V^T_{(:,\ 1)}, ..., V^T_{(:,\ d)})$ in V^T or column vectors $(A_{(:,1)}, ..., A_{(:,\ d)})$ in A. In fact, the latter ones might be a better choice for serving as document vectors because they are rescaled from the dimensionally reduced U and V by a factor of S after the SVD process. To utilize $(A_{(:,\ 1)}, ..., A_{(:,\ d)})$ for querying purposes, we only need to take one further step on the basis of version A, which is to scale k-dimensional qa back to t dimensions: $qb = U\ S\ qa = U\ S\ (S^{-1}\ U^T\ q) = UU^Tq.$[1] Some papers which have used this approach are [4], [7].

Query method. First, use formula $qb = UU^Tq$ to translate the original query q into a folded-in-plus-rescaled form comparable with any column vector $A_{(:,\ i)}$ in matrix A. Then compute the cosine between qb and each $A_{(:,\ i)}$ $(1 \leq i \leq d)$.

2.3 Version B'

Underlying Philosophy. All the reasoning behind version B sounds good except for one thing: since we are going to use t-dimensional column vectors $(A_{(:,\ 1)}, ..., A_{(:,\ d)})$ in A as document vectors, we may not need to first fold in q and then rescale it back to t-dim: we can just use the original query q (which is already t-dimensional) for

[1] It should be pointed out that because of dimensional reduction, $U^TU=I$ while $UU^T \neq I$. On the other hand, $U_0^TU_0=I_{rxr}$ and $U_0U_0^T=I_{txt}$.

comparing with each t-dimensional $A_{(:,\ i)}$ $(1 \le i \le d)$. Some papers which have used this approach are [3], [8].

Query method. Compute the cosine between q and each $A_{(:,\ i)}$ $(1 \le i \le d)$.

3 Analysis of the Three Versions of the Standard Query Method

We now compare the various version of the standard query method discussed in the previous section.

3.1 Comparison of Versions A and B

For version A[2]: $qa\_result = (V^T{}_{(CN)})^T \bullet qa_{(N)} = V_{(RN)} \bullet qa_{(N)}$. For simplicity, we drop off the normalization subscript here: $qa\_result = V \bullet qa = V \bullet (S^{-1}\, U^T q)$. Therefore,

$$qa\_result = VS^{-1}U^Tq\ . \tag{1}$$

For version B: $qb\_result = A_{(CN)}{}^T \bullet qb_{(N)}$. For simplicity, we also drop off the normalization subscript here. Observing $A = USV^T$, $qb = UU^Tq$, and $U^TU = I$, we have $qb\_result = A^T \bullet qb = (USV^T)^T(UU^Tq) = (VSU^T)(UU^Tq) = VS(U^TU)U^Tq$. Therefore,

$$qb\_result = VSU^Tq\ . \tag{2}$$

Comparing (1) and (2) shows that the difference between the two query results is a factor of S^2. Our experiments show that whenever a condition of $S \approx cI$ (where c is a constant and I is the identity matrix) is obtainable, the queried results between version A and version B demonstrate a high correlation. In a real-world application environment, the number of singular values (the rank of S) is normally in the hundreds, and the largest few singular values are usually many times bigger than the smallest few. Therefore, the approximation $S \approx cI$ usually does not hold, which means that the retrieval results between these two versions may be quite significant. Hence the importance of finding out which one is preferable.

3.2 Comparison of Versions B and B'

Let $qb\_result = (qb\_\text{result}\ {}_i)^T$ and $qb'\_result = (qb'\_result\ {}_i)^T$, where $1 \le i \le d$. After some simple mathematical transformations, we have:

$$qb\_result_i = \frac{(V^T{}_{(:,i)})^T\, S\ U^T q}{\left\|A_{(:,i)}{}^T\right\|} \bullet \frac{1}{\|qb\|}\ , \tag{3}$$

$$qb'\_result_i = \frac{(V^T{}_{(:,i)})^T\, S\ U^T q}{\left\|A_{(:,i)}{}^T\right\|} \bullet \frac{1}{\|q\|}\ . \tag{4}$$

[2] For a matrix, subscript $_{(RN)}$ denotes the row normalization process, while subscript $_{(CN)}$ denotes the column normalization process. For a vector, subscript $_{(N)}$ denotes normalization to unit length. Note that for two normalized vectors (i.e. unit vectors), cosine and dot product are the same.

Dividing (4) by (3) yields $\frac{qb'\_result_i}{qb\_result_i} = \frac{\|qb\|}{\|q\|}$. Since $qb=UU^Tq$, we have $\frac{qb'\_result_i}{qb\_result_i} = \frac{\|UU^Tq\|}{\|q\|}$, or $qb'\_result_i = (\|UU^Tq\|/\|q\|)\ qb\_result_i$. Because $\|UU^Tq\|/\|q\|$ does not depend on i, we have $(qb'\_result_1,\ldots, qb'\_result_d)^T = (\|UU^Tq\|/\|q\|)\ (qb\_result_1,\ldots, qb\_result_d)^T$. Therefore, $qb'\_result = (\|UU^Tq\|/\|q\|)\ qb\_result$. We see that for each particular query q, the results of versions B and B' differ by a constant factor $\|UU^Tq\|/\|q\|$, which we name the **α factor**. This α factor has two ramifications:

(1) For each particular query q, α is a constant, usually not equal to 1. There are two situations here. One is when the retrieval criterion is a particular cosine threshold (e.g. 0.5), in which case the two sets of retrieved documents from versions B and B' will probably vary a lot. The other situation is when the retrieval criterion is Choose-the-Best-*N*-Scores (CBNS) or Choose-the-Top-Percentage-Scores (CTPS), in which case the two sets of retrieved documents from versions B and B' are guaranteed to be identical. Here CBNS refers to the retrieval criterion in which all queried documents are sorted in descending order with respect to their cosine scores and only those documents with the best N ($1 \le N \le d$) scores are retrieved. For CTPS, documents are also sorted in descending order of their cosine scores but only a specific proportion (e.g. top 5%) of the d documents are retrieved.

(2) If a number of queries q_1, q_2,..., q_n are conducted, there will be n independent values of α: α_1, α_2,..., α_n. In order to achieve identical retrieval results for both versions over these n queries, the same CBNS/CTPS criterion need to be applied to both

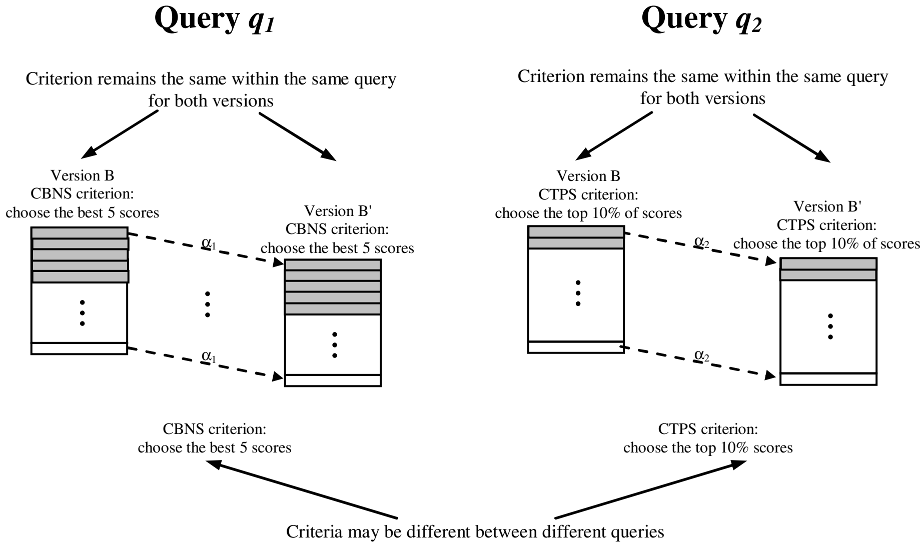

Query q_1
Query q_2
Criterion remains the same within the same query for both versions
Criterion remains the same within the same query for both versions
Version B
CBNS criterion:
choose the best 5 scores
Version B'
CBNS criterion:
choose the best 5 scores
α_1
α_1
Version B
CTPS criterion:
choose the top 10% of scores
Version B'
CTPS criterion:
choose the top 10% of scores
α_2
α_2
CBNS criterion:
choose the best 5 scores
CTPS criterion:
choose the top 10% scores
Criteria may be different between different queries

Fig. 1. Illustration of the equivalency principal for versions B and B’

versions for each particular query; however, different CBNS/CTPS criteria may be applied to different queries as long as the criterion remains the same for both versions within each query. We call this kind of requirement for identical retrieval results between versions B and B' over a number of independent queries the ***Equivalency Principle*** (see Figure 1). Because of this principle, versions B and B' are interchangeable.

4 Seeking the Best: Looking Beyond the Current Versions

We now strive to identify the better version between version A and version B (or its equivalent version B'). For this purpose, an intermediate approach, version C, is created by removing the S from the expressive form of VSU^Tq. Applying the transpose to the various formulas, we have, $qa\_result^T = q^TUS^{-1}V^T$, $qb\_result^T = q^TUSV^T$, *and* $qc\_result^T = q^TUV^T$.

For a given query q of t dimensions, we first reduce it to a vector of fewer dimensions, k, via dot-products with column vectors in U; we then adjust this transformed query vector q^TU by re-scaling it along the k dimensions with either S^{-1} (version A), S (version B) or the identity matrix I (version C). Thereafter, we normalize this transformed query vector q^TUS^i (i = -1, 0, or 1) so that it has a unit length while retaining the original direction (we can visualize this step as the concerned vector getting stretched out or pressed in so that its endpoint falls onto the unit hypersphere). Then, as a final step, we compute the angles between this transformed query vector and all the k-dimensional document vectors (column vectors in V^T) which are also located on the same unit hypersphere due to normalization. Here we see that S serves as a re-scaling factor between the query vector and each of the compared document vectors.

Now let's label these three versions with respect to the factor of S: version A is S-negative, version B is S-positive, and version C is S-neutral. One of the crucial steps in building the LSI model is the dimensional reduction. This is done by removing all the minor $(r - k)$ singular values, leaving only the k largest singular values in matrix S to retain the k most prominent conceptual features. The rationale behind this approach is that the noise has been extracted by the SVD process and is now distributed along the minor $(r - k)$ dimensions, whose subsequent removal cleared away much of the noise. We see here that the importance of a particular dimension is directly measured by its associated singular value. In the new rank k model, just as in the original rank r model, dimensions are of varying importance. In the S-positive version, we preserve this property of dimensions, while in the S-neutral version we discard this property. Using the S-positive version as a basis, in the S-neutral version we scale down the prominent dimensions and scale up the minor dimensions to the artificial effect that they are brought to be equal. As a result of this, in the S-neutral version, noise is also scaled-up so long as the minor dimensions are scaled-up. Therefore, version B (S-positive) is a rationally better approach than version C (S-neutral). As for version A (S-negative), it's just one step further along the direction of version C. Since we reject version C, we incidentally also reject version A. Looking one step further, we propose that the following versions can be even better options than version B (for identification purpose, we use notation Dx to indicate a version where S^x is applicable):

Version D1.5: $qd\_result^T = q^T U \boldsymbol{S}^{\boldsymbol{1.5}} V^T$
Version D2: $qd\_result^T = q^T U \boldsymbol{S}^{\boldsymbol{2}} V^T$
Version D4: $qd\_result^T = q^T U \boldsymbol{S}^{\boldsymbol{4}} V^T$
…

Since these alternative query methods rescale the singular values that remain intact in the standard query method of version B, we call this technique **singular value rescaling**, or **SVR** for short. It remains to be seen whether these alternative versions are realistically interesting approaches in terms of improved retrieval performance or if they are just some artificial and trivial devices that unduly stretch the LSI model beyond practicality. In the next section, we carry out some experiments in an attempt to find out the answer to this question.

4.1 Description of the Prototyping Experiments

To test the validity of the ideas presented in this paper, we chose to use the TREC-4 data set CD ordered from http://trec.nist.gov/ website. There are three categories of document data available from this CD:

- Congressional Record of the 103rd Congress: approximately 30,000 documents (235 MB)
- Federal Register (1994): approximately 55,000 documents (395 MB)
- Financial Times (1992-1994): approximately 210,000 documents (565 MB)

For our experiments, the category III data set (Financial Times 1992-1994) were chosen. Noticing that the sizes of the documents vary from just a few lines to hundreds of lines, we allowed only those documents with between 10 and 36 lines of text. There are a total of 200 topics (TREC topics 251 ~ 450) applicable to this data set. To avoid over-concentration of documents on just a few particular topics, no more than 40 documents per each topic were allowed in the experiments. A total of 1037 documents allocated through 96 topics were selected after these restrictions were applied.

To select the keywords used in the experiments, we first applied the Porter stemming algorithm[3] and the standard stopwords[4] exclusion to all the words in these 1037 documents. This process left us with 12,254 unique words, among which there were the following two types of words:

- Words that occur (once or many times) in one document only
- Words that occur once in two documents only (these words have a total frequency of 2)

[3] Available at http://www.tartarus.org/~martin/PorterStemmer/

[4] Available at http://www.lextek.com/manuals/onix/stopwords1.html

In order to reduce the number of words in the experiments for efficient computations, these two types of words were excluded from the 12254-count word list, obtaining a 4950-count list, the final keyword list used in the experiments. Because LSI works by detecting the relationship of term *co-occurrences* in the text corpora, the exclusion of these two types of words should have only a minimum impact on the performance result.

To make our experimental results more general, we randomly chose the following two sets of 15 topics each, among those that have a minimum number of six relevant documents:

Set A topics: 255, 285, 304, 318, 340, 351, 354, 376, 378, 389, 392, 400, 422, 425, 445

Set B topics: 274, 294, 307, 319, 326, 343, 350, 385, 392, 401, 421, 425, 431, 443, 449

Each topic contributed one query only. The 15 queries that were produced from set A are now called *training queries*, since they were used in the training part of the experiments to determine the optimal number of dimensions for the LSI model. The 15 queries that were produced from set B topics are now called *testing queries*, since they were used in the testing part of the experiments to either validate or refute the effectiveness of the proposed singular value rescaling technique. Since there were 4095 keywords and 1037 documents, a term-by-document matrix of size 4095x1037 was created using the term-frequency weighting [1]. Each column was normalized to avoid the crowding out of small documents by large documents. It turned out that this original term-by-document matrix has a full rank of 1037.

4.2 Analysis of the Prototyping Results

Table 1 summarizes the experimental results for version B, version A and straight lexical match. The data shown in this table were averaged over the 15 training queries. For each of these queries, the value participating in this average measures the area under the corresponding interpolated recall precision curve [9] over the full range [0,1] of recall. In Table 1a, the row where Dimension = 1037 gives the retrieval result of 0.5462 for straight lexical matching, because the LSI model degenerates into straight lexical matching at full rank, i.e. when no dimensional reduction is performed. The row where Dimension = 180 gives the optimal performance of 0.6325 for version B (which proved to be the same as version B'). In Table 1b, the row where Dimension = 140 gives the optimal result of 0.5984 for version A. The interesting fact here is that although both versions A and B were superior to the straight lexical match, version B performed even better than version A at their respective optimal dimensions. This confirmed the previous theoretical conclusion that version B is a better approach than version A. Note that the improvement ratio of version B over straight lexical matching was (0.6325 – 0.5462) / 0.5462 = 15.8%.

Table 2 summarizes the experimental result for standard version B along with various non-standard versions using SVR at the determined optimal dimension of 180. The data shown in this table were averaged over 15 testing queries. Notice that the averaged result of version B (= 0.6335) at Dimension = 180 in Table 2a was about the same as the averaged result of version B (= 0.6325) at Dimension = 180 in Table 1a.

Table 1. Results of version B (B'), version A, and straight lexical match

S_exponent = 1	
Dimension	**Average Result**
100	0.5591
120	0.5905
140	0.5994
160	0.6174
180	***0.6325***
200	0.6316
220	0.6251
240	0.6256
260	0.6239
280	0.6152
300	0.6141
400	0.6009
500	0.5863
600	0.5817
700	0.5726
800	0.5641
900	0.5540
1000	0.5465
1037	***0.5462***
Table 1a	

S_exponent = -1	
Dimension	**Average Result**
20	0.2635
40	0.4435
60	0.4974
80	0.5470
100	0.5885
120	0.5941
140	***0.5984***
160	0.5967
180	0.5931
200	0.5796
300	0.5152
400	0.4599
500	0.4106
600	0.3681
700	0.3384
800	0.3274
900	0.3078
Table 1b	

This fact is in line with the randomness of the selection processes that produced these two sets of queries, described earlier. Table 2a shows that, through applying the SVR technique, version $D^{1.6}$ (where the rescaling factor *S_exp* = 1.6) gives the optimal averaged result of 0.6687. The improvement ratio of version $D^{1.6}$ over version B is (0.6687 – 0.6335) / 0.6335 = 5.6%. Table 2b shows the individual results of version B and version $D^{1.6}$ over the 15 topics. Notice that, except for the queries on topics #343 and #392, version $D^{1.6}$ produces better results than does version B over the queries on all the other 13 topics. This shows that, although the improvement ratio of 5.6% over averaged results is moderate, the improvement has been quite consistent among all tested queries. In addition, we have that version $D^{1.6}$ performs steadily better than version B over the full range of Recall values.

Table 2. Results of singular value rescaling for dimension = 180

Version	S_exponent	Average Result
B (B')	**1**	**0.6335**
D1.2	1.2	0.6426
D1.4	1.4	0.6533
D1.5	1.5	0.6636
D1.6	**1.6**	**0.6687**
D1.7	1.7	0.6683
D1.8	1.8	0.6666
D2	2	0.6631
D3	3	0.6138
D4	4	0.5596
D5	5	0.4544
D6	6	0.3519
D7	7	0.2590
D8	8	0.1963
Table 2a		

Topic #	S_exponent=1	S_exponent=1.60
#274	***0.52556***	***0.58032***
#294	***0.84443***	***0.89399***
#307	***0.52520***	***0.58372***
#319	***0.67792***	***0.70311***
#326	***0.75722***	***0.79321***
#343	0.45782	0.44745
#350	***0.79519***	***0.87194***
#385	***0.41990***	***0.45463***
#392	0.62522	0.60545
#401	***0.65024***	***0.68549***
#421	***0.79614***	***0.82281***
#425	***0.71925***	***0.74758***
#431	***0.46439***	***0.52218***
#443	***0.77732***	***0.79893***
#449	***0.46652***	***0.51901***
Average =	0.63349	0.66865
Improvement Ratio =		5.6%
Table 2b		

5 Conclusions

In this paper we identified and analyzed three standard query methods (versions A, B and B') found in the current LSI text and image retrieval literature. Mathematical analysis showed that (i) the difference between the results of versions A and B is S^2 with S being the diagonal matrix of singular values in the dimension-reduced model, (ii) the retrieval results from versions B and B' are always identical if the Equivalency Principle is satisfied, and (iii) version B (B') should be a better option than version A. Experiments on a standardized TREC data set confirmed the latter two findings.

Furthermore, some interesting non-standard versions of query methods, applying the novel technique of *singular value rescaling* (SVR) were proposed and studied. The improvement ratio of (i) using SVR in addition to conventional LSI over (ii) using conventional LSI alone was 5.6%, according to our experiments. The discovery of singular value rescaling bears the practical significance that the current LSI information retrieval techniques may be significantly improved by simply adopting a novel query method which is computationally as efficient as the best standard query method, identified in this paper as version B.

We note that similar work on the scaling issues on text retrieval can be found in [10], which proposed a non-SVD-based iterative multiple-time (depending on the final dimension of the model) scaling technique. Our work is different, in the sense that SVR is a direct (instead of iterative) one-time (instead of multiple-time) scaling technique that builds on the basis of SVD. However, since both works involve scaling, it would be interesting to investigate the difference between these two techniques as part of our future work.

References

1. Dumais, S.: Improving the Retrieval of Information from External Sources, Behavior Research Methods, Instruments and Computers (1991) 229-236
2. Berry, M., Dumais, S., O'Brien, G.: Using Linear Algebra for Intelligent Information Retrieval, SIAM Review (1995) 573-595
3. Kolda, T.G., O'Leary, D.P.: A Semi-Discrete Matrix Decomposition for Latent Semantic Indexing in Information Retrieval, ACM Transactions on Information Systems (1998), 322-346
4. Deerwester, S., Dumais, S., G. Furnas, et. al.: Patent: Computer Information Retrieval using Latent Semantic Structure, U. S. Patent No. 4,839,853 (1989)
5. Berry, M., Fierro, R.: Low-Rank Orthogonal Decompositions for Information Retrieval Applications, Numerical Linear Algebra with Applications (1996), 301-328
6. Jiang, J.: Using Latent Semantic Indexing for Data Mining, MS Thesis, Department of Computer Science, University of Tennessee (1997)
7. Zhao, R., Grosky, W.I.: Negotiating the Semantic Gap: From Feature Maps to Semantic Landscapes, Pattern Recognition (2002), 593-600
8. Ding, C.H.: A Probabilistic Model for Dimensionality Reduction in Information Retrieval and Filtering, Proceedings of the First SIAM Computational Information Retrieval Workshop, Raleigh, NC (2000)
9. Salton, G., McGill, M.J.: Introduction to Modern Information Retrieval, McGraw-Hill, New York, (1983)
10. Ando, R.K.: Latent Semantic Space: Iterative Scaling Improves Precision of Inter-document Similarity Measurement, Proceedings of SIGIR, Athens, Greece (2000), 216-223

Networked Mining of Atomic and Molecular Data from Electronic Journal Databases on the Internet

Lukáš Pichl[1], Manabu Suzuki[1], Kazuyuki Joe[2], and Akira Sasaki[3]

[1] Department of Computer Software, University of Aizu, Ikki,
Tsuruga, Aizuwakamatsu 965-8580, Japan
lukas@u-aizu.ac.jp
http://www.u-aizu.ac.jp/~lukas/

[2] Department of Information and Computer Sciences, Nara Women's University,
Kita-Uoya Nishimachi, Nara 630-8506, Japan

[3] Kansai Research Establishment, Japan Atomic Energy Research Institute,
8-1 Umemidai, Kizu-cho, Sourakugun, Kyoto 619-0215, Japan

Abstract. Several centers of atomic and molecular data in the world maintain research databases for use in fusion plasma simulations, hadron therapy, modelling the universe and other areas. Among the data center activities, collection of experimental and theoretical results across the world has been of major importance. This includes the identification, relevance assessment and retrieval of journal articles, followed by the data extraction, data mining, format conversion and data input. The methodology of the process still largely relies on working groups of specialists and part-time human labor, in spite of recent modernization in journal publishing, especially the electronic journals newly available in subscription domain and the free-access online abstract databases. This work focuses on automating the above procedure to the maximum extent possible. In particular, we design a download robot that performs query search and abstract retrieval for the candidates of relevant articles over the internet at first stage, followed by fultext retrieval (pdf format), text extraction and a deterministic relevance judgement. As a demonstration, we have also developed a bibliography database for electron-molecule collisions that automatically updates its contents over the internet in regular time intervals. The present work belongs to the project for evolutional data collecting system supported by a JSPS project which involves several research institutes.

1 Introduction

There is an increasing demand for accurate and reliable atomic and molecular (A+M) data motivated in part by the international project for thermonuclear fusion reactor prototype which requires accurate simulations of fusion edge plasmas on an unprecedented scale. It is also important that the data are available

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 159–170, 2005.

in a widely compatible format, suitable for sharing, database input, browser display or as a source data file for plasma simulation programs. The applications of A+M data range from basic science to industry projects, for instance the positron emission tomography or proton therapy of brain cancer.

The standard procedure for A+M data collection has been literature search and article analysis by specialists, who evaluate general paper relevance [1], recognize the process of interest and identify tables and graphs with relevant data. Part-time laborers then extract numerical tables (e.g. copy and reformat data from text-based pdf files or process bitmap images of older articles with optical character recognition software) or even sometimes estimate numerical values from embedded graphics. After adding cataloguing information, data are input to the DB via browser input forms or by the data file upload. Much of this procedure is fairly straightforward and can be automated, resulting in substantially decreased costs.

The present work is interesting and useful as a case study of database automation endeavors in large research institutions, in particular National Institute for Fusion Science (NIFS). We therefore do not aim at providing insight to mobile database systems [2] or evolutionary database systems [3], as others have done. Our joint project promotes text classification for identification of A+M relevant articles based on their abstracts; this work in particular contributes by elaborating on online search features and the autonomous database input.

In general, the online databases of scientific journal articles, i.e. the major information resource for new A+M data, restrict the data access to standard html interface with pages generated on the fly. Therefore the the search query distribution methods across various publishers have to deal with particular formats for web access, html requests, data posting and query output pages. Except for query output, the current publisher systems are rather standard, and skeleton programs can be used efficiently.

Once the article abstract is retrieved from a certain publisher, it has to be converted and analyzed. Here it is where various problems may arise, such as different html formatting in author names, chemical species labels, electronic state tags, etc. Such kind of complications can be dealt with and is expected to disappear once the A+M XML becomes a standard. Based on the converted abstract contents, the article should be assessed as relevant or not, to decrease the load in further fultext download and processing.

By using machine-learning based methods we can asses abstracts (and thus also the articles) to a large extent, identifying papers which are likely to contain A+M data. For instance, the Learning Vector Quantization (LVQ) can be efficiently used, provided it is calibrated to prefer sensitivity to specificity.

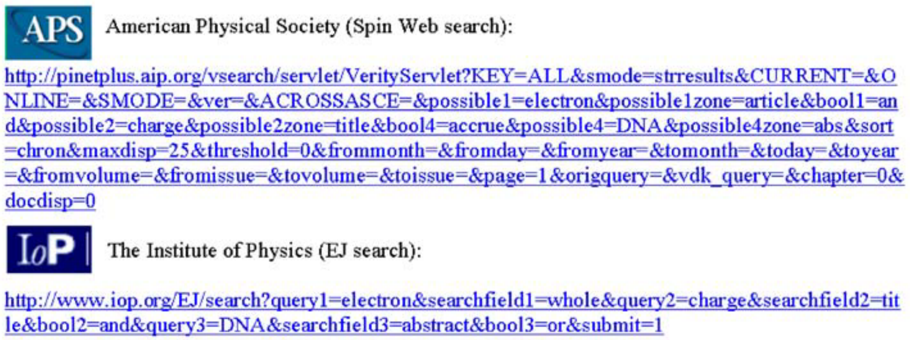
Articles assessed as relevant are downloaded from the publisher in pdf format. At present, there exists no reliable tool how to extract rich-featured journal article pdf files into html or A+M XML for further processing, including Adobe software. Superscripts of non-standard sizes, Greek characters, math-formatted inline expressions or equations are often lost or represented poorly. In spite of such complications, owing in part to specific features of article relevance assessment, quite reliable methods based on query look up with minimal formatting can be developed.

Here we present our first prototype solution for the above problems, namely a system which downloads and classifies articles from various publishers (in the form of joint search), retrieves and extracts pdf article, decides on paper relevance by a simple deterministic algorithm, and uploads accepted abstracts into a bibliography database in an autonomous way [4].

The paper is organized as follows. In Section 2, we discuss major online publisher sources for A+M research data and create an automated client program for query look up and abstract retrieval across these article sources. Section 3 deals with the retrieval of fultext source in pdf format, and the data extraction and conversion methods. A deterministic algorithm is designed for fultext-based paper relevance assessment and compared to AI based methods. In Section 4, we briefly describe an A+M bibliography database with autonomous input and methods of its management. We conclude with remarks and future prospects for the automation of numerical A+M databases in Section 5.

2 Search for DB-Entry Candidates (Abstracts)

The basic bibliographic entries in online databases of electronic journal publishers are author, title, journal, volume, issue, page number range, publication year, abstract; classification fields include method (theory, experiment), type of process, species involved, date of database entry and others. Typical are also the options for sorting query output or restricting search queries to a certain time period. Some of the search output fields (abstract, author name) are html formatted and include small figures for special characters or symbols.



American Physical Society (Spin Web search):

http://pinetplus.aip.org/vsearch/servlet/VerityServlet?KEY=ALL&smode=strresults&CURRENT=&ONLINE=&SMODE=&ver=&ACROSSASCE=&possible1=electron&possible1zone=article&bool1=and&possible2=charge&possible2zone=title&bool4=accrue&possible4=DNA&possible4zone=abs&sort=chron&maxdisp=25&threshold=0&frommonth=&fromday=&fromyear=&tomonth=&today=&toyear=&fromvolume=&fromissue=&tovolume=&toissue=&page=1&origquery=&vdk_query=&chapter=0&docdisp=0

The Institute of Physics (EJ search):

http://www.iop.org/EJ/search?query1=electron&searchfield1=whole&query2=charge&searchfield2=title&bool2=and&query3=DNA&searchfield3=abstract&bool3=or&submit=1

Fig. 1. Query form for online requests to the database of American Physical Society (upper part) and Institute of Physics (lower part). Their structure is practically identical, abstracting from naming conventions and formatting features

2.1 Online Publishers and Available Queries

We have selected the online journal archives of American Physical Society (APS, US based) and The Institute of Physics (IOP, UK based) as the sources for our prototype system. The two institutions cover approximately a 60% share of relevant A+M data and are therefore sufficiently representative. The available structure of online queries is quite similar, although the set of logic elements at APS is larger (including NOT and NEAR operators). Figure 1 shows the structural form for query posting, using

$$\text{“Electron” in ALL FIELDS } \langle\text{AND}\rangle \\ \text{“charge” in TITLE } \langle\text{OR}\rangle \text{ “DNA” in ABSTRACT}$$

for instance. Let us note that the data are posted in a hidden mode for the online IOP database. Otherwise, the query form is very much alike (formatting options are omitted in IOP case for clarity). The output pages, subject to data conversion and text extraction, are different from the viewpoint of html source, as shown in Fig. 2. String matching analysis in both cases is facilitated by the presence of a checkbox in page source (which is a rather general itemizing feature for commercial purposes). HTML source code for a single match is not very clean and has to be processed with care (cf. Fig. 3).

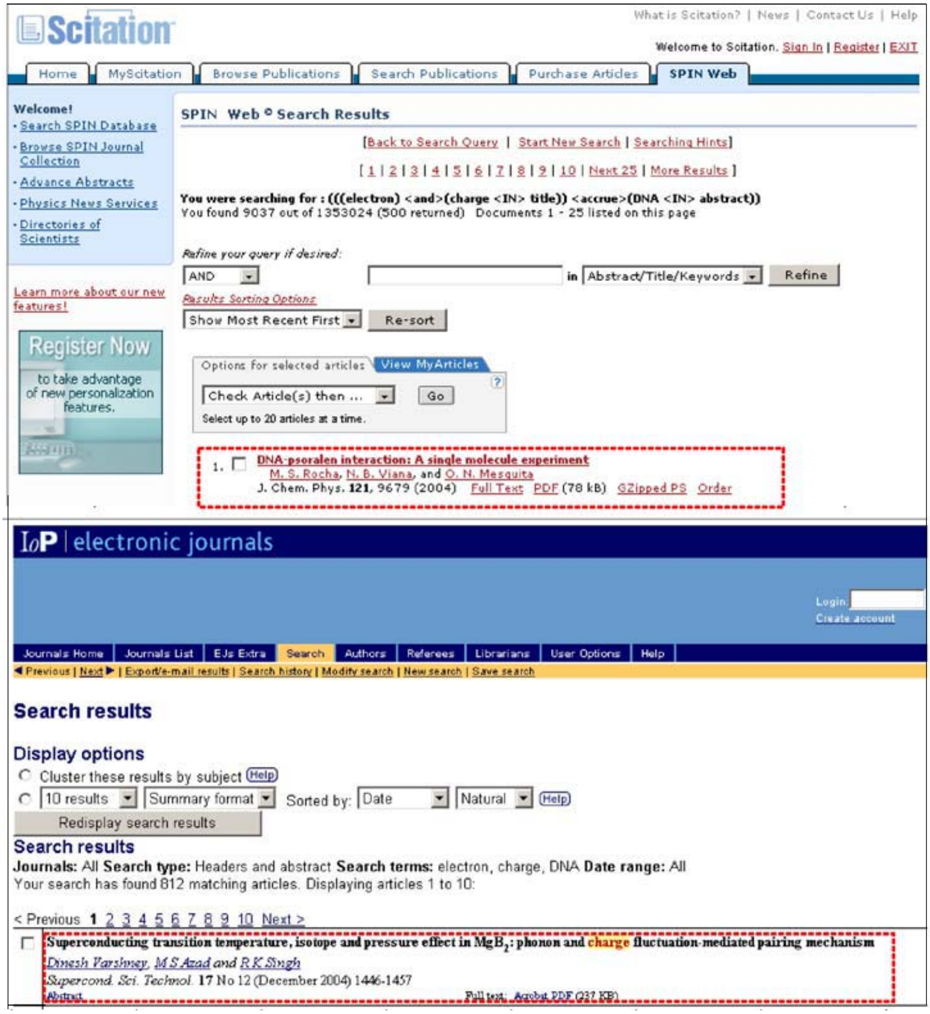
2.2 Retrieval and Analysis of Article Abstracts

The journal article abstracts are downloaded by using command line wget script which follows links in Figure 2. To prevent undesirable interference with publisher website monitoring software, abstract (and fultext article) downloads are scheduled in certain intervals that follow histogram of normal connections from subscribing institution. Because of low downloaded volume, the access rate remains within the conditions of subscription in either case. The stochastic download delay interval τ is set as

```
function $\tau${
 while(1){
    $t \leftarrow t_0 + 2T(r() - 0.5)$
    $c \leftarrow f_m r()$                                  (1)
    if($c < f(t)$) return $t$
 }
}
```

where $0 \leq r() < 1$ is a uniform random number and $f()$ is a probability density function (maximum value f_m). We adopt the normal distribution $f() = N(t_m, \delta)$ with $t_m = 10.0$ s and $\delta = 3.5$ s. Other forms of histogram can also be implemented.

Once the list of articles matching a certain predesigned query is downloaded from the database of the publisher, items such as author names, journal, article title etc. are extracted from the text (string matching in PHP) and input to the database. Since all items also include an abstract link, the text of each abstract is downloaded and stored locally as the analyzer script proceeds item by item.


Scitation
What is Scitation? | News | Contact Us | Help
Welcome to Scitation. Sign In | Register | EXIT
Home
MyScitation
Browse Publications
Search Publications
Purchase Articles
SPIN Web
Welcome!
Search SPIN Database
Browse SPIN Journal Collection
Advance Abstracts
Physics News Services
Directories of Scientists
Learn more about our new features!
Register Now
to take advantage of new personalization features.
[Back to Search Query | Start New Search | Searching Hints]
[1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | Next 25 | More Results]
Refine
Results Sorting Options
Re-sort
Options for selected articles
View MyArticles
Go
IoP | electronic journals
Login:
Create account
Journals Home
Journals List
EJs Extra
Search
Authors
Referees
Librarians
User Options
Help
◀ Previous | Next ▶ | Export/e-mail results | Search history | Modify search | New search | Save search
Redisplay search results
< Previous 1 2 3 4 5 6 7 8 9 10 Next >

SPIN Web © Search Results

You were searching for : (((electron) <and>(charge <IN> title)) <accrue>(DNA <IN> abstract))
You found 9037 out of 1353024 (500 returned) Documents 1 - 25 listed on this page

Refine your query if desired: AND in Abstract/Title/Keywords

Show Most Recent First

Check Article(s) then ...
Select up to 20 articles at a time.

1. □ DNA-psoralen interaction: A single molecule experiment
M. S. Rocha, N. B. Viana, and O. N. Mesquita
J. Chem. Phys. **121**, 9679 (2004) Full Text PDF (78 kB) GZipped PS Order

Search results

Display options
○ Cluster these results by subject (Help)
○ 10 results Summary format Sorted by: Date Natural (Help)

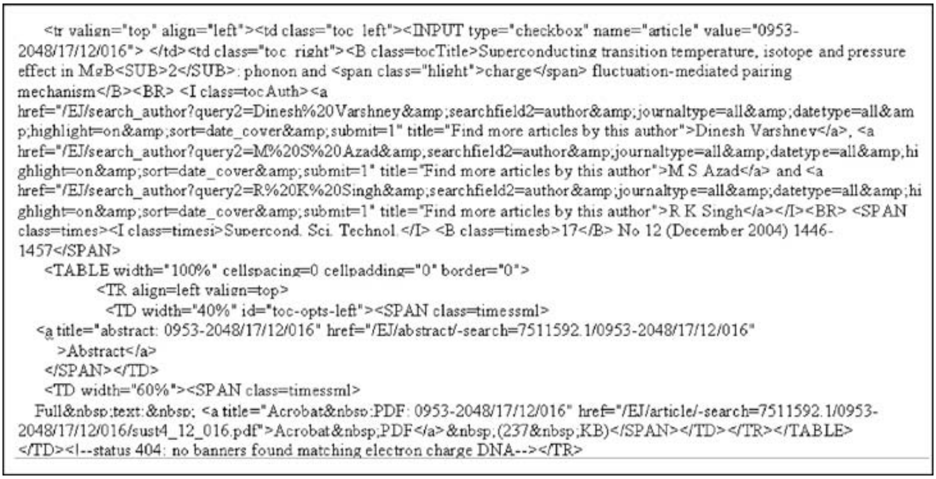
Search results
Journals: All **Search type:** Headers and abstract **Search terms:** electron, charge, DNA **Date range:** All
Your search has found 812 matching articles. Displaying articles 1 to 10:

□ **Superconducting transition temperature, isotope and pressure effect in MgB_2: phonon and charge fluctuation-mediated pairing mechanism**
Dinesh Varshney, M S Azad and R K Singh
Supercond. Sci. Technol. 17 No 12 (December 2004) 1446-1457
Abstract Full text: Acrobat PDF (237 KB)

Fig. 2. The output for query in Fig. 1 submitted to APS (top) and IOP (bottom). HTML source of both pages is retrieved by wget (linux command line), and bibliographic data are extracted using a string matching script written in PHP (relevant areas are indicated in the dashed line boxes)

3 Article Relevance Assessment (Pdf Fultext)

Before proceeding from the downloaded abstract to a fultext download, the abstract should be evaluated for relevance, so that the volume of downloaded pdf files is not excessive. This is a general task for machine learning algorithms which has been successfully treated by our colleagues in a pilot study by using learning vector quantization [5]. The abstract classification process consists of four steps.



```
    <tr valign="top" align="left"><td class="toc left"><INPUT type="checkbox" name="article" value="0953-
2048/17/12/016"> </td><td class="toc right"><B class=tocTitle>Superconducting transition temperature, isotope and pressure
effect in MgB<SUB>2</SUB>: phonon and <span class="hlight">charge</span> fluctuation-mediated pairing
mechanism</B><BR> <I class=tocAuth><a
href="/EJ/search_author?query2=Dinesh%20Varshney&amp;searchfield2=author&amp;journaltype=all&amp;datetype=all&am
p;highlight=on&amp;sort=date_cover&amp;submit=1" title="Find more articles by this author">Dinesh Varshney</a>, <a
href="/EJ/search_author?query2=M%20S%20Azad&amp;searchfield2=author&amp;journaltype=all&amp;datetype=all&amp;hi
ghlight=on&amp;sort=date_cover&amp;submit=1" title="Find more articles by this author">M S Azad</a> and <a
href="/EJ/search_author?query2=R%20K%20Singh&amp;searchfield2=author&amp;journaltype=all&amp;datetype=all&amp;hi
ghlight=on&amp;sort=date_cover&amp;submit=1" title="Find more articles by this author">R K Singh</a></I><BR> <SPAN
class=times><I class=timesi>Supercond. Sci. Technol.</I> <B class=timesb>17</B> No 12 (December 2004) 1446-
1457</SPAN>
    <TABLE width="100%" cellspacing=0 cellpadding="0" border="0">
          <TR align=left valign=top>
           <TD width="40%" id="toc-opts-left"><SPAN class=timessml>
   <a title="abstract: 0953-2048/17/12/016" href="/EJ/abstract/-search=7511592.1/0953-2048/17/12/016"
     >Abstract</a>
    </SPAN></TD>
    <TD width="60%"><SPAN class=timessml>
   Full&nbsp;text:&nbsp; <a title="Acrobat&nbsp;PDF: 0953-2048/17/12/016" href="/EJ/article/-search=7511592.1/0953-
2048/17/12/016/sust4_12_016.pdf">Acrobat&nbsp;PDF</a>&nbsp;(237&nbsp;KB)</SPAN></TD></TR></TABLE>
</TD><!--status 404: no banners found matching electron charge DNA--></TR>
```

Fig. 3. HTML source of a single matched article heading. The abstract link is followed by a wget script and its text is downloaded for relevance analysis

1. Prepare training data (papers and their abstracts) and test abstracts (T). Classify the training abstract data into (1) the A+M papers (A) and (2) the others (B).
2. Pre-process all abstracts of (A), (B) and (T). The pre-processing phase generates feature vectors based on frequency of word roots, technical terms of atomic and molecular physics, and specific notation of atomic states and processes.
3. Apply the LVQ algorithm with the feature vectors of abstracts in the sets (A) and (B).
4. Apply the learned reference vectors with the feature vectors of (T) for the recognition of the abstracts of atomic and molecular related papers. The result has a binary form: paper is A+M relevant or not (based on abstract analysis).

The first implementation of LVQ algorithm turned accurate in the order of ∼ 70%, and further enhancements are expected with the implementation of other machine learning algorithms, and also by allowing for a certain margin of A+M non-specific articles (few hundred percent allowable). Finally, more information on abstract relevance can be obtained from the full text of each article, as discussed in the following.

3.1 Format Conversion Tools

Virtually all online publishers use the pdf format for online journal articles, which is very useful from the viewpoint of file size or rich formatting features. Nevertheless, the pdf format is not suitable for fast string matching and text analysis. Since we analyze article abstracts at the HTML level, it would be

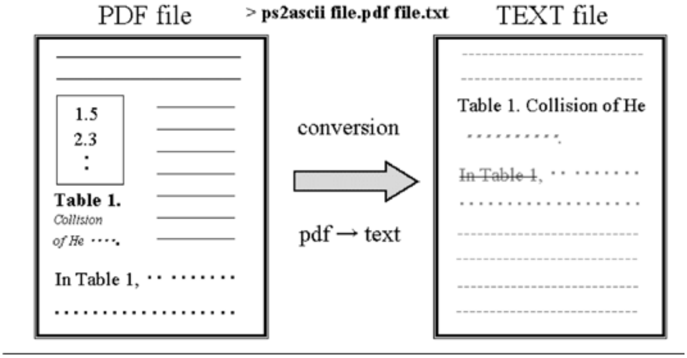


Fig. 4. Extraction of caption text from the pdf fultext of journal articles

useful to have also the html source for the fultext article. This is, in most cases, impossible (e.g. APS). Also, at present, there is no software (including Adobe (R)) for a reliable bug-free conversion from journal pdf articles into the html (doc, tex, etc.) format. In the following, we will show that a simple postscript based tool,

```
>ps2ascii file.pdf file.txt
```

which abandons all formatting, is up to the task in many cases, due to the simple algorithmic structure of finding relevant areas in fultext articles.

3.2 Algorithm for Relevance Matching

In order to analyze the fultext articles, we have to deal with various journal styles of major publishing houses. APS, IOP and EDP Sciences are the major journal publishers in the A+M field. It is a common standard that new research results are presented in the form of either tables or figures. That is where the information relevant to A+M databases appears. Fortunately, a simple string-matching based rule can be discovered for table and figure captions in each case. By searching for prefix-"Table N.", where N is an integer, and prefix≠"in " or "In ", we can identify all tables at EDP and IOP journals with 100% accuracy. APS, instead, identifies tables by the label "TABLE ", so that the string matching is even simpler. By taking the logical OR of the above two methods, the beginning of a table caption can always be identified. End of table caption coincides with the first subsequent end of line.

Similarly, figures in EDP journals are denoted with the prefix-"Fig. N.", where prefix≠"in " or "In ". In IOP journals, "Fig. N." is replaced with "Figure N.", otherwise the rule remains the same. Finally, APS figure indicator is again the simplest one, merely "FIG.". Also figure captions do not contain paragraphs and end up with the end of line. We therefore convert pdf files to ascii using the ps2ascii command on linux, and extract figure and table captions from the result. Then a search is performed for keywords relevant to A+M processes, e.g. "dissociative ionization" or "charge transfer" or "elastic scattering", and species names common in A+M papers “O” or “H” or “He” or “Ta” or “Fe”. If a hit in both categories is achieved in one caption (whether table or figure), the article is assessed as relevant and the processing stops.

The text extraction tool and caption matching is schematically explained in Fig. 4. The body text in the article is automatically omitted from the analysis of relevance. The text of captions extracted from each preselected article serves as an input for the relevance assessment algorithm, i.e. the caption-wise simultaneous lookup of process and species explained above and outlined in Fig. 5.

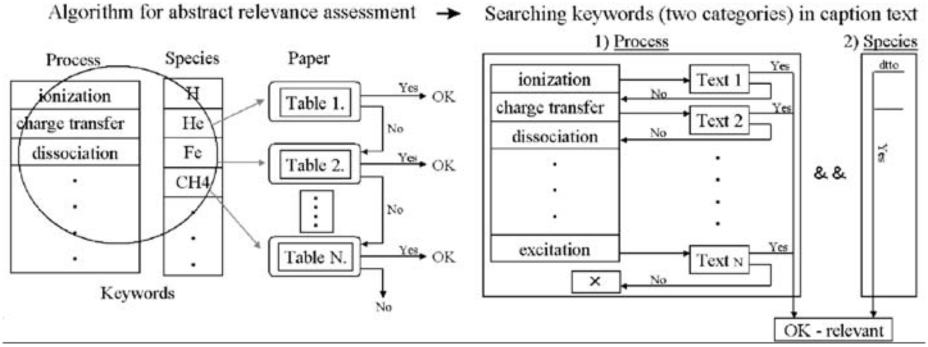


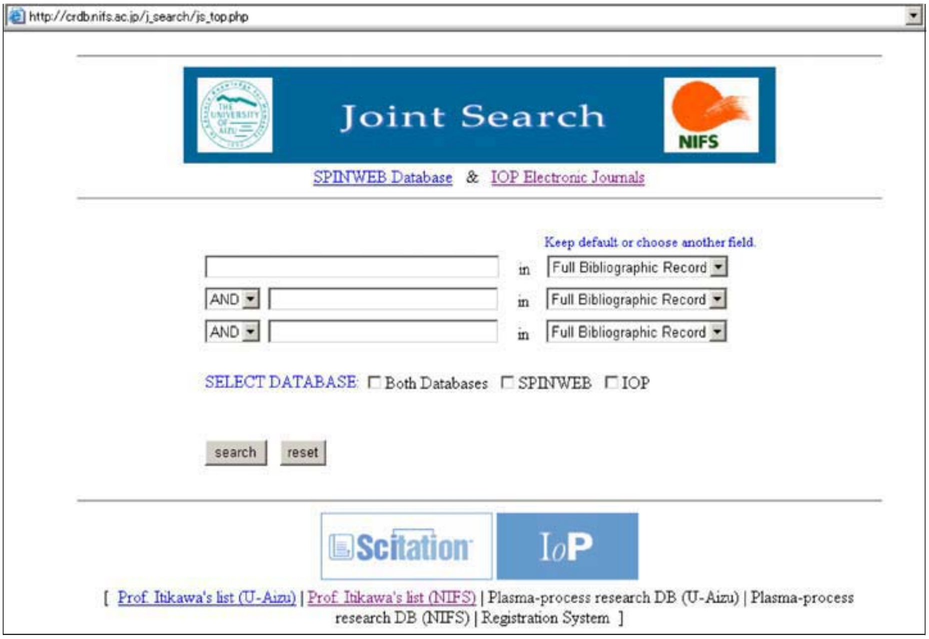
Fig. 5. Algorithm for simultaneous identification of relevant processes and species in the text of figure and table captions

The above simple procedure eliminates the need for grammar analysis and global comprehension of the text, because captions of figures and tables merely state the new relevant data. It is almost impossible that only old data are given in tables and figures, if the article reports on an original piece of research. Using a test set of 64 A+M relevant fultext articles and 103 irrelevant articles, we find that 52 keywords for A+M processes and 40 keywords for A+M species suffice for 90% accuracy. It is believed that by communicating to the specialists who collect A+M data at present, and by using a larger set of calibration data, the level of accuracy can reach near 100%, in principle. This was not possible with the present set of 167 journal articles, especially because of an occasional data loss in the ps2ascii conversion program.

The problem of relevance assessment belongs to the class of machine learning algorithms for information retrieval (IR) and has been intensively studied with regard to the capability of information storage and retrieval systems. Various modern approaches are based on neural networks (LVQ above, connectionist Hopefield network), symbolic learning (ID3, ID5R), genetic algorithms and other AI methods [6, 7, 8]. Applications of relevance assessment in IR to other fields (such as medicine) can also be found [9]. At present, we are not aware of any suitable A+M system to compare with. This is in part because of our strategy to combine fuzzy-like assessment (abstract relevance, which is based on condensed information) with rule-based assessment (caption relevance judgement that is especially useful for pre-selected articles). None of these two approaches is superior alone: abstracts do not always include sufficiently detailed information about the article text; caption extraction requires significant amount of disk space, connection bandwidth (download of pdf articles) and CPU time (text extraction). Major scientific bibliography databases available online do not offer AI features - these mostly rely on SQL queries (which include fultext analysis in some cases). Examples are the *INSPEC (R)* database or the *Web of knowledge (R)* [10].

4 Bibliographic Database with Autonomous Input

The system described above was implemented using a custom-built PC as a dedicated server. The hardware specification is as follows: Pentium 4 (FSB 533) 2GHz CPU, 1GB RAM and 120GB HDD, which matches the needs of research database users (large data set, limited access rates). The operating system is linux (Fedora Core 2) running Apache 2.0 HTTP server. The database management system is MySQL 3.23.54 with the connecting logic layer for web interface written in PHP 4.2.2. MySQL was chosen since it suffices for our purpose and for its ease of manipulation; more query-rich object relational database management system, such as postgress, can also be used. The above operating system,



http://crdb.nifs.ac.jp/j_search/js_top.php

Joint Search

SPINWEB Database & IOP Electronic Journals

Keep default or choose another field.

in Full Bibliographic Record

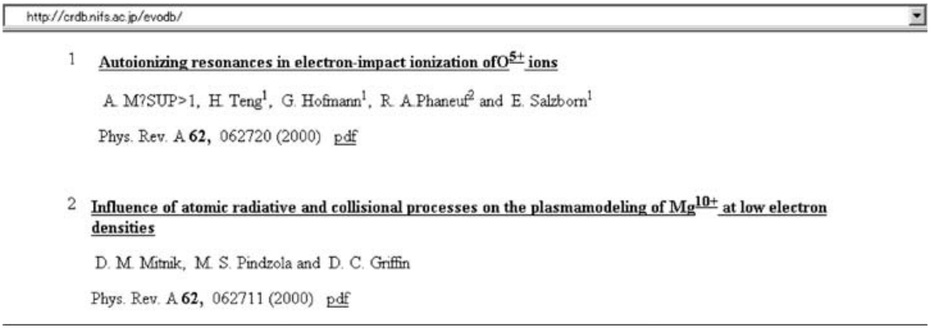
AND in Full Bibliographic Record

AND in Full Bibliographic Record

SELECT DATABASE: □ Both Databases □ SPINWEB □ IOP

[Prof. Itikawa's list (U-Aizu) | Prof. Itikawa's list (NIFS) | Plasma-process research DB (U-Aizu) | Plasma-process research DB (NIFS) | Registration System]

Fig. 6. Joint search tool developed on the basis of string matching programs for query distribution across the internet



http://crdb.nifs.ac.jp/evodb/

1 **Autoionizing resonances in electron-impact ionization ofO^{5+} ions**

A. M?SUP>1, H. Teng[1], G. Hofmann[1], R. A.Phaneuf[2] and E. Salzborn[1]

Phys. Rev. A **62,** 062720 (2000) pdf

2 **Influence of atomic radiative and collisional processes on the plasmamodeling of Mg^{10+} at low electron densities**

D. M. Mitnik, M. S. Pindzola and D. C. Griffin

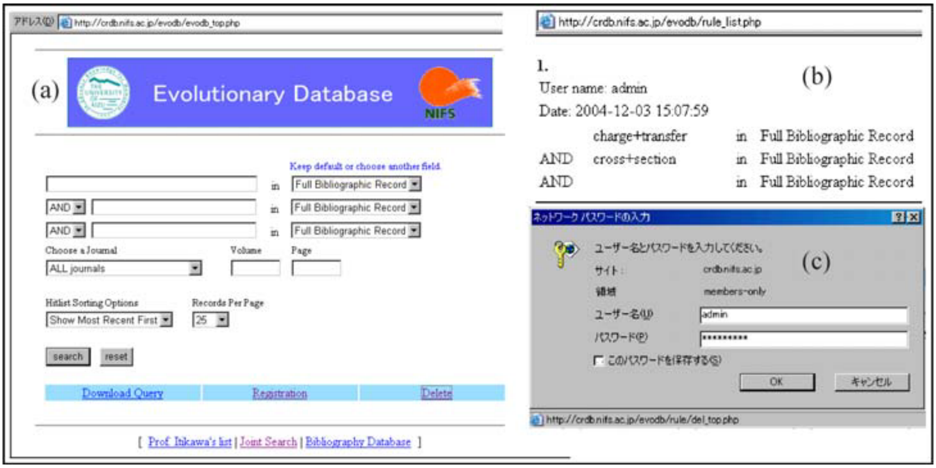
Phys. Rev. A **62,** 062711 (2000) pdf

Fig. 7. Data in the autonomous bibliography database which gradually retrieves and processes relevant abstracts from online publishers in preset time intervals

web and database server, and programming language compiler are free-software open-source products. The system has the following modules:

1. Online forms for registration (and deletion) of initial queries to be sent to the publisher website
2. Online interface for the autonomous database (search forms)
3. Database modules integrated with scripts for query scheduling, output retrieval, text analysis, extraction, and data input.

In weekly time intervals, queries are sent to the online publisher databases and the output is retrieved and analyzed. Since the search time period is also restricted to the weekly interval, the amount of matched articles is small (note that most online publishers use a listing cutoff of about 500 articles matching a query). The time restriction of each query also substantially decreases (although does not fully eliminate) the amount of simultaneous matches. These are eventually dealt with in a simple manner by the DB management system which does not allow for duplicate items. Because of the various html format that APS, EDP and IOP use, a lot of coding work is devoted to the analyzer scripts and string matching. Making use of such particular analyzers developed for APS and IOP, we have developed an online search interface that sends queries to, retrieves and displays the output from both publishers. This joint search tool is available online at NIFS [11]. Figure 6 shows its online interface. The data collected by this autonomous system consist of bibliography classification entries, html-formatted abstracts, and links to pdf fultext of each article as shown in Fig. 7 [12]. To prevent illegal interference with copyright issues, the pdf files (although downloaded from the publisher) are not made available online. The initial data set in the autonomous database coincides with that of the bibliography database by Y. Itikawa (private communication, cf. [1]) and expands in regular time intervals according to the set of custom queries which is administrated via a web query management form (Fig. 8 b, c).



アドレス(D) http://crdb.nifs.ac.jp/evodb/evodb_top.php

Keep default or choose another field

in Full Bibliographic Record
AND in Full Bibliographic Record
AND in Full Bibliographic Record

Choose a Journal: ALL journals
Volume
Page

Hitlist Sorting Options: Show Most Recent First
Records Per Page: 25

http://crdb.nifs.ac.jp/evodb/rule_list.php

1.
User name: admin
Date: 2004-12-03 15:07:59

	charge+transfer	in	Full Bibliographic Record
AND	cross+section	in	Full Bibliographic Record
AND		in	Full Bibliographic Record

ネットワーク パスワードの入力

ユーザー名とパスワードを入力してください。
サイト: crdb.nifs.ac.jp
領域 members-only
ユーザー名(U) admin
パスワード(P) **********
□ このパスワードを保存する(S)

http://crdb.nifs.ac.jp/evodb/rule/del_top.php

Fig. 8. The autonomous database system: (a) online search interface, (b) registered queries and (c) access for administration

5 Concluding Remarks

We have developed a free-software open-source system for A+M journal article abstract search over the internet. The abstracts are retrieved, processed, assessed for relevance, and stored in a custom bibliography database. In addition, we have found a simple but sufficiently robust solution for article A+M relevance assessment based on a pdf-to-text conversion tool and the analysis of extracted figure and table captions. The autonomous bibliography database which collects and inputs relevant data across publishers is available online, as well as the join search tool for abstract lookup at APS and IOP publisher archives. The present work is a step towards evolutionary A+M database systems which will identify, collect and input bibliography and numerical data without a need for human intervention.

Acknowledgements

The authors acknowledge a partial support by JSPS Grant-in-Aid. This work is also performed with the support and under the auspices of the NIFS Collaborative Research Program.

References

1. Itikawa, Y.: Annotated bibliography on electron collisions with atomic positive ions: excitation and ionization in 1995–1999. Atom. Data and Nucl. Data Tables. **80** (2002) 117–146.

2. Bhalla, S.: Evolving a model of transaction management with embedded concurrency control for mobile database systems. Information & Software Technology **45** (2003) 587–596.
3. van Bommel, P.: Experiences with EDO: An Evolutionary Database Optimizer. Data Knowl. Eng. **13** (1994) 243–263.
4. Ray, I., Ammann, P., Jajodia, S.: Using semantic correctness in multidatabases to achieve local autonomy, distribute coordination, and maintain global integrity. Inf. Sci. **129** (2000) 155–195.
5. Sasaki, A., Joe, K., Kashiwagi, H., et al.: Design and implementation of an evolutional data collecting system for the atomic and molecular databases, Joint ITC14 and ICAMDATA2004 Conference, Ceratopia Toki, Toki, Gifu, Japan, October 5-8, 2004.
6. P. E. Utgoff: Incremental induction of decision trees. Machine Learning **4** (1989) 161–186.
7. J. R. Quinlan. Learning efficient classification procedures and their application to chess end games. In Machine Learning, An Artificial Intelligence Approach, Pages 463–482 (Michalski, R. et al., Editors), Tioga Publishing Company, Palo Alto, CA, 1983.
8. G. Salton. Automatic Text Processing. Addison-Wesley Publishing Company, Inc., Reading, MA, 1989.
9. Chen, H., Lally, A., Zhu, B., Chau, M.: HelpfulMed, Intelligent Searching for Medical Information over the Internet, Journal of the American Society for Information Science and Technology (JASIST), **54** (2003) 683–694.
10. Commercial: http://www.iee.org/Publish/INSPEC/, http://www.isinet.com/.
11. Joint search (APS and IOP), http://crdb.nifs.ac.jp/j_search/js_top.php.
12. Autonomous bibliography database, http://crdb.nifs.ac.jp/evodb/evodb_top.php.

Development of a Data Mining Application for Huge Scale Earth Environmental Data Archives

Eiji Ikoma[1], Kenji Taniguchi[2],
Toshio Koike[2], and Masaru Kitsuregawa[3]

[1] Center for Spatial Information Science, The University of Tokyo,
Komaba 4–6–1, Meguro-ku, Tokyo, 153-8904, Japan
Phone: +81-3-5453-5690, Fax: +81-3-5453-5699
eikoma@csis.u-tokyo.ac.jp
[2] Graduate School of Engineering, The University of Tokyo,
Hongo 7–3–1, Bunkyo-ku, Tokyo, 113-8656, Japan
[3] Institute of Industrial Science, The University of Tokyo,
Komaba 4–6–1, Meguro-ku, Tokyo, 153-8505, Japan

Abstract. The amount of earth environmental data has increased explosively because of recent advances in observational techniques. However, such valuable data has not been used adequately because researchers cannot handle the volume.

In this study, we collaborated closely with some researchers in the field and developed a support system for analyzing various kind of natural phenomena that have been difficult to analyze in the past.

This was enabled by developing an analyzing toolkit for a huge amount of earth environmental data using the data mining technique, and developing an easy-to-use interface for non-computational researchers.

Moreover, we discuss some new knowledge that was acquired during the development of this system.

Keywords: Spatial DB, Temporal DB, Data Mining, Data Visualization, User Interface.

1 Introduction

Recently, there has been drastic progress in techniques for observing the earth environment, and as a result, the volume of remote sensing data such as satellite data and point data acquired from various kind of land observation instruments, has increased. Lots of useful and detailed data, which was very difficult to get before, are now available, and this contributes the progress of various research fields.

However, because of the sheer volume of new data, researchers have not been able to process much of it using traditional analytical techniques. Those not familiar computers have been unable to use the new data. Much of the important information collected in the past few years is simply being stored, then it has been sleeping at their storages.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 171–185, 2005.

In this study, in collaboration with researchers who are using those huge ammount of earth environmental data practically, we have developed a data mining system which targets the real data needed to supporting their analysis. We also developed user interface systems which enable researchers who are not familiar with computers to use our system easily.

Moreover, we provide a practical research example using the tools developed in this research.

2 Background

2.1 Increase in Available Earth Environmental Data and Archival Methods

The success of data gathering using satellite sensing has lead to the increase of the number of sensing satellites and the development of various new kinds of sensors.

The volume of earth environmental data acquired this way has increased. Moreover, in the fields of meteorology and climate change research, improvements in computer technology has enabled to analyze climate forecasts and create simulations with higher spatial and time resolutions.

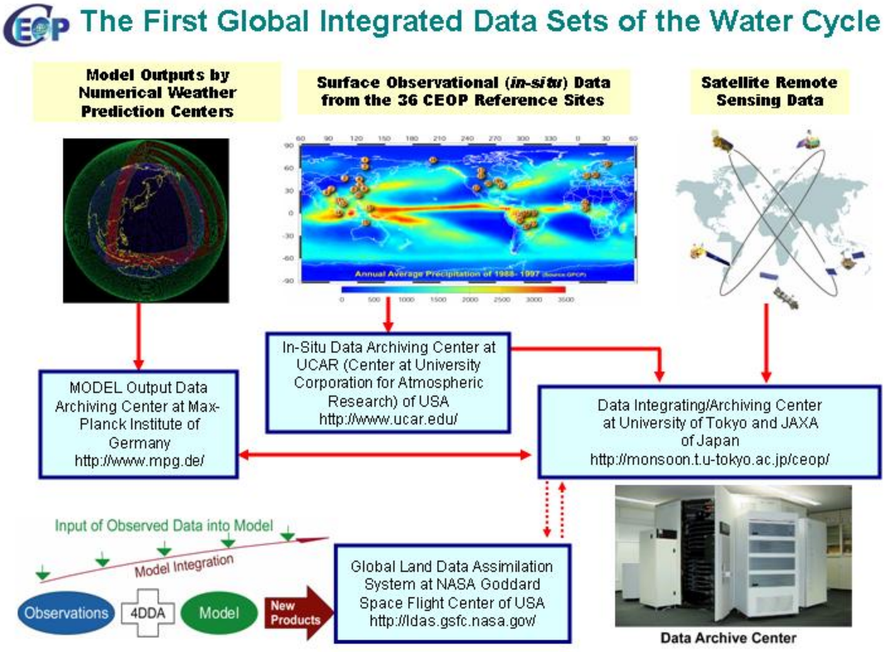

CEOP
The First Global Integrated Data Sets of the Water Cycle
Model Outputs by Numerical Weather Prediction Centers
Surface Observational (in-situ) Data from the 36 CEOP Reference Sites
Satellite Remote Sensing Data
Annual Average Precipitation of 1988-1997
MODEL Output Data Archiving Center at Max-Planck Institute of Germany http://www.mpg.de/
In-Situ Data Archiving Center at UCAR (Center at University Corporation for Atmospheric Research) of USA http://www.ucar.edu/
Data Integrating/Archiving Center at University of Tokyo and JAXA of Japan http://monsoon.t.u-tokyo.ac.jp/ceop/
Input of Observed Data into Model
Model Integration
Observations
4DDA
Model
New Products
Global Land Data Assimilation System at NASA Goddard Space Flight Center of USA http://ldas.gsfc.nasa.gov/
Data Archive Center

Fig. 1. CEOP Project

On the other hand, the amount of data which a researcher can handle per year is limited , and the amount is likely to continue to increase. Under these circumstances,unless something is done, lots of data will be left in archival format and not used in research. JAXA (Japan Aerospace Exploration Agency), NASA (National Aeronautics and Space Administration), ESA (European Space Agency), and the climate forecast centers in Asia, Europe, South and North America have collaborated to start CEOP (Coordinated Enhanced Observing Period). CEOP designated the period from Oct.2002 to Sep.2004 as an Enhanced Observation Period (EOP), and has collected huge amounts of data including satellite (remote sensed) data, predicted data from each participating country's numerical prediction division, earth observed data collected by operating meteorological agencies, and various kinds of data generated by other projects and organizations (See Figure 1).

The amount of satellite data collected during the CEOP period is almost 90 terabytes for one year; numerical predicted data and objective analyzed data is about 55 terabytes, with more observed data incoming. However, the research about how to analyze this huge amount of data, how to extract important information and how to acquire new knowledge from this data has not progressed adequately.

2.2 Correlation of Natural Phenomena

In meteorology, the methods to abstract the correlation between phenomena by analyzing statistics have been used before. The most general method is to see a correlation coefficient. For example, Wallace and Gutzler[3] checked the

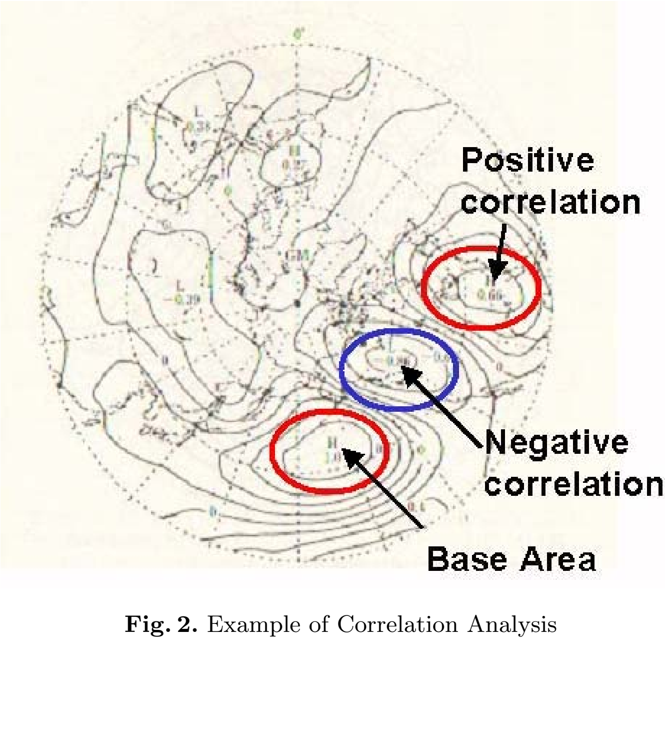


Fig. 2. Example of Correlation Analysis

distribution of correlation coefficients based on the fluctuation of the height of 500 hPa at the lattice point of (45N, 165E) . As a result, it is clear that a positive correlation area is located in the north Pacific Ocean and from the east coast to the central area of North America, with a negative correlation area located from the west coast to the central area of North America.

Figure 2 shows that in cases where the air pressure at the base lattice point becomes high, the air pressure at the positive (negative) correlation area becomes high (low). Because such distributions of correlation co-efficients is widespread from the Pacific Ocean to the North American coast, it has been dubbed the PNA(Pacific North America) Tele-connection Pattern .

Currently this phenomenon is understood as a stationary Rossby wave which was energized by a heat source in the equatorial zone dynamically, so the detection using the analysis of correlation is explained conceptually. As just described, the traditional approach is discover a phenomenon and try to formulate a theory to explain it. A newer approach is to use models and theories to actively search for correlations between phenomena.

3 Development of Data Mining Systems for Earth Environmental Data

Recently, advances in information technology have created computers which can process huge amounts of data quickly. However, at the time when users try to clarify the natural phenomena using earth environmental data as I mentioned before, one major problem is that it is not easy to use these computers for higher order computing or analysis. Indeed, when these users want to analyze data, most are not skilled enough with computers to use anything but low level, off-the-shelf software for a standard personal computer.

This means that even though there is a huge amount of data available, users have to decrease spatial and temporal resolution drastically, and which may mean the loss of new knowledge, or misinterpretation of data trends.

In this study, collaborating closely with researchers in earth environment fields , we developed tools which can analyze huge amounts of earth environmental data utilizing huge data archive systems. And we also developed a system that has a web-based interface which enables users to access the system in their research.

In this section, we describe the kind of data enabled on our system, analyzing tools we have developed, its user interface and usage of same, system structure, and visualization.

3.1 Target Phenomenon

The summer Asian monsoon has a great impact on the area, where there is a concentrated population and the economy is progressing rapidly. However, most of the physical processes of the seasonal progression or yearly variations in monsoons has not been unexplained yet.

Summer Indian monsoons have been studied for some time and it is understood that a strong west wind which blows across Indian subcontinent from the Indian Ocean and Arabian Sea brings rain by carrying moisture vapor to the Indian subcontinent.

However, this process that builds the west wind is not well understood. One of the reasons is that the temporal and spatial resolution of available data has not been high, so it restricts the ability to trace the daily variations adequately. Recent progress in sensing techniques and information technology has improved the quality and quantity of available data. In this study, as an example, utilizing these huge amounts of useful data, we work on analyzing the process of the Indian Monsoon focusing its start time.

3.2 Using Data

For analyzing the precipitation distribution in the summer monsoon term and seasonal change, we use monthly average data for global precipitation from 1979 to 2000 and daily average data from 1997 to 2000, which are in the GPCP (Global Precipitation Climatology Project) dataset. The resolution is 1 degree per pixel on every side .

Because OLR (Outgoing Long-wave Radiation) is useful to describe summer monsoons as an indication of convective activity in a tropical area, we use OLR data from 1975 to 2000 provided by NOAA (National Oceanic and Atmospheric Administrator). The resolution is 2.5 degrees on every side.

We also use re-analysis data of NCEP/NCAR (National Center for Environmental Prediction / National Center for Atmospheric Research) for atmospheric data, including as wind speed, geopotential height or atmospheric temperature. The resolution is also 2.5 degree on every side and frequency is daily.

For observing the seasonal cloud change we use the dataset provided by ISCCP (International Satellite Cloud Climatology Project) whose temporal resolution is three hours and spatial resolution is 2.5 degrees on every side.

We use sea surface temperature data provided by TMI (TRMM Microwave Imager) on TRMM (Tropical Rainfall Measuring Mission). This data has a 0.25 degree spatial resolution, and three day average temporal resolution from December 1997.

The total amount of data is about 900 GB.

3.3 Development of Correlation Analysis Tools

As mentioned in section 2.2, statistical analysis is the method of choice for extracting correlations between phenomena in meteorology. However, in the very large data sets that we targeted for this study, statistical analysis is quite difficult using general software. Moreover, the data we are targeting has not only simple correlations between two data, but also various other kinds of correlation such as spatial differences, temporal differences, and the difference of values; thus it requires tools which enable flexible handling and analysis and the ability to specifying conditions. In this study, we developed tools which can analyze the correlation while specifying conditions such as time(term), space(area), value,

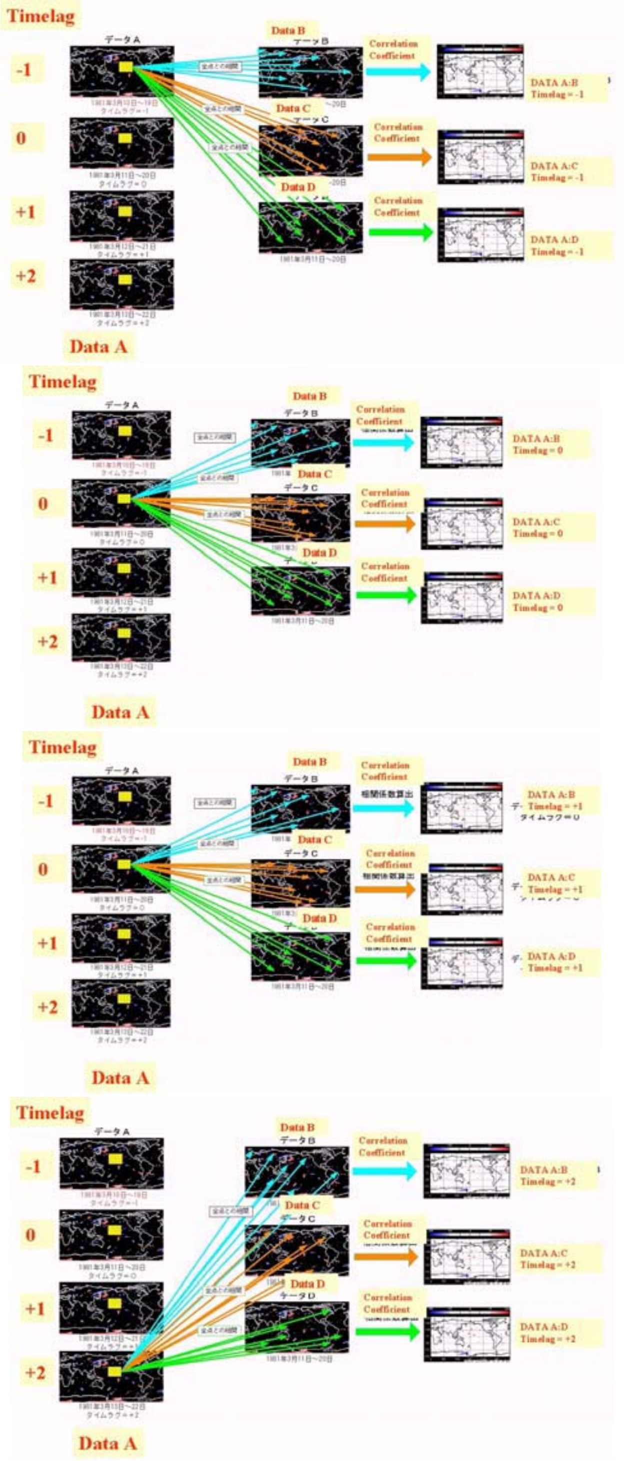


Fig. 3. System Structure

temporal resolution, spatial resolution, while coordinating with a huge scale data archive system. With these tools, assuming that some natural phenomena may happen not at the same time as the base phenomena but with some delay (time lag), a user can analyze the correlation between phenomena with different times (time lag correlation) (See Figure 3).
Example:

To analyze the data A (at lat.20N - lat.45N, lon.150E - lon.170E, 11th. Mar. 1981 - 20th. Mar.1981 time sequential data) :

1. Calculate the spatial average of base area , and make time sequential data as the correlation to base area
2. Prepare time sequential data from 11th. Mar. 1981 to 20th Mar. 1981, containing all point of the other data (target data)B, C, D. If those data have 1 degree spatial resolution, each data has 360 times 180 points.
3. Calculate the correlation between base data and all target data, and generate each point of correlation coefficient.
4. Next, analyze time lag correlation. As in step two prepare time sequential data from 9th. Mar. 1981 to 18th Mar. 1981 (2 day before), 10th. Mar. 1981 to 19th Mar. 1981 (1 day before), 12th. Mar. 1981 to 21st Mar. 1981 (1 day after) containing all point of the data B,C,D.
5. Calculate the correlation between base data and target data with time lag

Using this method, users can find not only the correlation between spatially separate points but also temporally separate points (such as a phenomenon happening a few days after the base phenomenon). We developed a Graphical User Interface (GUI) which can operate these tools via the Web. With this GUI, users can specify various conditions and visualize the results. It also allows multiple users to use it from remote locations.

3.4 System Structure

The system structure is shown in Figure 4.

The condition and processing order specified through the web interface by the user are sent to data mining processing and data visualization processing through a web server. Data mining processing generates SQL based on the conditions sent by the user, and sends the request for data to a database on the data archive system.

Acquired data is used for correlation analysis calculations in the data mining process based on these conditions, and the results are then sent to data visualization processing.

There, the data is processed to visualize based on the conditions sent from the web server, and show the results on the web interface via the web server where user can check the result on web interface.

Hardware and software used in this system are shown on Table 1.

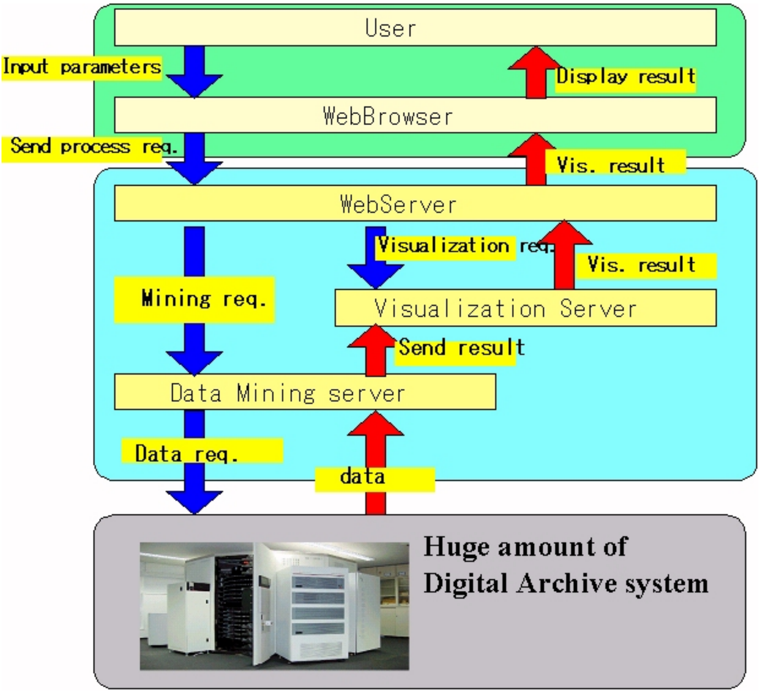


Fig. 4. System Structure

Table 1. Hardware and Software

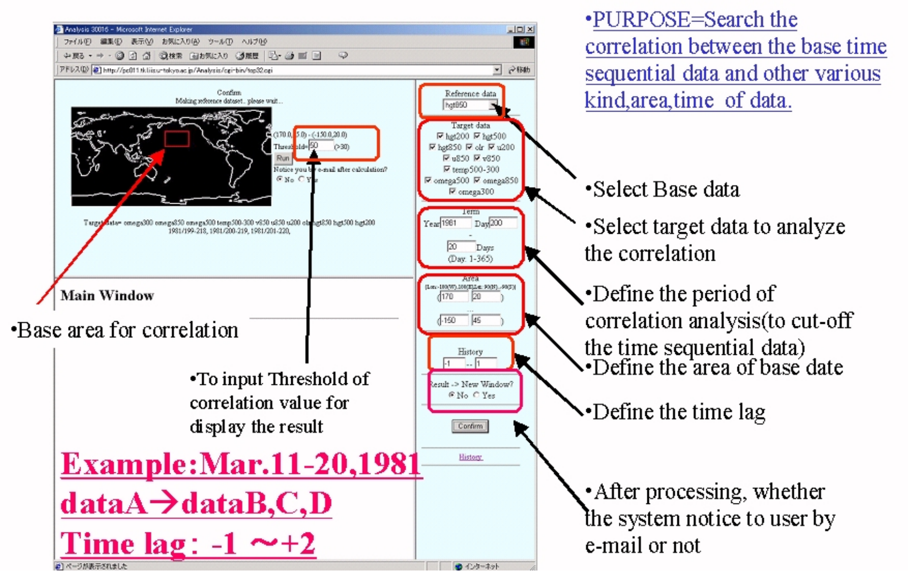
WebServer, Data Archive Server	Sun Enterprise 6500 (UltraSPARC-336MHz × 8,4GB Memory)
Data Mining Server	Dell Poweredge1600c (Redhat Linux9,Xeon-3.2GHz × 2,4GB Memory)
Data Visualization Server	SGI ONYX4 Ultimate Vision (IRIX6.5,800MHz-R16000 × 10, 4GB Memory)
Data Archiver	Sun StorEdge A5000(250GB), STK WolfCreek 9960 + RedWood SD-3 System(0.5PB)
Visualization Display (Huge size display wall)	MITSUBISHI LVP-FD10(50inch x 15)
RDBMS	PostgreSQL 6.5.4
other	apache httpd,csh,awk,sed,c,java1.0,vrml2.0 etc.

3.5 Correlation Analysis System

When users log on our system, Top Page (Figure 5) will appear.

In the right frame, user can specify as follows:

- Choose 1 base data for correlation analysis
- Choose some (multiple) target data for correlation analysis


Analysis 30016 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る 検索 お気に入り 履歴
アドレス(D)
移動
Run
Confirm
History
ページが表示されました
インターネット

http://pc011.tk.iis.u-tokyo.ac.jp/Analysis/cgi-bin/top32.cgi

Confirm
Making reference dataset.. please wait...

(170.0, 5.0) - (-150.0,20.0)
Threshold= 50 (>30)
Notice you by e-mail after calculation?
● No ○ Yes

Target data= omega300 omega850 omega500 temp500-300 v850 u850 u200 olr hgt850 hgt500 hgt200
1981/199-218, 1981/200-219, 1981/201-220,

Main Window

Reference data
hgt850

Target data
☑ hgt200 ☑ hgt500
☑ hgt850 ☑ olr ☑ u200
☑ u850 ☑ v850
☑ temp500-300
☑ omega500 ☑ omega850
☑ omega300

Term
Year 1981 Day 200
-
20 Days
(Day: 1-365)

Area
[Lon:-180(W)..180(E),Lat: 90(N)..-90(S)]
(170 20)
...
(-150 45)

History
-1 .. 1

Result -> New Window?
● No ○ Yes

•PURPOSE=Search the correlation between the base time sequential data and other various kind,area,time of data.

•Select Base data

•Select target data to analyze the correlation

•Define the period of correlation analysis(to cut-off the time sequential data)

•Define the area of base date

•Define the time lag

•After processing, whether the system notice to user by e-mail or not

•Base area for correlation

•To input Threshold of correlation value for display the result

Example:Mar.11-20,1981
dataA→dataB,C,D
Time lag: -1 ~+2

Fig. 5. Toppage of our system

– Year, Month, Date
– Term
– Specify base area for base data
– Choose the window for displaying results (Same window of this page or other)

Then, the user's specified area on a world map will display in the upper left frame for checking the area, and user will be asked the value of the threshold for visualization and the mode for processing (whether system will start the process and display results immediately or the system will notify the user by e-mail of the results after processing) at the same time.

After answering and pushing the start button, the process will start and the results (Figure 5) will be displayed.

In this window, the results of different days of lag are lined up at horizontal direction, and the results of different value are lined up at vertical direction. In other words, one result data is the result of calculated correlation coefficients between the time sequential area average data of the base area of the base data, and all points of the world of one target data of one time lag period, year, month, date. Positive correlation points are shown as red points, and negative ones are shown as blue points, the darkness of color means it is strong and weak.

Using this method, the user can understand the change of correlation as time goes on by comparing the results laterally - it means that how much time lag has an effect on this result, and which data has a strong correlation with the base data by comparing the result at vertical direction.

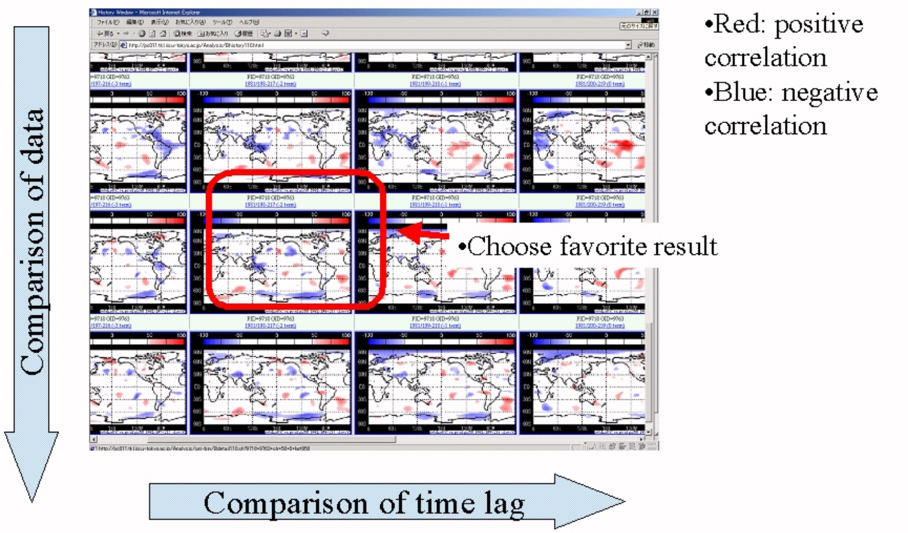
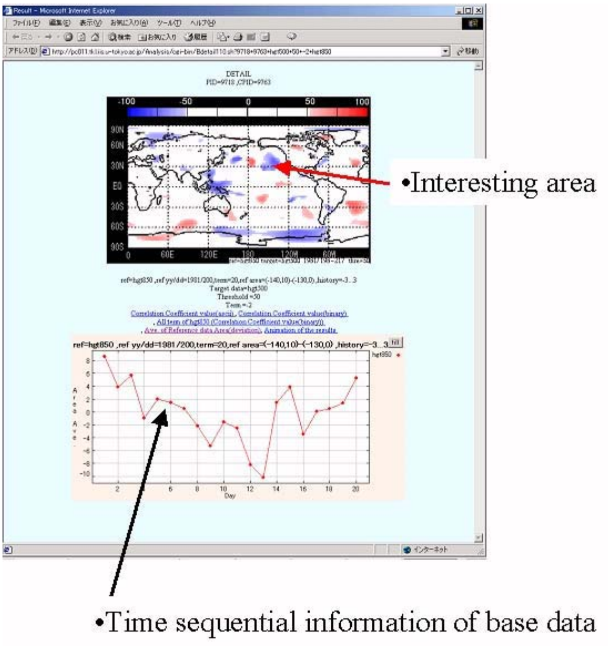


Fig. 6. Result of correlation analysis



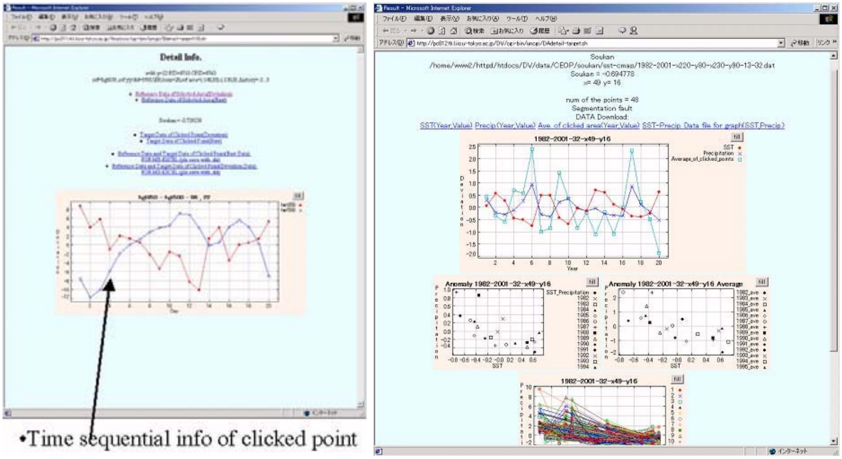
http://pc011.tkl.iis.u-tokyo.ac.jp/Analysis/cgi-bin/Bdetail110.sh?9718+9763+hgt500+50+-2+hgt850

DETAIL
PID=9718 ,CPID=9763

ref=hgt850 ,ref yy/dd=1981/200,term=20,ref area=(-140,10)-(-130,0) ,history=-3..3
Target data=hgt500
Threshold =50
Term =-2
Correlation Coefficient value(ascii), Correlation Coefficient value(binary)
, All term of hgt850 (Correlation Coefficient value(binary))
, Ave. of Reference data Area(deviation), Animation of the results

ref=hgt850 ,ref yy/dd=1981/200,term=20,ref area=(-140,10)-(-130,0) ,history=-3..3

Fig. 7. Result of 1 detail data



Detail Info.

Soukan
/home/www2/httpd/htdocs/DV/data/CEOP/soukan/sst-cmap/1982-2001-x220-y80-x230-y80-13-32.dat
Soukan = -0.694778
x= 49 y= 16

num of the points = 48
Segmentation fault
DATA Download:
SST(Year,Value) Precip(Year,Value) Ave. of clicked area(Year,Value) SST-Precip Data file for graph(SST,Precip)

•Time sequential info of clicked point

Fig. 8. Result of detail analysis

If a user clicks interesting data on this window (Figure 5), the window shown at Figure 6 will be displayed.

In this window, the chosen correlation analysis result at a higher resolution is displayed, and the graph of area average time sequential data of the base data is displayed in the lower window. If the user clicks a point such as one with very high correlation on the global map of this window, Figure 7, 8 will be displayed to confirm more detailed information on each point and each time.

3.6 Data Visualization on Huge Display Wall

We are assuming that our system will be used by researchers who are not familiar with high performance computers. So, we use a web browser for visualization because it is easy to use and commonly found on most personal computers. We are also developing the result visualization system for huge displays (Display Wall- Figure 9)

Because the display wall can allow for much higher resolution than a standard size computer screen, we can show and users can examine much more detail or more data at once. This system is still under development with the help of researchers who have expertise in earth environmental science, but even in these early stages, when those researchers tried system, they found some phenomena which were not available through traditional methods. We are planning to continue this research with them to find other new phenomena.

3.7 Contribution to Earth Environmental Research

As I mentioned at 2.1, our system is collaborating with CEOP Project and using the data collected by CEOP members(Show 3.2). This system is originally started to be developed by the request of them. Now, some of the researchers

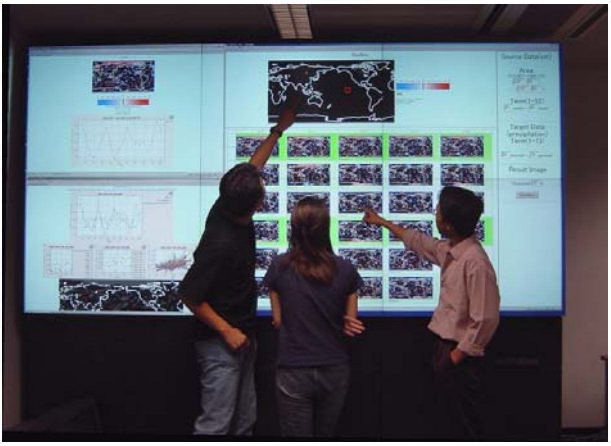


Fig. 9. Visualization on Display Wall

belonging with CEOP Project are using our system and they have already found lots of beneficial result for Earth Environmental research. In next section, we describe an example result briefly.

4 Applications for Practical Research - Analysis of Indian Monsoon

Now, actual research about Indian monsoon is in progress in the form of the “Earth Water Cycle Informatics” Project (carried out as one of the Special Coordination Funds for Promoting Science and Technology funded by the Ministry of Education , Culture, Sports, Science and Technology of Japan).

In this section, we describe this research briefly.

4.1 Background

In order to plan stable and effective water resource management, adding to the longitudinal climate change forecast, high accuracy in medium and short term weather forecasts is required.

In Asia, the summer Asian monsoon has a great impact, but most of the physical processes of season progress and year by year variation are poorly understood.

Therefore, we start our example analysis using the Indian monsoon, as it’s processes are somewhat better known. We focusing on the start of the Indian monsoon Season using our tools and various kinds of useful data available thanks to remote sensing techniques.

Table 2. The monsoon onset day and feature of each year

Year	Onset (pattern)	Year	Onset (pattern)
1981	Jun.4 (Non-Cyclone)	1992	Jun.16 (Non-Cyclone)
1982	Jun.7 (Non-Cyclone)	1993	Jun.4 (Non-Cyclone)
1983	Jun.16 (Cyclone)	1994	Jun.7 (Cyclone)
1984	Jun.1 (Non-Cyclone)	1995	Jun.10 (Non-Cyclone)
1985	May.26 (Non-Cyclone)	1996	Jun.10 (Cyclone)
1986	Jun.8 (Cyclone)	1997	Jun.16 (Non-Cyclone)
1987	Jun.3 (Cyclone)	1998	Jun.9 (Cyclone)
1988	Jun.4 (Non-Cyclone)	1999	May.19 (Cyclone)
1989	May.29 (Non-Cyclone)	2000	May.16 (Non-Cyclone)
1990	May.18 (Cyclone)	2001	May.17 (Non-Cyclone)
1991	Jun.5 (Cyclone)	2002	Jun.15 (Non-Cyclone)

4.2 Outline of This Research

In this research, we analyze the example in detail using about 22 years of multivariate daily average data input into our system using our tools as described above. The details are discussed in [4].

As a result of our analysis of the west wind which brings the rain in the summer Indian monsoon season, it is clear that there are two factors involved in the start of the season . One is the change of atmospheric field according to the cyclone. The other is by generating the field to have west wind gradually without any special event.

Table 2 is the result of the analysis of the seasonal changes until the constitution of west wind from 1981 to 2002, and whether it results from a cyclone or not. The beginning time of the monsoon is defined as the day of constitution of west wind, and it is also the day when the average wind speed of the area at lat.10N - lat.15N, lon.60E - lon.70E and 850 hPa. It is also referring to the plot data of the wind system.

Nine examples in this 22years are caused by cyclone. Cyclone causes the rapid change in atmospheric field , but the time and place of onset is different each year. On the other hand, where the cause is not cyclonic, there are some common characteristic points in the seasonal progression in 2001.

Briefly, a north west wind over the west coast of Middle East Asia and the Indian subcontinent is generated by the heat of the Indian subcontinent and the Iran - Pakistan area, and it carries warm air over the land, heated up over the Arabian Sea. The heat of the Arabian Peninsula progresses sufficiently, then the south west wind which blows from Eastern Africa to the Arabian Sea caused by the temperature inclination between the Arabian Sea and Peninsula, and finally the temperature of the Arabian Sea falls again. By increasing the temperature inclination between the warm area (Arabian Peninsula area and Iran - Pakistan area) and cold area (Arabian Sea), a strong west wind is generated on the Arabian Sea. In other words, for monsoons to be generated these two temperature inclinations between Arabian Peninsula and Arabian Sea, and between

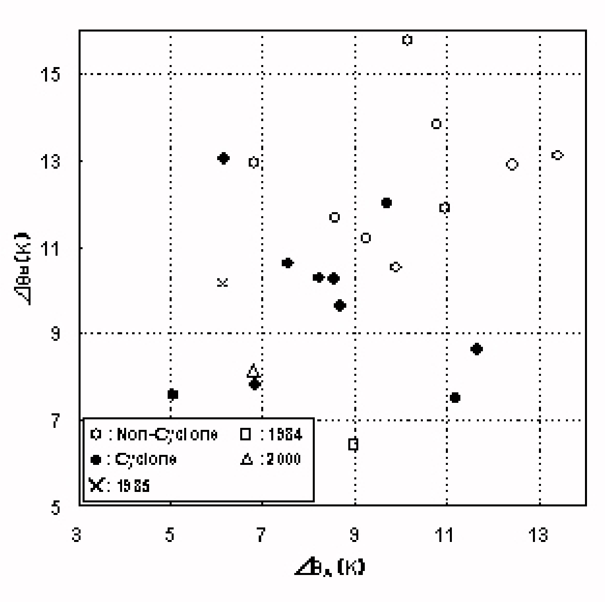


Fig. 10. Temperature inclination of the monsoon onset

Middle-East Asia and Arabian Sea are necessary to build up the west wind over India.

Figure 10 shows the onset day of monsoons caused by the two factors as described above.

The temperature inclination of the monsoon onset in the case of cyclones is smaller than in case of not cyclonic. That is to say, under the situation where temperature inclination becomes large enough, a west wind has already been generated. Whether a cyclone rises or not has an influence on those two patterns of west wind generation.

Consequently, we learned that if we can predict the rise of cyclones when the temperature inclination is not enough large we may be able to to predict the west wind initiation, which in turn means the beginning of an Indian monsoon.

5 Conclusion

Progress in sensing technology in the field of climatology and meteorology, which uses earth environmental data, has enabled researchers to find new knowledge because of an increase in data volume. It has also brought with it some problems regarding methods to use those data effectively and environments to use them practically.

In this study, collaborating closely with the researchers who are using earth environmental data, we used real data in our huge scale data archiving system and developed analyzing tools with data mining techniques.

We also developed a web based interface on it to support their specialist analyses by building the special environments for the researchers.

As a result, the research about summer Indian monsoons revealed two seasonal progress patterns required for the process of building up, which has not found before.

In the future we will develop analyzing tools which enable more high dimensional methods, and also develop more flexible and effective tools using the feedback opinion from active users.

Moreover, using the fact that practical research is progressing on our system, we will also try to learn from its usage logs.

Acknowledgements

This study was carried out as one of the Special Coordination Funds for Promoting Science and Technology funded by the Ministry of Education, Culture, Sports, Science and Technology of Japan.

References

1. CEOP Homepage(`http://www.ceop.net/`).
2. National Centers for Environmental Prediction (`http://ncep.noaa.gov/`).
3. Wallce, J.M. and Gutzler D.S., "Teleconections in the geopotential height field during the Northern hemisphere winter", Mon. Wea. Rev., vol.109,pp.785-812, 1981.
4. Kenji Taniguchi, Toshio Koike, Eiji Ikoma, Masaru Kitsuregawa "The research about the inclination of Summer Indian Monsoon using integrated data set", Jounal of the Society of Civil Engineering, Vol.48, Feb. 2004.
5. Webster, P. J.,Magana, V. O, Palmer, T. N., Shukula, J., Thomas, R. A., Yanai, M. and Yasunari, T.: Monsoons:Process, predictability, and the prospects for prediction, J. Geophys.. Res., Vol.03, pp.451-510, 1998.
6. Chelliah, M. and Arkin, P.: Large-scale variability of monthly longwave radiation anomalies over the global tropics, J. Climate, Vol.5, pp.371-389, 1992.
7. Schiffer, R.A., and Rossow, W.B.: The International Satellite Cloud Climatology Project (ISCCP): The first project of the World Climate Research Program, Bull. Amer. Meteor. Soc., Vol.64, pp.779-784, 1983.

Design of Automation Systems for Web Based Courseware Using XML and XSLT

Yukari Shirota

Faculty of Economics, Gakushuin University,
1-5-1 Mejiro, Toshima-ku, Tokyo 171-8588, Japan
yukari.shirota@gakushuin.ac.jp

Abstract. Reported is our approach for Web based mathematical courseware generation. To decrease development costs of the courseware, we have developed a system -- e-Math Interaction Agent -- that automatically generates learning materials using Semantic Web technologies, such as XML and XSLT. Knowledge databases containing math formulas and basic economic knowledge form the core mechanism of the system. Given the necessary mathematical problem definition data, the system can automate the target courseware by using these knowledge bases. The system differs from existing courseware automation systems in that it features (1) interactive dialogues with a virtual character that are pre-programmed into the XSL stylesheets, (2) a solution plan and calculations that are automated from a knowledge base of mathematical formulas and economic rules, and (3) mathematical software that generates mathematical expressions in MathML format and image files. My final goal is to formalize a teaching model for a wide range of mathematical problems that includes how to solve the problems and interactively and visually guide students.

1 Introduction

Today, an increasing number of universities use distance learning systems that leverage the World Wide Web. However, teachers developing the corresponding learning materials face a cost problem in that the work takes much practice and devotion on their part. To solve this issue, we have developed a system -- e-Math Interaction Agent – that automatically generates learning materials using Semantic Web technologies, such as XML and XSLT. Our target field is economical mathematics. Thus, knowledge databases containing mathematical formulas such as differentiation and integral rules and basic economic knowledge form the core mechanism of the system. Given the necessary mathematical problem definition data, the system can automate the target courseware by using these knowledge bases.

The system differs from existing courseware automation systems in that it features (1) interactive dialogues with a virtual character that are pre-programmed into the XSL stylesheets, (2) a solution plan and calculations that are automated from a knowledge base of mathematical formulas and economical rules, and (3) mathematical software that generates the mathematical expressions in MathML format and image files. My

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 186–203, 2005.

final goal is to formalize a teaching model for a wide range of mathematical problems that includes how to solve the problems and guide students. When teachers use our system they will be released from tedious XML programming activities and thus able to devote their energies to more creative work.

The Semantic Web is a revolutionary new framework for creating intelligent software applications that automate reasoning and decision-making processes[1,2,3]. These technologies include Extensible Markup Language (XML)[4], Web services, the Resource Description Framework (RDF)[5,6], Extensible Stylesheet Language Transformations (XSLT)[7]. Today, Web pages are often automatically updated using XSLT stylesheets to which data is input in the form of XML files. For example, weather information and stock market sites are automatically updated. XML is used to handle content data. XML documents are validated by a Document Type Declaration (DTD) document or an XML schema. XML has its own style language called Extensible Style Language (XSL). It provides a standard way of extracting what information in an XML document should be included in the presentation, and expressing how this information should be presented. XSL consists of two parts, a transformation language named XSLT and a formatting language named XSL FO. XSLT is used to transform documents into different forms, and an XSLT stylesheet is used to specify the exact format of the presentation. XSLT technologies are also used in some mathematical courseware generation systems such as ActiveMath[8] and WME[9].

In general, the advantages of automatic updates by XML and XSLT are their low cost and speedy turnaround time. Meanwhile, their disadvantage is that they produce Web pages of uniform appearance whose content, layout and generated dialogues can thus be tedious for the viewer.

Our research target is an automation of courseware in an economical mathematics field. When teaching an economical mathematical problem, a human teacher should explain the economical and mathematical relationship in various ways and from various angles:

* In words, and using visual materials, such as graphs.
* As an economical relationship, and as a mathematical relationship.
* Using mathematical symbolic calculations and concrete value calculations.

In the generated courseware, too, the virtual teacher has to explain the relationships repeatedly and in various ways. In addition, many graphs should be automated so that students can see the relationships interactively and visually from various angles.

We have been building our e-Math distance learning system for economical mathematics since 2001[10,11]. The system is currently published on the Web, with Web-based learning materials available to registered students within the campus network. The goal of the system is to simulate the cleverest teacher's guidance, interaction, and dialogue with students. To this end, we have been developing the e-Math Interaction Agent as an extension of our e-Math distance learning system.

This paper describes how the e-Math Interaction Agent dynamically automates Web-based materials to be presented interactively on Web browsers. In the next section, the existing related work will be described. Then, we will explain the design principles and a system model of our proposed courseware automation. In Section 4, the developed system architecture will be described. In Section 5, the prototype system that automates learning materials to teach optimization problems in mathematics will be shown. Discussions and conclusions are given in the last section.

2 Related Work

In the section, we survey various mathematical courseware related works. They are divided to three groups: (a) publishing mathematical expression work, (b) representation work of knowledge on mathematical formulas, and (c) mathematical application work.

First let me explain about works to make publishing mathematical expressions on the Web. MathML[12] is an XML application for markup of mathematical expressions and a low-level specification for describing mathematics as a basis for machine to machine communication. It provides a much needed foundation for the inclusion of mathematical expressions in Web pages. The transformation systems from the input text-based or LaTeX-based math formats to the MathML-based math formats are IBM's Techexplorer[13], Yuan's system[14], and Design Science's MathType and WebEQ[15]. WebEQ provides a Java applet to display WebTeX and MathML in a Web browser. We can say that the main purpose of these systems is to make mathematical documentations on the Web.

OpenMath[16],[17] and OMDoc[18] are standards for describing mathematical formula semantics. OpenMath is a standard for representing mathematical objects with their semantics, allowing them to be exchanged between computer programs, stored in databases, or published on the Web[19,20]. OpenMath has a strong relationship to the MathML recommendation from the Worldwide Web Consortium[21]. MathML deals principally with the presentation of mathematical objects, while OpenMath is solely concerned with their semantic meaning or content. OMDoc is an extension of OpenMath. OMDoc has been widely used as a standard for mathematical knowledge representation[22,23,24].

In a higher application layer, making interactive Web page functions are implemented by Maplets[25], WebMathematica[26], White's Mathwright Microworlds[27], Wang's WME (Web-based Mathematics Education) framework[28], and ActiveMath[29]. The generated Web pages provide interactive calculations and interactive graph drawing functions. Maplet is a simple specification of Maple's graph drawing functions. The goal of the WME framework is to establish open standards of distributed system on the Web for mathematics education. This WME framework uses XSLT stylesheets and is similar to ours. The educational content pages in WME framework are written in their defined language MeML (Mathematics Education Markup Language)[30]. Those that may be considered related to MeML are OpenMath, DocBook[31], OMDoc, MathBook [32], and SCORM[33]. MathBook is intended to be a DTD simplified from DocBook and OMDoc for Web-based math learning. SCORM (Sharable Content Object Reference Model) is a standard for curriculum and reuse of learning materials[34]. The main difference between our proposed method and WME is that our method is a meta-level description to define the mathematical solution plan and the latter is a language to describe mathematical courseware on Web pages. In our system, objects to be created and the creation process are defined in a solution plan and a guidance plan of the problem type. Input the small input data specific to the problem, the needed learning materials are generated. While WME's goal is the open architecture of the system, our goal is to generate teachers' dialogues from various viewpoints for student guidance. Therefore, mathematical concepts/semantics used in teachers' explanations are in advance defined in a solution plan. The concepts are defined as metadata, so that the necessary format translations can be operated such as a mathematical equation, a concrete value, and a concept name.

ActiveMath is a Web-based learning system which can dynamically generate interactive mathematical courseware, using XSLT technologies like our e-Math Interaction Agent system. However, AcriveMath is a big project of which educational environment integrates several mathematical systems for calculation and proof such as the proof planner ΩMEGA[35] and the Computer Algebra Systems MuPAD (Multi Processing Algebra Data Tool)[36], and Maple. The knowledge representation on which ActiveMath focuses corresponds to metadata of OMDoc and OpenMath, namely, the concepts in mathematical formulas. On the other hand, our target concepts are mathematical terms and equations appeared in mathematical word problems and more content based[11].

3 Design of Automation Systems

One of the key goals of a distance learning system is to help students interactively and naturally. This goal is, however, difficult to achieve. Even the cleverest teacher requires much time, practice, devotion, and patience. To develop well-designed teacher-student interactions, we first needed to find some exemplary teaching principles. We decided to take Professor Pólya's advice because he made important contributions to mathematical education[37,38,39,40]. The framework Pólya suggested has been extended by Schoenfeld[41]. However, there had been no implementation of his framework in mathematical educational systems. Currently, ActiveMath group and Cairns have been developing their systems[42,43,44] like us.

Pólya recommended that the procedure for solving a mathematical problem consist of the following phases:

(1) Understanding the problem.
(2) Devising a solution plan.
(3) Carrying out the plan.
(4) Checking the results.

This is a natural flow in solving a problem.

The final goal of this research is to formalize a teaching model for a wide range of math problems, including how to solve the problems and guide a student. Our target math problems are "problems to find," not problems to prove. The aim of a "problem to find" is to identify a certain object, the unknown of the problem. If we wish to solve a "problem to find," we must first be familiar with its principal parts, specifically the unknown, the data, and the condition:

(1) What is the given data?
(2) What is the given condition?
(3) What is the unknown?

Then, we must try to do the following things to solve the problem:

(4) Find the relationships between the data and the unknown.
(5) Separate the various parts of the condition.

This framework is common for all "problems to find."

On the other hand, the principle parts of a "problem to prove" are the hypothesis and the conclusion of the theorem that has to be proved or disproved. The common framework for solving a "problem to prove" is defined as follows:

(1) What is the hypothesis?
(2) What is the conclusion?
(3) Find the relationships between the hypothesis and the conclusion.
(4) Separate the various parts of the hypothesis.

"Problems to find" are more important in elementary mathematics and economical mathematics, while "problems to prove" are more important in advanced mathematics. In addition, the "problem to find" is in general much easier to teach. Thus, we have decided firstly to study a teaching model for "problems to find."

Next, we shall explain our proposed model of general solution processes for math problems (See Figure 1). The input data is the definition data of a math problem. Suppose that the math problem is named "Problem A." We wish to automatically generate a solution plan specific to Problem A. Namely, the output of the automatic generation process is the learning material specific to Problem A. The core parts of the process are the "general solution plan model" and "the general model of teacher interaction" that are represented

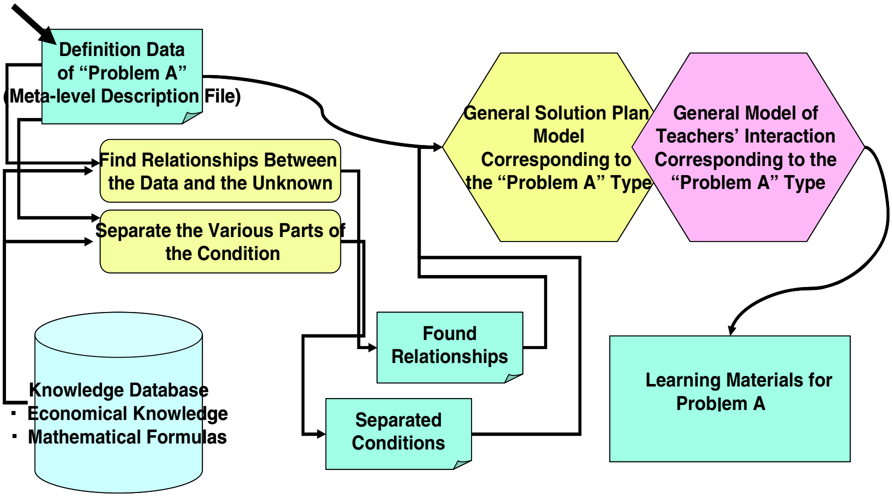


Fig. 1. The automatic generation model of courseware using the general solution plan model

by the two hexagons (See Figure 1). The “general solution plan model” describes how to solve a problem of the same or a similar type. The “general model of teacher interaction” defines what a virtual teacher dialogues with a student and how the virtual teacher guides a student. It is also defined for the same or similar types of problems.

These two models must be defined in advance by a “system supervisor” who is both a computer expert and a math teaching specialist well-versed in solving the problems and teaching them to students. The “system supervisor” at first divides the math problems into categories. Next, each category is assigned a type name. The following are typical type names:

(a) Optimization problem of single variable functions.
(b) Optimization problem of multivariable functions.
(c) Constrained optimization problem with Lagrange multipliers.
(d) National income determination modeling problem*.

For each problem type, the “system supervisor” creates the above-mentioned “general solution plan model” and “general model of teacher interaction” in advance. Various versions of the “general solution plan model” and ones of the “general model of teacher interaction” may be created depending on the system supervisors, even if the problem type is the same one.

4 System Architecture

In this section, we will outline our e-Math Interaction Agent.

Figure 2 illustrates the data flow of the e-Math Interaction Agent. The Interaction Agent is a system that can automatically generate Web-based learning materials from a meta-level description file. Teachers have only to write the meta-level description files. The Interaction Agent then repeatedly generates and presents learning materials. For example, let us consider an optimization problem that determines the maximum profit of a firm with a given demand function and an average cost function. The meta-level description file is shown in Figure 3. When this file is input, the XML-based learning materials from Figure 4 to 7 are automatically generated and displayed. As shown in Figures 4 to 7, the learning material contains many mathematical expression images.

As shown in Figure 2, the solution plan corresponds to the above-mentioned “general solution plan model” illustrated in Figure 1. A number of solution plans are stored in a solution plan database as a part of the knowledge base. The main process of the Interaction Agent is written in Perl, which is a CGI program. The Interaction Agent system includes the following four sub-modules:

(1) Inference Engine: Prolog Interpreter.
(2) Mathematical Software.
(3) Equation Server.
(4) Web Page Generator.

* The nation income determination problem is a typical economic application.

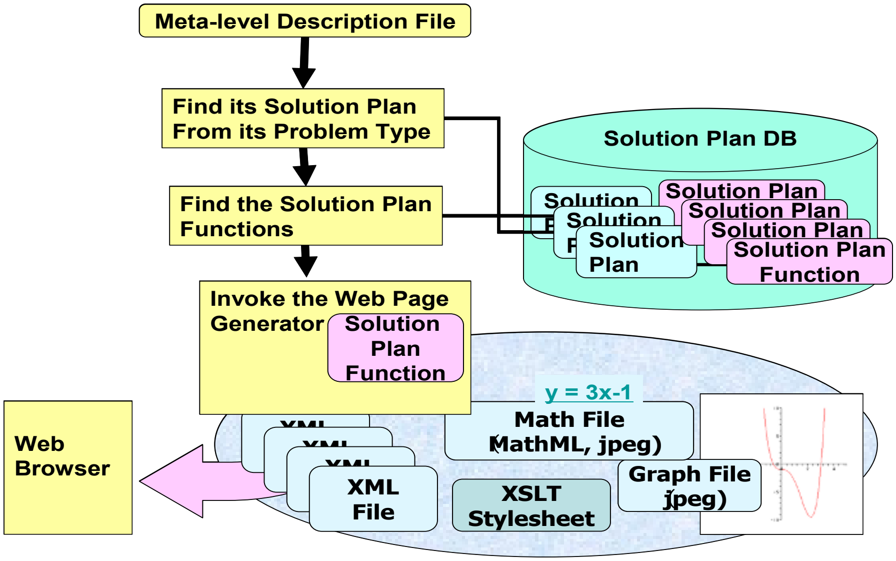
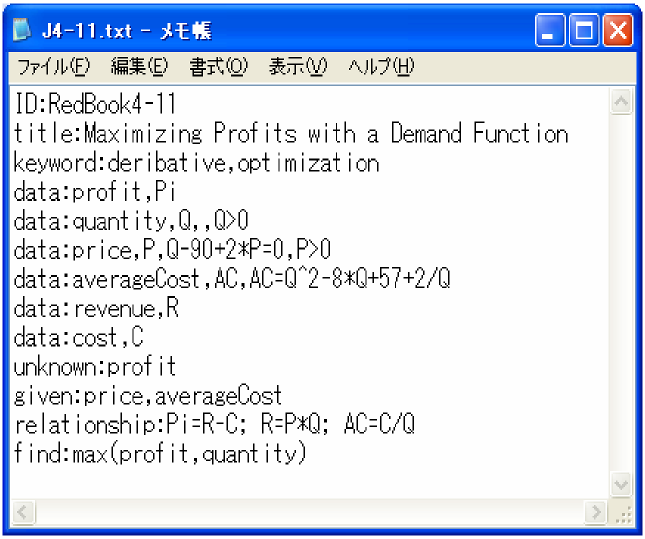


Fig. 2. The system flow of the e-Math Interaction Agent



```
ID:RedBook4-11
title:Maximizing Profits with a Demand Function
keyword:deribative,optimization
data:profit,Pi
data:quantity,Q,,Q>0
data:price,P,Q-90+2*P=0,P>0
data:averageCost,AC,AC=Q^2-8*Q+57+2/Q
data:revenue,R
data:cost,C
unknown:profit
given:price,averageCost
relationship:Pi=R-C; R=P*Q; AC=C/Q
find:max(profit,quantity)
```

Fig. 3. A meta-level description file that a teacher writes and inputs to the Interaction Agent

These four sub-modules are invoked by the main process “e-Math Interaction Agent” when it requires them.

The inference engine (a Prolog interpreter) is invoked to find a solution plan. The inference engine is also used to find the relationship between unknown and given data. As a rule, the individual relationships between data are stored in the knowledge base. For example, economical relationships among total cost, average cost and marginal cost are defined and stored. Because our primary target domain is mathematical economics, the main Prolog rules are in the form of (1) economical formulas and relationships and (2) mathematical formulas and knowledge.

Next, let us explain the mathematical software module. Solving a mathematical problem dynamically requires a mathematical symbolic computation function. For example, when the system solves a quadratic equation, expands a power function and calculates derivative and integral problems, the mathematical software is dynamically invoked. Because it was difficult for us to develop a symbolic computation module in Prolog from scratch, we decided to employ the widely used mathematical software, Maple[45]. The step-by-step calculations that are resultant data from Maple are also available if the student pressings the "More Detail" Button. This software's graph drawing function is also helpful. The graph image helps many students understand the problem (See Figure 7).

Now let us consider the next module, "Equation Server." Until recently, it was difficult to display mathematical expressions on Web browsers. Now it is finally possible to view MathML equations on most. In addition, a wide variety of software tools are available for authoring MathML expressions and for converting mathematical content in other formats into MathML ones.

In the e-Math Interaction Agent system, both MathML-based descriptions and equation images, such as those in jpeg and gif formats, are used to display the equations. The Equation Server generates the mathematical equation files to be embedded in the XML files and presented. For example, the XML file on the screen in Figure 4 includes four mathematical equation files. As the Equation Server, we use the WebEQ system, which is a Java-based suite of programs for authoring and displaying mathematics on the Web[46].

Next, we explain our developed module "Web Page Generator." In our implementation of the Interaction Agent of the e-Math system, our required solution plan is described as Perl scripts and stored in a knowledge base. An inference engine, which is a Prolog interpreter, finds these stored solution plans. The inference engine infers the required relationships among the economical data, the combination of mathematical formulas, and the combination of solution plan functions. These are finally embedded in the learning materials using Semantic Web technologies, such as XML and XSLT. The metadata schema of the learning materials is defined in advance, as shown in Table 1.

The browser screen consists of three frame areas: a blackboard area, an interaction area, and a student input area. The created metadata properties are embedded in a XML file, surrounded by the corresponding tags.

The XML files do not yet exist when the Interaction Agent begins guiding or interacting with a student. Rather, they are dynamically created by the Web Page Generator as required. When the next Web page needs to be displayed, the XML file corresponding to it is automatically generated. Figure 8 is an example of a XML file which was generated by the Web Page Generator. The XML file corresponds to the screen shown in Figure 7. As shown here, many math expression jpg files are embedded in the XML file.

Table 1. Tags defined for learning material XML files

Blackboard Area XML Tags:

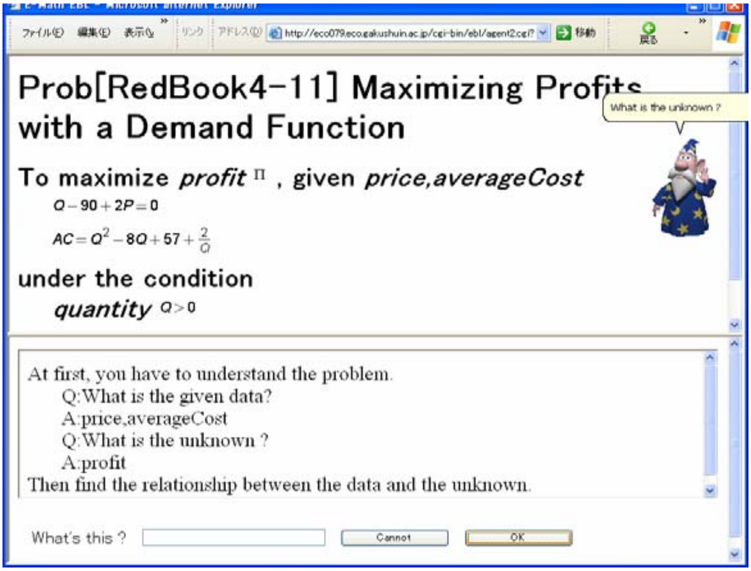
```
<xdoc>
<blackboard>
<statement> page title </statement>
<relationship> math term </relationship>
<simplify>
  <answer><url> URL of Maple programs </url></answer>
  <equation> equation </equation>
</simplify >
</blackboard>
</xdoc>
```

Interaction Area XML Tags:

```
<xdoc>
<interact>
<msg>
   <guid> guidance statement </guid>
   <Q> question statement </Q>
   <A> answer statement </A>
   <E> answer single math equation </E>
   <EE> answer multiple math equations </EE>
   <endmsg> the final statement </endmsg>
</msg>
<what></what>
<txtbox></txtbox>
<next_page> URL of the next page </next_page>
</interact>
</xdoc>
```

After this, the inference engine tries to devise a solution plan to calculate the answer sought. The inference engine finally creates the solution plan by unification. Based on this plan, the Web Page Generator generates a XML file step by step. The solution plan consists of several steps, with one Perl procedure per step. The invocation of the Perl procedure generates one XML file corresponding to each solution step. This XML file generation is dynamically executed, so that the content can be changed depending on the student situation and input. The generated XML file is displayed through XSLT stylesheets. These stylesheets are defined in advance based on the XML tags and stored in the XSLT stylesheet database. Many XML files are linked to each XSLT stylesheet.

Let us imagine a human teacher trying to help a student understand the target problem. The teacher repeatedly asks the same thing in many different ways: “What do you want to find? What is required? What are you supposed to seek?” These questions encourage the student to think. In the e-Math system, instead of a human teacher, a virtual



Prob[RedBook4-11] Maximizing Profits with a Demand Function

To maximize *profit* Π , given *price,averageCost*

$Q-90+2P=0$

$AC=Q^2-8Q+57+\frac{2}{Q}$

under the condition

quantity $Q>0$

At first, you have to understand the problem.

Q:What is the given data?

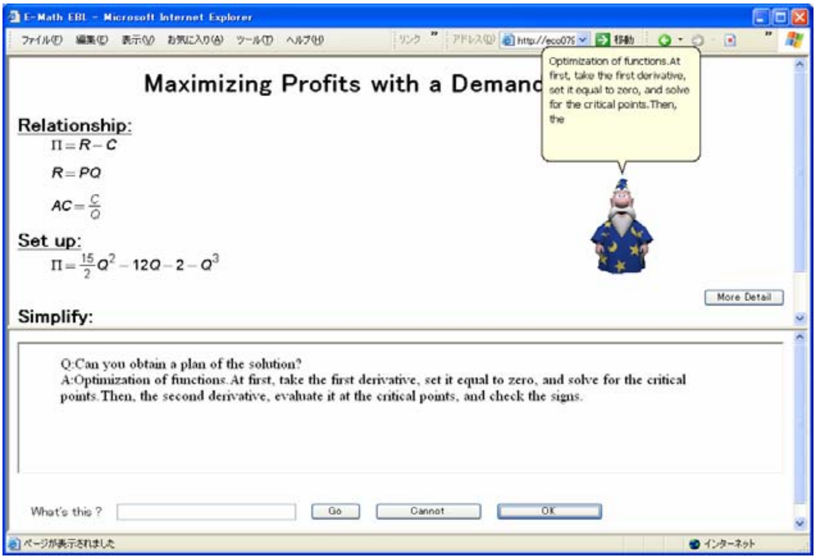
A:price,averageCost

Q:What is the unknown ?

A:profit

Then find the relationship between the data and the unknown.

Fig. 4. The generated learning materials that explain the problem of maximizing profits with a demand function and an average cost function



Maximizing Profits with a Demand

Relationship:

$\Pi=R-C$

$R=PQ$

$AC=\frac{C}{Q}$

Set up:

$\Pi=\frac{15}{2}Q^2-12Q-2-Q^3$

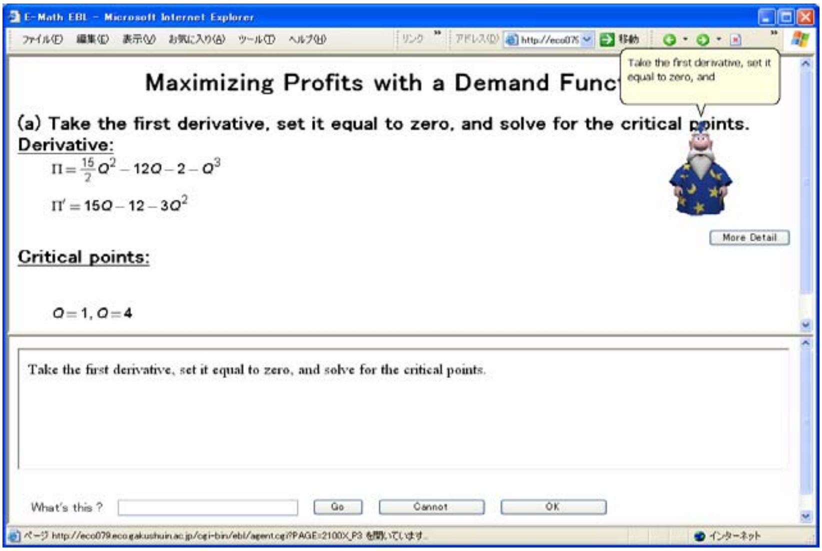
Simplify:

Q:Can you obtain a plan of the solution?

A:Optimization of functions. At first, take the first derivative, set it equal to zero, and solve for the critical points. Then, the second derivative, evaluate it at the critical points, and check the signs.

E-Math EBL - Microsoft Internet Explorer
Optimization of functions. At first, take the first derivative, set it equal to zero, and solve for the critical points. Then, the
More Detail
What's this ?
Go
Cannot
OK

Fig. 5. The generated learning materials in which the Pi (profit) equation is written in terms of only one independent variable, Q (quantity)


E-Math EBL - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
リンク
アドレス(D) http://eco079
移動
Take the first derivative, set it equal to zero, and
More Detail
Go
Cannot
OK
ページ http://eco079.eco.gakushuin.ac.jp/cgi-bin/ebl/agent.cgi?PAGE=2100X_P3 を開いています...
インターネット

Maximizing Profits with a Demand Func

(a) Take the first derivative, set it equal to zero, and solve for the critical points.

Derivative:

$$\Pi = \frac{15}{2}Q^2 - 12Q - 2 - Q^3$$

$$\Pi' = 15Q - 12 - 3Q^2$$

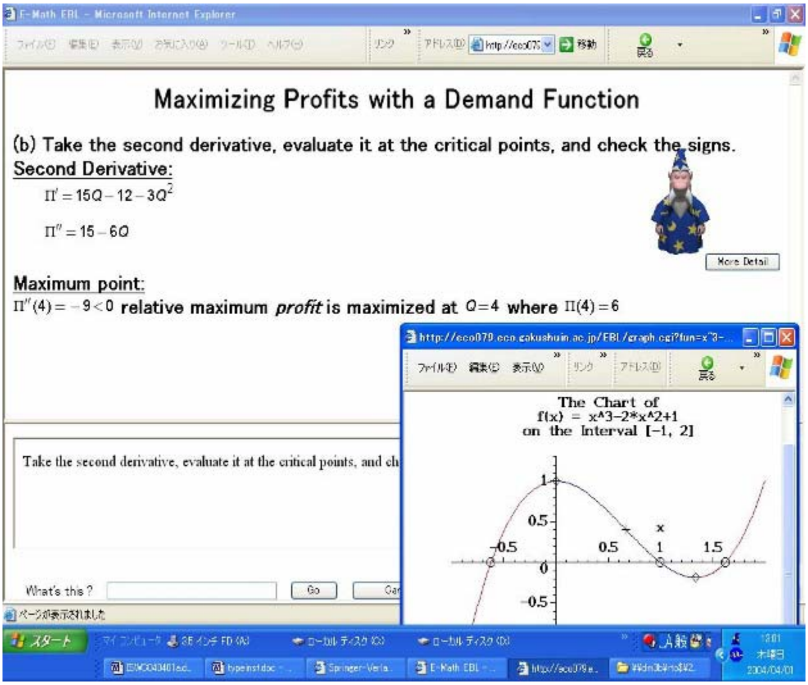
Critical points:

$Q=1, Q=4$

Take the first derivative, set it equal to zero, and solve for the critical points.

What's this ?

Fig. 6. The generated learning materials in which the first derivative is calculated, set equal to zero, and solved


E-Math EBL - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
リンク
アドレス(D) http://eco079
移動
戻る
More Detail
Go
Car
ページが表示されました
http://eco079.eco.gakushuin.ac.jp/EBL/graph.cgi?fun=x^3-...
ファイル(F) 編集(E) 表示(V)
1
0.5
-0.5
0
0.5
x
1
1.5
-0.5
スタート
12:01
木曜日
2004/04/01

Maximizing Profits with a Demand Function

(b) Take the second derivative, evaluate it at the critical points, and check the signs.

Second Derivative:

$$\Pi' = 15Q - 12 - 3Q^2$$

$$\Pi'' = 15 - 6Q$$

Maximum point:

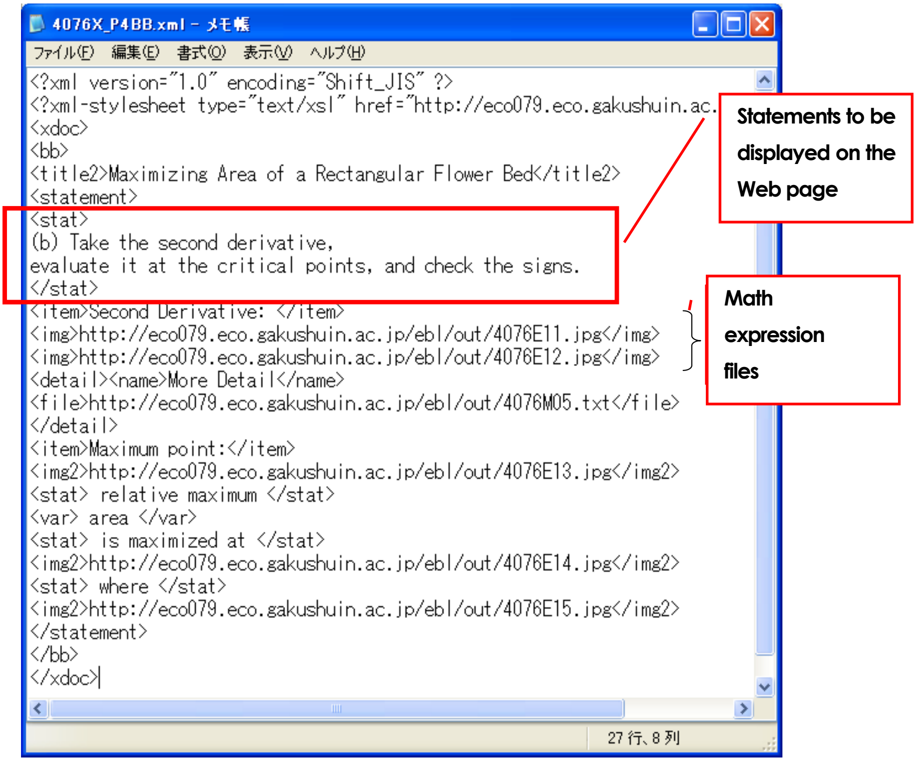
$\Pi''(4) = -9 < 0$ relative maximum *profit* is maximized at $Q=4$ where $\Pi(4)=6$

Take the second derivative, evaluate it at the critical points, and ch

What's this ?

The Chart of
f(x) = x^3-2*x^2+1
on the Interval [-1, 2]

Fig. 7. The generated learning materials in which the answer is shown after the student tests to determine whether the critical points are maxima or minima


4076X_P4BB.xml - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)
Statements to be displayed on the Web page
Math expression files
27 行、8 列

```
<?xml version="1.0" encoding="Shift_JIS" ?>
<?xml-stylesheet type="text/xsl" href="http://eco079.eco.gakushuin.ac.
<xdoc>
<bb>
<title2>Maximizing Area of a Rectangular Flower Bed</title2>
<statement>
<stat>
(b) Take the second derivative,
evaluate it at the critical points, and check the signs.
</stat>
<item>Second Derivative: </item>
<img>http://eco079.eco.gakushuin.ac.jp/ebl/out/4076E11.jpg</img>
<img>http://eco079.eco.gakushuin.ac.jp/ebl/out/4076E12.jpg</img>
<detail><name>More Detail</name>
<file>http://eco079.eco.gakushuin.ac.jp/ebl/out/4076M05.txt</file>
</detail>
<item>Maximum point:</item>
<img2>http://eco079.eco.gakushuin.ac.jp/ebl/out/4076E13.jpg</img2>
<stat> relative maximum </stat>
<var> area </var>
<stat> is maximized at </stat>
<img2>http://eco079.eco.gakushuin.ac.jp/ebl/out/4076E14.jpg</img2>
<stat> where </stat>
<img2>http://eco079.eco.gakushuin.ac.jp/ebl/out/4076E15.jpg</img2>
</statement>
</bb>
</xdoc>
```

Fig. 8. A sample XML file generated by the Web Page Generator

teacher appears and asks the questions. The virtual teacher can be selected from among the virtual characters available in Web browsers. For example, Microsoft Agent[47] and TVML characters[48,49] are available. TVML (TV program Marking Language), proposed by NHK (Japan Broadcasting Corporation) Science and Technical Research Laboratories, is a scripting language for producing entire TV programs. The TVML script is translated into computer graphics animation with synthesized speech, virtual camera movement, and real video.

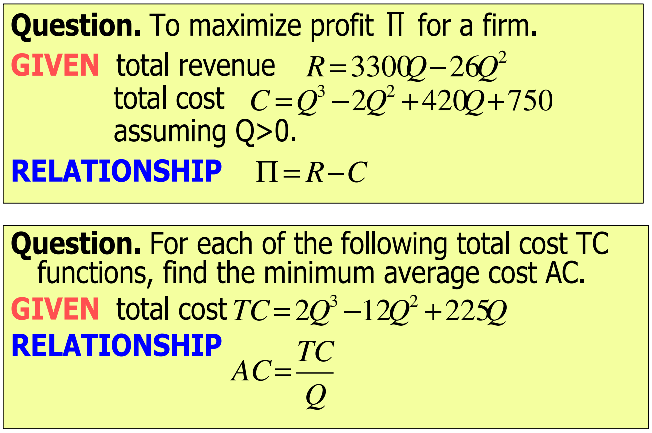
In the current version of the e-Math Interaction Agent, we use a Microsoft Agent character (See Figure 3). The questions or dialogues are prepared in advance and stored in the database by a system administrator. The Interaction Agent selects the appropriate narrations from the database and has the virtual teacher say them. The automation of the narrations is one of our future tasks. The mechanism to make the virtual teacher speak the selected dialogue is written in VBScript. The VBScript[50] program is embedded in the corresponding XSLT stylesheet. If the teacher wants to replace the virtual teacher with a different character, such as a TVML character, this can be done by writing the corresponding XSLT stylesheet and selecting it.

5 Prototype for Optimization Problems

Here we describe a prototype system of the Interaction Agent. Suppose that the target type is an optimization problem of single variable functions. Figure 9 shows schemas of two typical economical optimization math problems. Let us consider the solution plans for the optimization problems. As the system supervisor, we have defined two solution plans for the optimization problems: They are max() and min(). These solution plans share the same scheme, which includes the following steps:

(1) Determine the quantity to be maximized or minimized and write the equation for it.
(2) Use the constraints of the problem to write the equation in terms of only one independent variable, and simplify the equation.
(3) Find the first derivative, set it equal to zero, and solve the equation.
(4) Find the second derivative and test to determine whether the critical points are maxima or minima.
(5) Check for inflection points.
(6) Answer the question posed in the problem.

As the system supervisor, we have devised these steps to fit onto four Web pages. Thus, when the meta-level description file is input into the Interaction Agent system, the Interaction Agent automatically generates XML files corresponding to Figures 4 to 7. Figure 4 corresponds to step (1) and Figure 5 corresponds to step (2). Step (3) is executed as shown in Figure 6 and steps (4), (5), and (6) are executed as shown in Figure 7. For each step, the solution plan function is designed to define the learning material corresponding to the learning step. The solution plan function is written in Perl. The



Question. To maximize profit Π for a firm.

GIVEN total revenue $R = 3300Q - 26Q^2$
total cost $C = Q^3 - 2Q^2 + 420Q + 750$
assuming Q>0.

RELATIONSHIP $\Pi = R - C$

Question. For each of the following total cost TC functions, find the minimum average cost AC.

GIVEN total cost $TC = 2Q^3 - 12Q^2 + 225Q$

RELATIONSHIP $AC = \frac{TC}{Q}$

Fig. 9. Two typical economical optimization problems

four solution plan functions are defined in advance and stored in the "Solution Plan DB" (See Figure 2).

Our defined attributes in a meta-level description file are as follows:

(1) Data (name, symbol, given expression, condition).
(2) Unknown.
(3) Given.
(4) Relationship.
(5) Find.

As these data schemas are available to define all mathematical problems except proof problems, our proposed methods to automate mathematical learning materials have high descriptive power.

Next let us explain the meta-level description file shown in Figure 3. The attribute "data" consists of four items:

(1a) The name used in a word problem.
(1b) The symbol used in mathematical expressions.
(1c) The given expression, such as "Q-90+2*P=0."
(1d) The condition, such as "Q>0."

The attributes "unknown" and "given" refer to the unknown and given data in the problem. The defined data includes the unknown data, the given data, and the other variables used in the solution. The attribute "relationship" refers to mathematical expressions within the defined data. For example, let us consider an optimization problem that identifies the maximum profit of a firm (See Figure 4). In the meta-level description file shown in Figure 3, there are three economical relationships:

(1) A profit function, Pi=R - C.
(2) A revenue function, R=P*C.
(3) An average cost function, AC=C/Q.

These relationships are common economical knowledge. Sometimes, however, students of economical mathematics lack this kind of common knowledge. Therefore, the educational system has to help them use knowledge bases. In the example, the given demand function "Q-90+2*P=0" is first transformed to P=P(Q). Then, the revenue value "R=P*Q" is calculated by Maple.

The attribute "find" refers to the solution plan of the problem. For an optimization problem, two kinds of solution plans are defined: max() and min(). In the problem, the unknown data is expressed as an equation in terms of only one independent variable, and solved. These calculations are executed by the system, instead of the teacher, as shown in Figure 5. The following expression is created by the Interaction Agent using the mathematical software Maple:

$$\pi = \frac{15}{2}Q^2 - 12Q - 2 - Q^3$$

In interacting with students, a good teacher lets the students guess the answer before revealing it. Based on this principle, some parts of the learning materials are not displayed when the next Web page appears. Only when a student presses the "More Detail" button are the hidden materials displayed in another window. For example, the calculation process shown is displayed after the "More Detail" button is pressed. This shows how to calculate the first derivative step by step, applying some mathematical differentiation formulas, such as a product rule or a chain rule. Even if a student is stuck, the student will understand the differentiation process by observing these steps. The XML tag that is used to make the "More Detail" button is represented as "<detail>". Various buttons have been developed to help stuck students. When a student presses the "Graph" button, a graph of the equation, including critical point markings, is displayed (See Figure 7). These graphs are useful in helping students to understand the solutions and the scheme of the problem.

We have collected more than 50 optimization problems from widely-used mathematical textbooks[51,52,53,54]: Describe the meta-level description files for these problems, and input these files to our Interaction Agent system, the system has successfully automated the courseware to teach these problems.

6 Conclusions

In this paper, we have described the e-Math Interaction Agent that dynamically automates on-line Web-based materials. For such automation, Semantic Web techniques are effective. In our implementation of the e-Math Interaction Agent, we used XML and XSLT to automate learning objects. In general, the advantages of automatic generation by XML and XSLT are their low cost and speedy turnaround time. Meanwhile, their disadvantage is that they produce Web pages of uniform appearance whose content, layout and generated dialogues can thus be tedious for the viewer. The learning materials generated by our system can explain economical and mathematical relationships appeared in the given word problems in various ways and from various angles: (a) in words, and using visual materials, such as graphs, (b) as an economical relationship, and as a mathematical relationship, and (c) using mathematical symbolic calculations and concrete value calculations. In the generated courseware, the virtual teacher explains the relationships repeatedly and in these various ways. The student can also see the relationships interactively and visually from various angles through 3D graphs.

Acknowledgements

This research is supported in part by the Japanese Ministry of Education, Science, Sports, and Culture under Grant-in-Aid for Scientific Research (C) (2)15606014.

References

1. Michael C. Daconta, Leo J. Obrst, and Kevin T. Smith: The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management, John Wiley & Sons Inc., 2003.
2. Vladimir Geroimenko: Dictionary of XML Technologies and the Semantic Web, Springer, 2003.
3. John Davies, Dieter Fensel, and Frank Van Harmelen: Towards the Semantic Web: Ontology-Driven Knowledge Management, John Wiley & Sons Inc., 2003.
4. XML: http://www.w3c.org/TR/xmlschema-2/.
5. Dan Brickley and R.V. Guha: RDF Vocabulary Description Language 1.0: RDF Schema, 2002-04-30, W3C Working Draft, http://www.w3.org/TR/rdf-schema.
6. Resource Description Framework: http://www.w3c.org/RDF/.
7. W3C: XSL Transformations (XSLT) Version 1.0, W3C Recommendation, 16 November 1999, http://www.w3.org/TR/xslt.
8. ActiveMath: http://www.activemath.org/.
9. WME Efforts and Related Systems: http://icm.mcs.kent.edu/research/wme.html.
10. Yukari Shirota: "Knowledge-Based Automation of Web-Based Learning Materials Using Semantic Web Technologies," Proc. of The Second International Conference on Creating, Connecting and Collaborating through Computing (C5), Kyoto, Japan, January 29-30, 2004, pp.26-33.
11. Yukari Shirota: " A Semantic Explanation and Symbolic Computation Approach for Designing Mathematical Courseware," Proc. of The Third International Conference on Creating, Connecting and Collaborating through Computing (C5), Kyoto, Japan, January 28-29, 2005, (to appear).
12. W3C Math Home: http://www.w3.org/Math/.
13. IBM: "Hypermedia Browser Techexplorer", www-3.ibm.com/ software/network/techexplorer/.
14. Michael Juntao Yuan: "Building dynamic Web sites with mathematical content", http://www-106.ibm.com/developerworks/java/library/j-jspmath/?loc=j.
15. Design Science Inc.: MathType and WebEQ, http://www.dessci.com/en/products/mathtype/.
16. Stéphane Dalmas, Marc Gaëtano, Stephen M.Watt:"AnOpenMath 1.0 Implementation", Proc. of ISSAC 1997, pp. 241-248.
17. Andreas Strotmann: "The Categorial Type of OpenMath Objects", Proc. of MKM 2004, pp. 378-392.
18. Michael Kohlhase, "OMDoc: Towards an Internet Standard for the Administration, Distribution and Teaching of mathematical Knowledge", Proceedings of Artificial Intelligence and Symbolic Computation, Springer LNAI, 2000.xml.coverpages.org/omdoc.html
19. The OpenMath Society: The OpenMath website at http://www.openmath.org/cocoon/openmath/overview/index.html.
20. The OpenMath Society:"The OpenMath Standard," 2004 at http://www.openmath.org/cocoon/openmath//standard/ om20/index.html.
21. Worldwide Web Consortium: "W3C Math Home" at http://www.w3.org/Math/.
22. OMDoc: http://www.mathweb.org/omdoc/.
23. Michael Kohlhase: "OMDoc: An Open Markup Format for Mathematical Documents (Version1.1), June 5, 2003 at http://www.mathweb.org/omdoc/.
24. Michael Kohlhase: "OMDoc: an infrastructure for OpenMath content dictionary information," ACM SIGSAM Bulletin, Vol. 34, No. 2, June 2000, pp. 43-48.

25. MapleSoft: Maple, http://www.maplesoft.com/.
26. Wolfram Research: Mahtematica, http://www.wolfram.com/products/webmathematica/index.html.
27. James E. White: Mathwrite, http://www.mathwright.com/hr_default.html.
28. Paul S. Wang, Norbert Kajler, Yi Zhou, Xiao Zou: "WME: towards a web for mathematics education", Proc. of ISSAC 2003, pp. 258-265.
29. The ActiveMath group: Erica Melis, Jochen Büdenbender, George Goguadze, Paul Libbrecht and Carsten Ullrich: "Knowledge Representation and Management in ActiveMath," Annals of Mathematics and Artificial Intelligence , Vol.38, No.1—3, 2003, pp.47-64.
30. Paul S. Wang, Yi Zhou, and Xiao Zou: "Web-based Mathematics Education: MeML Design and Implementation", Proc. of ITCC (1) 2004, pp. 169-175.
31. Norman Walsh, et al.: "DocBook XML 4.2," at http://www.oasis-open.org/docbook/xml/4.2/index.shtml.
32. H. Cuypers and H. Sterk, "Mathbook, web-technology for mathematical documents", Electronic Proceedings of the BITE conference 2001, Eindhoven, Nederland. http://www.riaca.win.tue.nl.
33. ADL (Advanced Distributed Learning): Sharable Content Object Reference Model at http://www.adlnet.org/.
34. ADL (Advanced Distributed Learning): "SCORM ContetnAggregation Model Version 1.3", 2004.
35. Erica Melis, Jörg H. Siekmann: "Knowledge-Based Proof Planning," Artififitial Intelligence, Vol. 115, No. 1, 1999, pp. 65-105.
36. MuPAD home page: http://hpc.cs.ehime-u.ac.jp/MuPAD/.
37. G. Pólya: Induction and Analogy in Mathematics (Mathematics and Plausible Reasoning, Volume 1), Princeton University Press, 1968.
38. G. Pólya: Patterns of Plausible Inference (Mathematics and Plausible Reasoning, Volume 2), Princeton University Press, 1968.
39. G. Pólya: How to Solve It (Second edition), Penguin Books, 1957.
40. G. Pólya: MATHEMATICAL DISCOVERY on Understanding, Learning, and Teaching Problem Solving Volume 1 and 2, John Wiley & Sons, N.Y., 1962.
41. H. Schoenfeld: "Learning to Think Mathematically: Problem Solving, Metacognition, and Sense Making in Mathematics," chapter 15. McMillan Publ.Company, New York, 1992.
42. Paul A. Cairns, Jeremy Gow: "On Dynamically Presenting a Topology course," Ann. Math. Artif. Intell. Vol. 38, No. 1-3, 2003, pp. 91-104.
43. E. Melis and C. Ullrich: "How to teach it - Polya-scenarios in ActiveMath," In U. Hoppe, F. Verdejo, and J. Kay, editors, Artificial Intelligence in Education, IOS Press, 2003, pp. 141-147.
44. Georgi Goguadze, Erica Melis, Carsten Ullrich, Paul A. Cairns: "Problems and Solutions for Markup for Mathematical Examples and Exercises," MKM 2003, pp. 80-92.
45. Maplesoft: Maple, http://www.maplesoft.com/.
46. Design Science: WebEQ, http://www.dessci.com/en/products/webeq/.
47. Microsoft Corporation: Microsoft Agent, http://www.microsoft.com/msagent/default.htm.
48. M. Hayashi: Image Compositing Based on Virtual Cameras, IEEE MultiMedia, Vol. 5, No. 1, pp. 36-48, 1998.
49. NHK: TVML, http://www.strl.nhk.or.jp/TVML/.
50. Microsoft Corporation: Microsoft Agent, http://www.microsoft.com/msagent/default.htm.
51. Edward, T. Dowling: Theory and Problems of Introduction to Mathematical Economics ('Third Edition'), McGraw-Hill, 1980.

52. D. Downing: CALCULUS, The Easy Way (‘Third Edition’), Barron's Educational Series Inc., 1996.
53. D. Ebner: MATH WORD PROBLEMS, The Easy Way, Barron's Educational Series Inc., 2002.
54. B. L. Bleau: Forgotten Calculus (‘Third Edition’), Barron's Educational Series Inc., 2002.

Towards Open Education Through Distributed and Networked Information Systems - An Experience-Based Approach

Tosiyasu L. Kunii

IT Institute, Kanazawa Institute of Technology,
1-15-13 Jingumae, Shibuya-ku, Tokyo 150-0001, Japan
Phone: +81-3-5410-5280, Fax: +81-3-5410-3057
tosi@kunii.info
http://www.kunii.com/

Abstract. To overcome overwhelming and global international struggles to secure limited resources such as oil and land, the potential role of open source education through networked and distributed information systems (DNIS) on the Web to create advanced IT experts as unlimited global resources is increasing rapidly. An experience-based summary of global open education is presented solely for promoting its practices. My life has been benefited from practicing open education, first at an elementary school and later at a graduate school. The openness has been local because of the lack of globalization mechanisms in education. It is fairly recent that we have effective global educations mechanisms for global interactivity and global two way communications such as the web and cyberspaces, distributed and networked information systems (DNIS) in particular. Compared to local open education, global open education removes the boundaries of ages, organizations, nations, sexes, and disciplines. Many unseen barriers exist to prevent global open education, mostly originating from survival intuitions and fights embodied in life itself. Since the barriers are rooted in the nature of life, it is hard to practice global openness in education. Hence it is important to cooperate for us to practice it to see real advances in our knowledge.

1 Prologue

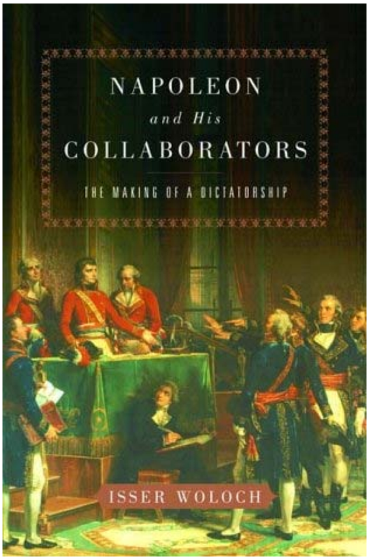
Living in the ear of global fighting for limited resources such oil and land, it is crucial to realize that there is a way to create unlimited and more valuable resource of advanced IT experts internationally based on open source education through distributed and network information systems (DNIS) on the Web. Open source has finally passed the level of critical mass to serve for such purpose as educational sources. Although no truly usable popular educational courseware has to come, knowing the real potential is the first step towards the practices of the courseware development and advanced expert level internal IT education.

As the first step, I believe it is important to make my life long experiences on open education presented in 2001 as dali2001 at the University of Aizu [10] publicly available through this paper.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 204 – 212, 2005.

2 Public Education: The Dawn of Global Education as the Foundation of a National Business Model

Historically it has been distinct that *public education* has served as the fundamental mechanism of the power shift to a public nation from an aristocratic country. Many republics and even public government-based empires have been created by this mechanism to recreate ever expanding bureaucrats and private enterprise leaders to govern such nations[1*, 2].



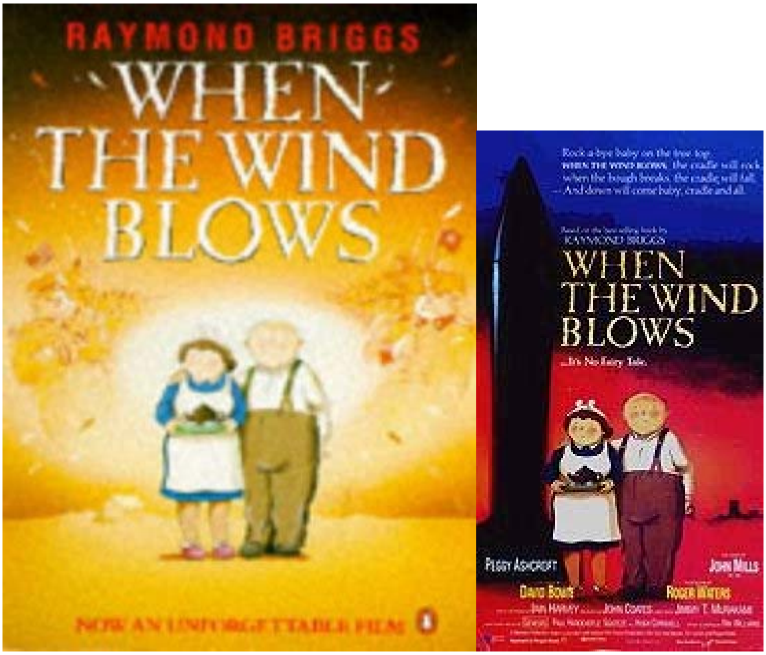
----

*[1] One typical case is that of Napoleon Bonaparte. An encyclopedic description states what he had as a national business model to use public education as a key mechanism: "In 1808, when Napoleon reorganized the French educational system under the jurisdiction of the University of France, the University of Paris was reopened. Faculties of literature, law, medicine, and science, together with a later-abolished faculty of theology, were established at the Sorbonne, which had been designated the seat of the academy of Paris (one of the 17 educational districts into which France was now divided) and the seat of the University of Paris itself. A library was established at the Sorbonne in 1808; its collection today numbers more than 3 million volumes. Under terms of the Orientation Act of 1968, which reformed French higher education, the university was reconstituted as 13 autonomous teaching and research faculties. These were founded during 1968-71."

As the contents of public education, the main successful line on top of literacy has had the skeleton:

Science -> engineering -> commerce -> finance

for accelerated formation of national power. Needless to say, there have been a military educational line. Ironically, too powerful weapons to destroy the entire


RAYMOND BRIGGS
WHEN THE WIND BLOWS
NOW AN UNFORGETTABLE FILM
Rock a-bye baby on the tree top
WHEN THE WIND BLOWS the cradle will rock
when the bough breaks the cradle will fall
And down will come baby, cradle and all.
Based on the best selling book by
RAYMOND BRIGGS
WHEN THE WIND BLOWS
...It's No Fairy Tale.
PEGGY ASHCROFT
JOHN MILLS
DAVID BOWIE
ROGER WATERS
JOHN COATES
JIMMY T. MURAKAMI

meteorology of the globe and then the entire life on the earth [3,4], has made this line basically ineffective and obsolete, although still practiced.

2 Research Universities as a Spirally Growing National Business Model

The notion of the *research universities* was coined as a national business model by 16 American universities at the end of 1800s under the slogan of "*to advance knowledge*". After founding the Association of American Universities (AAU) in 1900, the research university model based on *publishing* original scientific journals built on a *peer and open review* system of professional society's created by AAU as the core, had made America the world leading country in research, industry, commerce and financing in 30 years [6]. Indeed, the term "*publish or perish*" has been characterizing the nature of American science. Further, since it is *open*, it has been *spirally growing* as we see today. Thus the research university model is shown to be a firm national business model to spirally grow the nation run by the model without limitation, advancing knowledge through discoveries and inventions. Its limitation was clearly observed when the nuclear winter simulation was conducted by the U. S. A. and Soviet Union team chaired by Carl Sagan [3]. This is a clear message conveying that mere advancing knowledge may end up with destroying the entire life on the earth.

3 Meiji Restoration as a Fast Catch Up National Business Model

Thirty years earlier than AAU was founded and after three hundred years of closing the country to the world, Japan has opened up the country in 1868 to catch up the world progress as fast as possible. It is generally called the Meiji Restoration. The Meiji Restoration model is basically a public education model. The emphasis is on enlightenment of people using imported knowledge combined with a re-engineering model to analyze existing advanced systems and to crack them down to reutilizable resources. As the core of its higher education system, imperial universities were established, not as research universities defined above, but as enlightenment universities to illustrate already found knowledge outside the country. As a natural consequence, higher education in Japan has produced the bottom level number of Nobel laureates. On the other hand, it has successfully produced bureaucrats, business leaders, politicians, military power, and professors as enlighteners. Since the model is for fast catch up, after the maturity of the country reaching the level of the world, the leaders of Japan are lacking the abilities to clearly see the future and to advance knowledge. Their common everyday saying "*unclear future*" to describe the future of Japan as reported in news is a definite evidence of the nature of their business model established at the Meiji Restoration as a national business model for non original enlightenment and catch up.

4 Cyber Education as a Global National and International Business Model

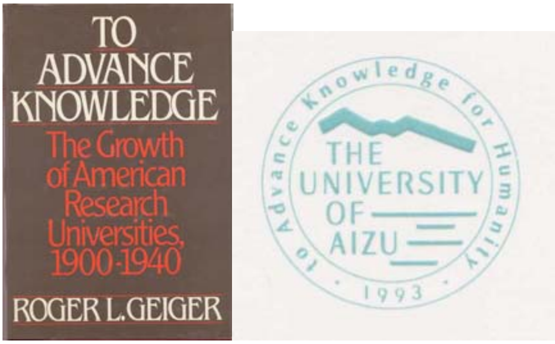
The University of Aizu is modeled to enhance the research university concept eliminating inhumane aspects by explicitly adding "for Humanity" to the logo.

Let me first refer a page at the beginning of the book "Cyberworlds" [5] based on a Japan-France Workshop on Synthetic Worlds held at the University of Aizu in mid 1990s:

"Think. Do we live our life to end up sucked into black holes? Is that the destiny of the human race? Numerous people die on this earth from hunger, from disease, or even through futile hostility. If this is one side of human reality, the other side is presented here. Can we not synthesize better worlds and then make them real? Living in the digital era, a bit in a computer can be transformed into a step movement through devices such as stepping motors, linear cars, and direct control robots. Now we can be the creators of synthetic worlds. Let us cooperate toward a common goal. Cyberworlds is the manifesto and the records of the pioneers in this field. Yes, it is a book of wisdom and an open invitation to synthetic worlds, still very primitive and humble. Further, we should not let fear of failure stand in our way; to err is human.

Tosiyasu Laurence Kunii and Annie Luciani"

I drafted it to signify the meaning of the logo of the University of Aizu, stating "To Advance Knowledge for Humanity". It is intended to deliver a firm message when I coined it in 1992: The globe consists of regions, and the University of Aizu in a serene setting between Mt. Bandai and Lake Inawashiro, aims at a global and humane higher education as a research university to practice the web-based global research and education through it to ultimately practice a global and open national and international business model. For the purpose of fostering understanding of the true meaning of research universities, I also distributed copies of book "To Advance Knowledge" by Roger L. Geiger published by Oxford University Press in 1986 [6] to professors joining the University of Aizu from fourteen different countries.


TO ADVANCE KNOWLEDGE
The Growth of American Research Universities, 1900-1940
ROGER L.GEIGER
to Advance Knowledge for Humanity
THE UNIVERSITY OF AIZU
1993

The open resources on the web have the potential for global research and education removing all boundaries. Mechanisms based on research university national business model are to be generalized to reach the level of an international business model. I have been pursuing the research university international business model of the University of Aizu in Tokyo Metropolis. Experimental practices at a one year professional IT master course named IT Professional Course (ITPC) opened in 2000 at Hosei University has produced successful results in bringing students into global and open business startups, for example to run a matchmaking office for attorneys-at-patent based on open software. Open software used are Linux to build client-server systems and PostgreSQL to implement database servers. For the undergraduate first year students at Faculty of Computer and Information Sciences of Hosei University opened in 2000, Linux kernels has served as an excellent material for practicing application migration to different processors on Linux kernels.

Such practices are directly targeted at realizing an open and global graduate- and undergraduate- research university curricula for IT professionals as the core of a global world business model to shift the world industry from computers to *web-based information appliances*.

It is intended to extend such cyber education to turn ever expanding population in developing countries into creative, innovative and humane IT professionals through solar battery operated two way broadband satellite communication systems. It is expected to brake the current disastrous prediction of population study authorities stating that the population explosion in developing countries will case the undersupply of even primitive literacy education instructors at public schools. It is clear that a simple public school model cannot save the population on the earth, and in contrast, the web-based global and open education model, in short a *cyber education model*, turns it into valuable world growth resource to advance knowledge for humanity. Humanity education for IT professionals requires extensive research. We know very little on it. How to and where to submit original works as well as how to perform peer reviews on the web pose other key research themes. Since they are the key elements of the web-based international business, we cannot go any further without finding answers to them. Yet, there is no shortcut. Web-base journals and web-based reviews are popular practices and at least we need to follow these lines. Currently popular varieties of web-based computer contests are easily extended to web-based peer review systems and is among the most promising approaches. It is worth noting that Russia and Eastern European countries are quit advanced on web-based publishing [7, 8].

Other closely related important issues include four themes:

1. Early education: How to make global education available to younger generations; education takes time.
2. Regionally rooted education: How to root global education in regions to construct cultural ecology to ferment creative cultural environments; the world obviously consists of regions and people there.
3. Culture education: How to educate cultural heritage and inherit cultural accumulation; the height and the depth of educational contents largely originate in the cultural heritage.
4. Æsthetic education: How to educate refined sense as the ultimate goal for quality education rather than quantity education.

These four themes are closely related to our vision into the future. It is clear without roots there is no life nor culture. They are also tied with ethics. Upright people serve as the backbone of our future.

5 Practicing Experiences of Global Openness Education at Elementary and Secondary Schools

On August 14, 1945 when I was a seven years old elementary school boy, there was a real refreshing and vitalizing event to get exposed to a sudden openness in education after so many years of wartime nationwide information freeze and control. Almost all facts had been closed by our government for long. Realizing the discipline on facts was *science*, I have decided to dedicate my life to science and choose university professorship to pursue and prevail science. Still science for me is a true refreshment, romance and culture. At the time I reached the fifth grade in my elementary school, I organized a student open research group getting fifty students to join and my funding proposal being approved by the school. We researched on the science of the material world, and we were quite convinced on the existence of a consistent truth to govern the material world when we encountered with the theory of elementary particles.

Just before my going to the secondary school, the head of the chemistry student research group, Mr. Hata, visited my home and appointed me his successor. At the secondary school, there was a movement of new open education. The whole afternoon classes were replaced by student open research activities, having teachers as advisors.

The halves of the school achievement scores were based on such open research activities. I mostly relied on university reference books to find out necessary details on practicing science. In Joseph McCarthy' advise to the U. S. President Truman and the resulting McCarran Act had forced Japan to stop open education, monitored by newly created assistant principals. Our student strike for three months was unsuccessful to move the closed education back to the open and refreshing education. We all suffered from mental choking. No more active student participation to education had created dark campuses all over Japan. It was like back to our wartime. All official school textbooks have been controlled and censored by our government and the major parts of wartime records of outside invasions were deleted. Thus, open education has been refreshing for people, but nightmares for dictators.

6 Practicing Experiences of Global Openness Education at Undergraduate and Graduate Schools

After boring high school years having students working almost entirely in university entrance exams, I was hoping to do open research at the University of Tokyo. There was neither open research nor open education. After watching what could be done to it, I have started an open study student group TSG (Theoretical Science Group) in 1959. It is still active and we celebrated the 50th anniversary in 1999. I saw many Vaio machines brought into the celebration party and leaned that Vaio machines had been developed by ex TSG members working in Sony as information home appliances rather than as computers. It indicate the potential of open education. Also, when Department of Information Science was created in 1975, many TSG members enrolled as students. Later, after my open systems education based Berkeley UNIX source code, a numbers of them has become systems experts and worked at Software Research Center of Ricoh Co., Ltd. under the direction of Dr. Hideko S. Kunii to build the entire networked educational and administration systems based on BSD UNIX in 1993 for the University of Aizu.

At a graduate school, I had pushed open research and with other graduate students, formed a research group that could pursue model-based scientific research to study bioengineering. Computing the models of nonexistent molecules presented me the thrilling fact of a creation of cyberworlds inside computing as the cyber genesis [9] in late 1960s. Researches on open worlds inside computers, cyberworlds, was the purpose of founding Information Science Laboratory at Faculty of Science of the University of Tokyo in 1970. It has been promoted to Department of Information Science in 1975. It was a natural extension to coin and found the University of Aizu with the goal to research on further open and global cyberworlds and extend open and global education through distributed and networked information systems (DNIS). Its facility has included one workstation for everybody on campus with 24-hour open campus year round. Open recruiting of faculty members has succeeded to gather the majority, actually close to 70%, from the world to make it promising as an ever growing international research university.

7 Epilogue

On the globe, the globalization has been progressing to increase the instability of economy and societies. If we properly use the major globalization technology such as networked and distributed information systems [DNIS] on the Web using abundant potential advanced IT educational materials of open sources to create unlimited valuable human resources in advanced IT, the current ever intensified international fights for limited resources such as oil and land will gradually lose their positions and meanings. The theory of evolution of life by Louis Lapicque as explained by Paul Chauchard [11] as the Lapicque diagram clearly proves the reason as I have stated in 2004 [12].

dali 2001 started by Carl Vilbrandt was truly a manifest of the ultimate goal of open and global education at the University of Aizu. "dali", Digital and Academic Liberty of Information, has conveyed it as well as the people joining the university. Regarding dali2001 and its successors, unfortunately, the promised proceedings has never been published and the demand to get my paper presented in 2001 [10] has been increasing. Practicing publishing is truly the first step to advance knowledge as explained in Chapter2.

In reflection, the contents of the experience have been increased the importance ever since by further troublesome international globalization. Further, open source governed by freedom to make source codes freely available as defined by Richard Stallman as GPL [13] has passed its critical mass to really serve as advanced IT expert educational materials if we properly develop courseware, although it itself is a heavy task requiring many international talented IT experts and big support. The work here is expected to be a first step towards it.

As I have stated in [12], as a matter of fact, on the web, the cyberworlds of GPL-based open sources have more potentials to adapt to the rapidly changing computing applications than closed proprietary software because of the GPL to borrow and utilize functions mutually and returning the results to open sources according to the GPL [13]. The potentials of GPL-based open source software in education to develop IT professionals in exploding population areas on the earth will save the human future in overcoming the critical shortage of IT professionals in developing fundamental software such as embedded OS and real time controllers which can never been successfully developed on top of the proprietary and closed OS. It is simply because, unlike developing application software running on the basic system software, developing basic system software itself requires in-detail knowledge of the core software and hardware functionality such as interrupt mechanisms, scheduling mechanisms, queue handling, device drivers, input/output interfacing, and storage structures. With proprietary software, the source code accesses are very tightly controlled by non-disclosure agreements with legal penalties, making it impossible to learn insides for practicing open education. Even such scientifically and legally clear facts have been very often ignored for short term profits of limited owners of proprietary software. In terms of *political economy*, it is an extremely hazardous situation both domestically and internationally, and in reality invading the human future for the sake of such limited relatively short term profits in human history.

On the other hand, we should not be confused such educational merits of open sources with the daily convenience and benefit of the use of supported proprietary software. Without support, we cannot use any software to achieve our daily jobs. The current unnecessary misunderstanding of the use and the education need to be cleared as soon as possible.

Acknowledgements

The author's thanks go to Professor Subhash Bhalla for their invitation to deliver this at DNIS 2005 as a keynote. The theme of the paper was also based on his advise considering the increasing importance of DNIS in global higher education in IT and computer sciences. The author is also grateful to Professor Carl Vilbrandt for initiating dali meetings at the University of Aizu in 2001 to promote open education.

References

[1] Paris, Universities of," Microsoft® Encarta® Online Encyclopedia 2001. http://encarta.msn.com.
[2] Issar Woloch, "Napoleon and his Collaborators: The Making of a Dictatorship", W.W. Norton & Company, (February 2001).
[3] R.P. Turco, O.B. Toon, T.P. Ackerman, J.B. Pollack, Carl Sagan, "Nuclear Winter: Global Consequences of Multiple Nuclear Explosions", *Science*, V. 222, No; 4630, December 23, 1983.
[4] Raymond Briggs, "When the Wind Blows", Penguin Books (1983).
[5] Tosiyasu L. Kunii and Annie Luciani (Editors), "Cyberworlds", Springer-Verlag, 1998.
[6] Roger L. Geiger, "To Advance Knowledge", Oxford University Press, 1986.
[7] Computer Graphics and Geometry, http://cg-g.newmail.ru/
[8] Spring Conference on Computer Graphics, http://www.isternet.sk/sccg/
[9] Tosiyasu L. Kunii, "Discovering Cyberworlds", IEEE Computer Graphics and Applications, pp. 64 - 65, January/February 2000.
[10] Tosiyasu L. Kunii, "Practicing Global Openness in Education: From Elementary Schools to Graduate Schools", dali2001(Digital and Academic Liberty of Information), March 26-29, 2001, Aizu-Wakamatsu, Japan.
[11] Paul Chauchard, "Précis de Biologie Humaine – Les Bases Organiques du Compotement et de la Pensée -", Presses Universitaires de France, 1957.
[12] Tosiyasu L. Kunii, "The Potentials of Cyberworlds –An Axiomatic Approach-", Proceedings of International Conference on Cyberworlds, 18-20 November 2004, pp. 2- 7, Tokyo, Japan, IEEE Computer Society Press, Los Alamitos, California, U. S. A.
[13] Richard M. Stallman, Lawrence Lessig (Introduction), Joshua Gay, "Free Software, Free Society: Selected Essays of Richard M. Stallman", Free Software Foundation; 2002.

Exploring Web Logs with Coordinated OLAP Dimension Hierarchies

Mark Sifer

School of Information Technologies, University of Sydney, NSW, 2006, Australia
sifer@it.usyd.edu.au

Abstract. Multi-dimensional data occurs in many domains while a wide variety of text based and visual interfaces for querying such data exists. But many of these interfaces are not applicable to OLAP, as they do not support the use of dimension hierarchies for selection and aggregation. We introduce an interface technique which supports visual querying of OLAP data. It is based on a data graph rather than a data cube representation of the data. Our interface presents each dimension hierarchy in a zoomable panel which supports selection and aggregation at multiple levels. Users explore data and query by making selections in several dimension views. Three view coordinations are identified; progressive, global and result only. We demonstrate our interface technique with an example web log dataset of site visits organised into time, downloads, visitor address and referrer address dimensions. This article provides an extended treatment of an earlier short paper [6].

1 Introduction

Concerns about data integrity and update performance have driven much database research, while user interfaces were often an add-on. Updates and queries are often done directly by applications, or via a standard language like SQL for ad-hoc queries, or via tools such as query by example which translate to SQL. However, with On-Line Analytic Processing (OLAP) systems [5] there has been a reversal of concerns. Typical OLAP data does not change, as it is usually historical data, while a major concern is supporting the ad-hoc exploration of the data by an analyst or other users looking for trends or patterns at varying levels of detail, perhaps integrated with decision support applications.

A data cube combined with dimension hierarchies is the standard model for OLAP data, which is often generated from relational data organised in a star schema. A data cube can be queried or restricted by slicing and dicing dimensions. It can be aggregated or deaggregated through roll-up and drill-down operations, while views of the cube can be altered via rotations. The standard interface for exploring data cubes is the cross-tab or pivot table which are multi-dimensional spreadsheets. They allow values within a query result set to be compared, but don't provide views which show query results in a larger context, which would aid data exploration.

This paper presents a new interface technique for querying OLAP data. The technique's underlying model is a data graph (defined in section two) rather than a data cube. Our interface uses coordinated views of each dimension hierarchy, which support

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 213–224, 2005.

both selection of dimension categories and aggregation of values through these categories. We present three view coordinations to satisfy different exploration roles: progressive, global and result. A progressive coordination supports drilling down into the data perhaps searching for an interesting trend, in a way that allows users to rapidly vary the query without getting lost. Once an interesting subset is found, users can see it in full detail with the result coordination. Then to see how the found subset is distributed across each dimension category the global coordination is used.

The demonstration dataset contains visits to a web site. Our LogConverter program converts a standard Apache web server text log into a Structured Graph Format (SGF) [2] XML document. Our interface design has been implemented in the Structured Graph Viewer (SGViewer) which loads an SGF file. SGViewer is a design exercise rather than a deployable tool, as it supports only 10K leaf nodes in five dimensions.

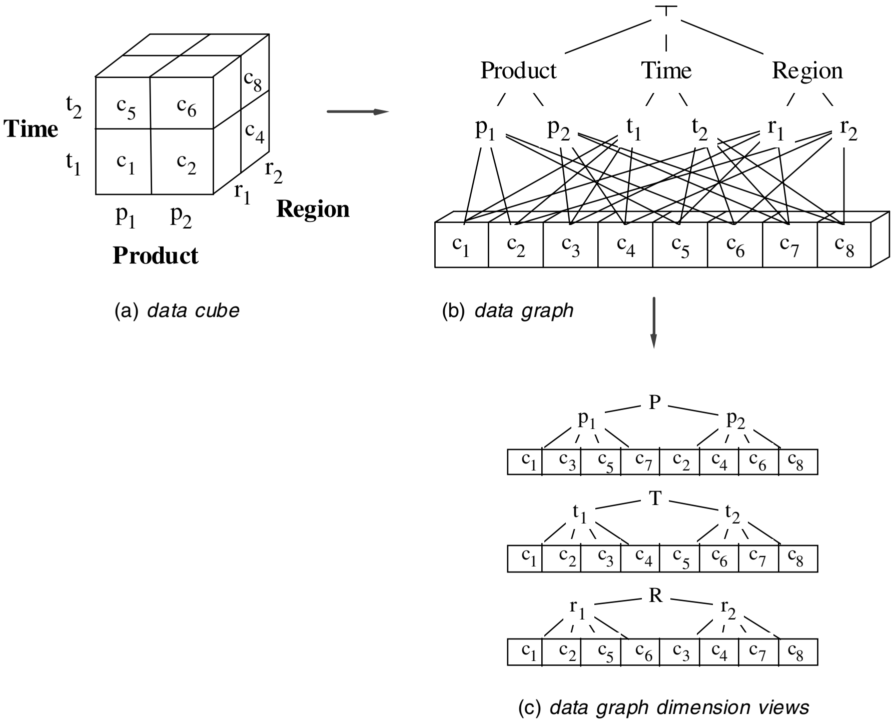
Section two presents our data model; a data graph. Section three presents the multiscale visualisation used in each dimension view. Section four presents an example use of our interface technique to explore web log data. Sections five and six cover our system architecture and making more complex queries. A discussion of results, other related work and conclusions are then given.

2 Dimension Views of a Data Graph

The data cube model is the conceptual framework upon which standard OLAP user operations are built. Data cube operations such as slice and dice combine in ways that are consistent with each other and are consistent with the model. Alternative models would give rise to alternative sets of user operations. Such operations would also need to be consistent with each other and their model. This section introduces our data graph model and its selection operations. Later sections will show how these selection operations integrate with our multi-scale tree visualisation to support the aggregation of values through the dimension hierarchies.

Figure 1(a) shows a product sales data cube. It containing eight facts or cells that have been numbered c1 to c8. The dimension hierarchies shown on the cube sides are product, time and region. Each cell captures the number of products sold at a given time, in a given region for a given product type. Figure 1(b) shows a data graph, an alternative representation of the same information. The cells are laid out as a one dimensional array below the dimension hierarchies. Each cell is placed below the dimension categories it belongs to. The data graph only includes cells that have a value. The data cube is a visual representation of a multi-dimensional matrix, while the data graph is a visual representation of a partial order of dimension categories overlapping on a common set of leaf nodes. The focus of the former is the adjacency relationship between cells in the cube, which supports the comparison of values in adjacent cells, while the focus of the later is the dimension hierarchies that can support the display of value overviews and distributions.

Clearly the diagram shown in figure 1(b) could not be the direct basis for user interaction, even for this tiny dataset. Figure 1(b) becomes more readable if it is separated into independent views of each dimension, as we do in figure 1(c). It shows three views of the data cells, one for each dimension. For example the top view shows the product dimension view; which is the data graph with all non-product dimensions



(a) *data cube*

(b) *data graph*

(c) *data graph dimension views*

Fig. 1. The transformation of data representations. From (a) data cube representation on the top left to (b) data graph representation on the top right to (c) data graph dimension views on the bottom

categories removed. Users interact with the data graph through the independent dimension views. Like a data cube it is queried by making successive selections in each dimension. When a data cube dimension is set to a single category there is a unique result, a slice of the initial data cube. When a category is selected in a data graph dimension view, there are several ways the other dimension views could be affected. Three such coordinations are: (i) progressive, (ii) result only and (iii) global context.

Figure two shows a progressive coordination. To compare the number of sales of each product type in time one and region two, t1 then r2 are selected. The leftmost view shows the initial state, a full view of the eight cells in all dimensions. After time one selected cells c1, c2, c3 and c4 are highlighted in all dimension views. The time dimension view retains all cells, while the product and region dimension views retain only the selected four cells. The time view retains it's selection context, while product and region views show the intermediate result. After region two is selected, the result set includes only cells c3 and c4, which are highlighted. The cells included in the time and region views remain unchanged except for a highlighting change; the change of cells c1 and c2, from highlighted to not highlighted. While the product view has changed to include only the result cells c3 and c4 and their dimension values.

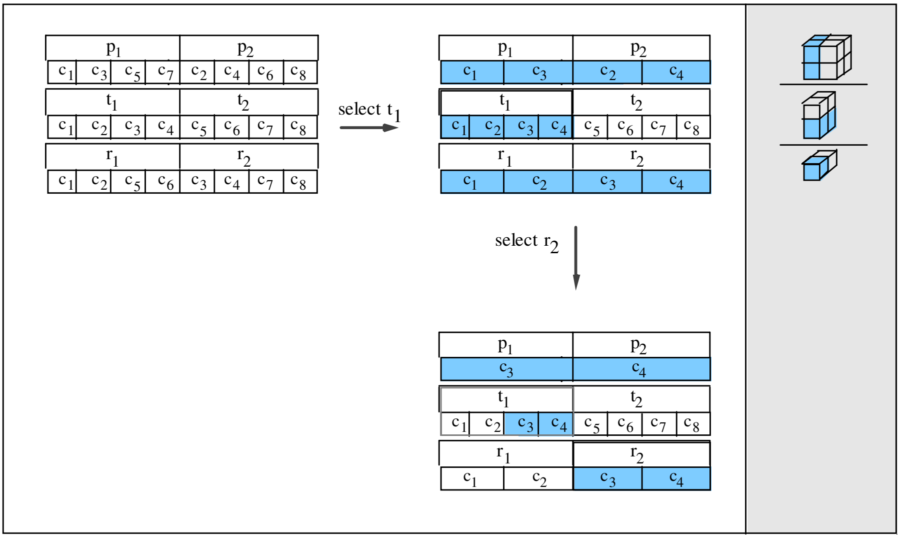


Fig. 2. Progressive query views. Dimension categories t1 then r2 are selected

The progressive coordination preserves the state of each dimension selection, so categories that were not selected remain visible for later selection. After t1 and r2 were selected, their alternatives t2 and r1 remain visible for later selection. The coordination works in the following way. As a query proceeds dimension categories are selected, setting that dimension. Dimensions that were set earlier are called prior dimensions. Dimensions that have not been *set*, are called unset. Each dimension view includes those cells that are in categories that have been selected in all prior dimensions, that is the intersection of category selections of prior dimensions. If a dimension selection is changed, all later dimensions are affected. For example, if the time dimension selection was changed to t2 the region dimension view would then change to show cells c5, c6, c7 and c8. Category r2 would remain selected, but the cells highlighted in it would change to c7 and c8, while the product dimension would change to include only c7 and c8.

Figure three shows the result only coordination after t1 and r2 were selected. Only cells that are in the result set and their dimension categories appear. Figure four shows the global context coordination where all dimension categories and cells are shown but where cells in the result set are highlighted. This allows a user to see the distribution of result cells which are highlighted in each dimension hierarchy, showing which categories have some matches and which do not.

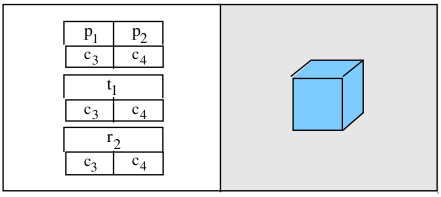


Fig. 3. The result only view

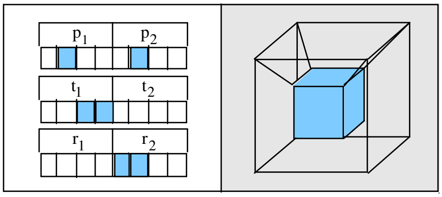


Fig. 4. The global context view

3 Our Multi-scale Hierarchy Visualisation

Our interface uses a multi-scale tree visualisation for presenting each dimension view. It addresses several requirements. Users need to select dimension categories as they drill-down through the data. They also need to see intermediate results in a way that shows aggregated values and result distributions to guide the next query step. Figure five shows our interface presenting the web log data in four dimension views: time, downloaded tools, visitor address and referrer address. Each view uses a variation of a nested tree layout where parents appear above their children.

The top panel shows the time dimension in four levels. The total time period, the monthly trend, the daily trend and the individual visits at the lowest level. Both the monthly and daily trends can be read. The monthly trend is slightly increasing while the daily trend is highly variable. While the time dimension is a balanced tree, the other dimensions are not.

In the top time view, box fill is varied at each tree level to create a nested bar chart, where both monthly and daily trends can be read. In the bottom visitor and referrer views, box widths are varied to show proportions, so the wide yahoo bar in the refer-

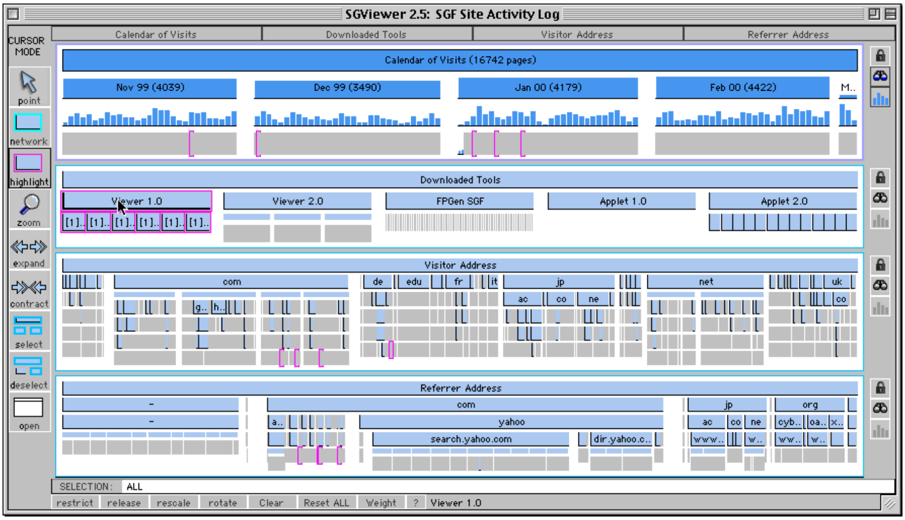


Fig. 5. The complete view of the log data

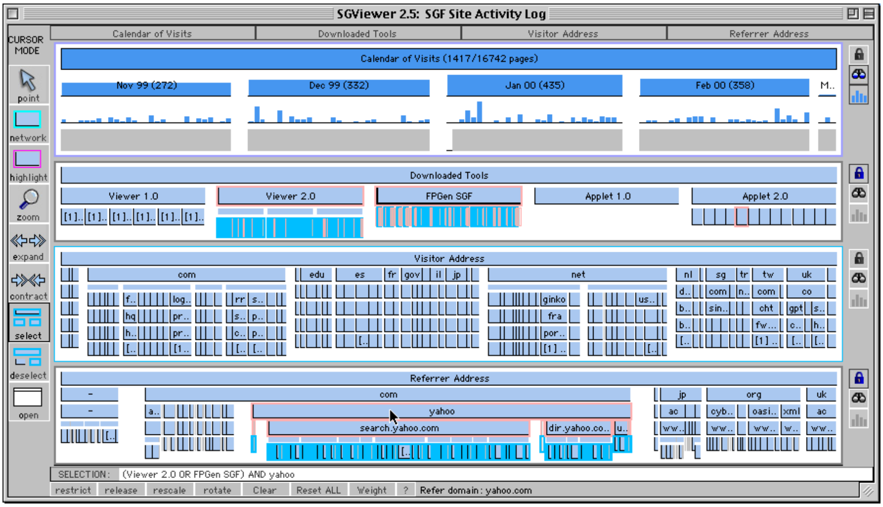
rer view indicates yahoo was the biggest referrer. The lowest level of each dimension view contains the actual visits.

In figure five, the visits where viewer 1.0 tool was downloaded have been highlighted. The six highlighted visits are shown in the download view and other views with pink borders. The borders have a fixed width so they are visible even when the node they enclose is not. For example most of these highlighted visits were in January, shown by three pink leaf box borders under the January box.

The tree layout is configurable. The time tree was laid out with each day box set to equal width. In the downloads tree each sibling was set to equal width. In the visitor and referrer trees, node proportions reflect the sum of leaf node values (visit page hits). Tree configuration is done at startup via attributes in the input SGF file, but it can also be changed on demand via a pop-up menu. To see some first level visitor addresses in more detail, say the addresses between “jp” and “net”, an animated zoom is used. This could be combined with changing the layout to equal sibling widths and a change to a bar chart mode to present relative values with bar heights so narrow visitors addresses are equally visible.

4 Exploring the Data

This section shows how our tool supports the progressive, global and result view coordinations. Consider the following query: a user wants to find the main contributor to site visits where viewer 2.0 or FPGen were downloaded. To answer this, first viewer 2.0 and FPGen in the downloaded tool view are selected. The other views then change to show only these visits. The biggest category in an unset view is yahoo in the referrer view, so it is selected. The resulting trend of visits referred by yahoo where a download of viewer 2.0 or FPGen occurred is shown in the time view. This is shown in figure six where a text field below the bottom view records the selection. A



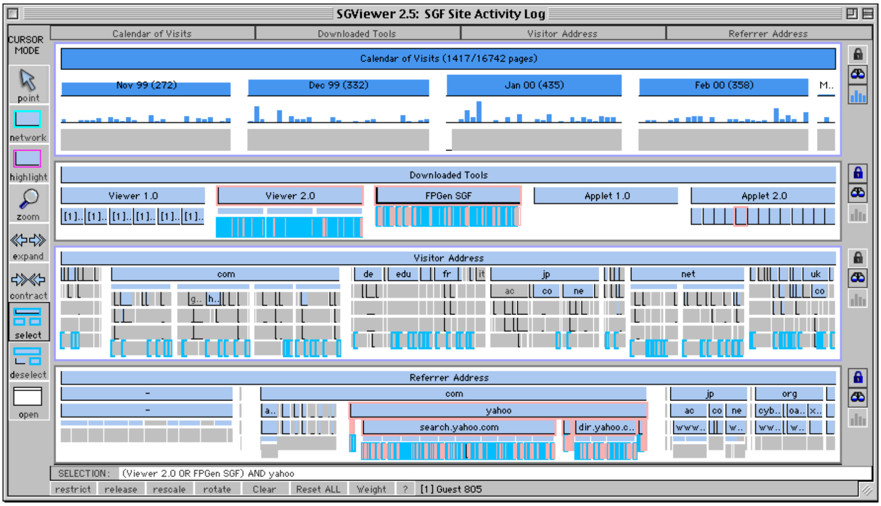
SELECTION: (Viewer 2.0 OR FPGen SGF) AND yahoo

Refer domain: yahoo.com

Fig. 6. The progressive view after viewer 2.0 or FPGen and yahoo were selected

user could explore further by selecting .com in the visitor address view to drill-down further into the trend, or the user could change their selection of downloads or referrer address as the alternative tools and referrers and their proportions are still visible.

The time bar chart in figure six view is a special case; all visits appear in this view so trends can be read correctly, but only those visits from yahoo where viewer 2.0 or FPGen was downloaded contribute to the bar chart bars. Unlike our abstract dimension views in section two, a blue panel border is used rather than highlighting all nodes in the other unset view, the visitor address view.

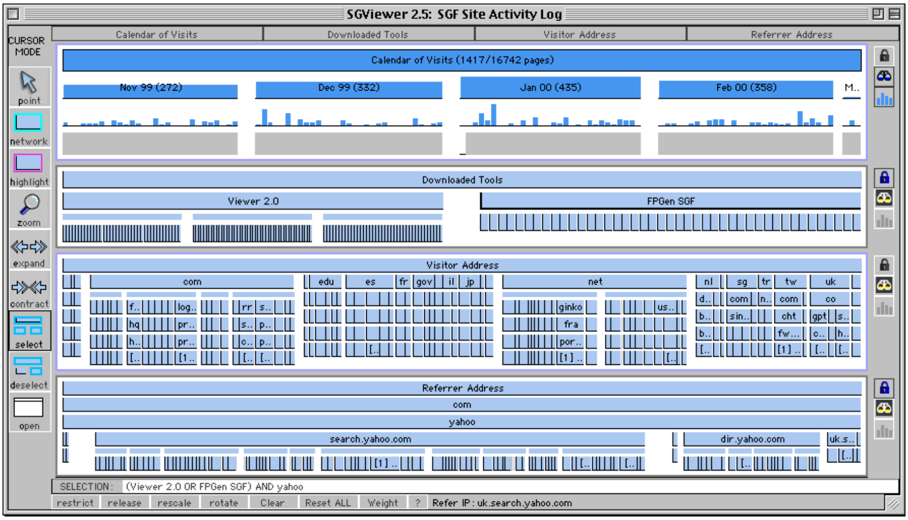


SELECTION: (Viewer 2.0 OR FPGen SGF) AND yahoo

Fig. 7. The global view after viewer 2.0 or FPGen and yahoo were selected

A user can press the space bar at any time to get a global or result view. Figure seven shows the global view of the previous query. All dimension hierarchies are fully visible and visits that are in the query result are highlighted with a blue border. This allows clustering of results within parts of the dimension hierarchy to be seen at a glance. There is an even distribution of results across most of the visitor address dimension. However there are few visits in the result set from ".jp" where there are few visits with borders and none from "ac.jp" where there are no visits with borders below. Showing a query result distribution via borders on leaf nodes is approximate only, as it gives feedback on the number of the results not their values. Further when a result set contains adjacent leaf nodes they will be represented with only one visible border when their combined width is less than one pixel. Counting results via the presence of borders is a rough approximation. The inverse of the global view, the result view is shown in figure eight, where all dimensions are restricted to the result set.

By using the global (binoculars) button on the RHS of each dimension view a user can selectively see the global context or result only view for that dimension. By using the weight button at the bottom of the frame a user can toggle between aggregating the sum of page hits or the count of visits shown through the relative box width proportions in each panel's tree layout.



SELECTION: (Viewer 2.0 OR FPGen SGF) AND yahoo

Refer IP: uk.search.yahoo.com

Fig. 8. The result view after viewer 2.0 or FPGen and yahoo were selected

5 Architecture

The input to SGViewer is a structured graph format (SGF) XML file. This file contains both the web site log dataset and configuration information that tells the viewer how to present each dimension. The log SGF file is created by another java program, logConverter that inputs a standard apache web log (set to include refer pages). Our web site was one of many served by the local web server; so grep was used to filter the servers log for our site's log subset. Figure nine provides an overview of this.

Some approximations were made in creating the web site log dataset. We relied on Internet Protocol (IP) addresses to identify each visitor. However many web users can go through the same proxy server, while some proxy systems can send page requests from a single user through multiple proxy servers. Also there can also be web caches is various places, from a visitors own browsers cache to a large organisation's firewall caching requests to minimise internet bandwidth usage. Some of these problems can be solved by using cookies. But overall, for a small site like ours these approximations seemed reasonable, while for the accurate tracking of a large site they would not be.

The SGF format used by the viewer does not contain any web site log specific tags. It encodes an arbitrary simple structured graph: a set of nodes, dimensions for these nodes, and possibly some networks as well. The viewer has been used for other applications such as multi-classification sitemaps by inputting an SGF file with different content. The viewer can operate as either a java application or applet. When used as an applet, URL's can be associated with nodes so that clicking a node in open cursor mode opens the associated URL in a new browser window. A limitation of this application/applet design is the viewer supports only around 10,000 data points in five dimensions. Beyond this excessive amounts of memory are consumed.

The viewer also has the potential to support other classification visualisations. Each panel visualisation component implements a limited java interface that is passed the panel's context, classification, the full dataset, and outputs the panel's selection set.

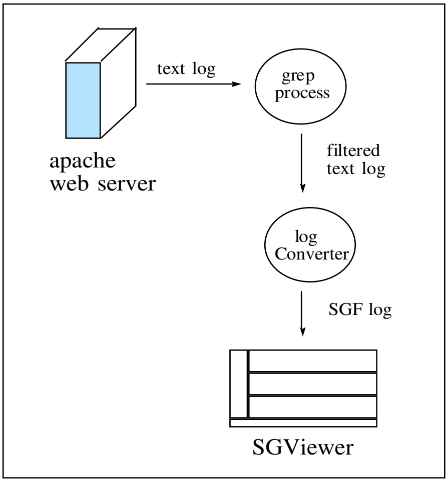
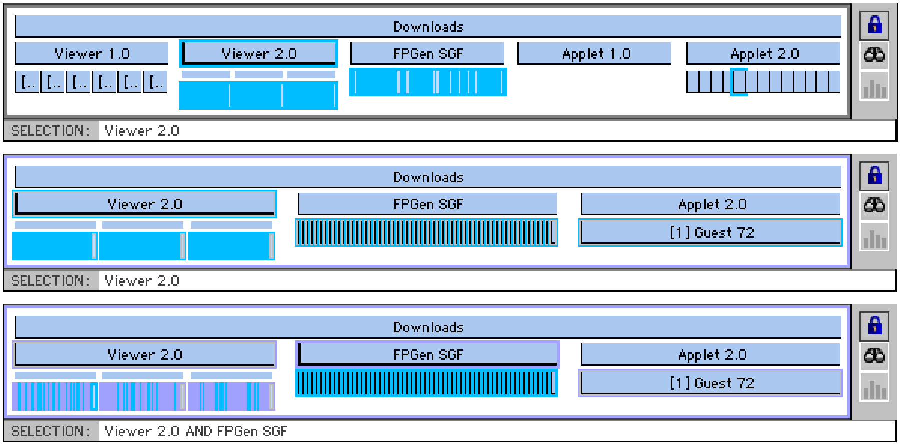


Fig. 9. The system for transferring web site log data to SGViewer

6 Making More Complex Queries

The queries demonstrated so far have been of the form: a conjunction of disjunctions. This should be sufficient when each dimension is a tree. But the viewer also supports dimensions that are arbitrary rooted partial orders, such as the downloads dimension.



SELECTION: Viewer 2.0

SELECTION: Viewer 2.0

SELECTION: Viewer 2.0 AND FPGen SGF

Fig. 10. Making an intra-classification AND query. In the downloads panel the user selects viewer 2.0 (top), clicks the restrict button located at the bottom left of the frame (middle) then selects FPGen (bottom)

The downloading of multiple tools during a single visit requires visit facts to have multiple parents, one for each tool downloaded. In such cases, queries such as what is the trend for visits where both viewer 2.0 and FPGen were downloaded are reasonable. Further, to find out what the specific interest in FPGen was, a query like: what is the trend for visits where FPGen but not viewer 2.0 was downloaded, could be asked. The former query requires an intra-classification AND operation, while the latter requires a classification AND NOT operation.

Intra-classification AND operations are made via a local restrict. Figure ten shows the progress of a local restrict operation in the downloads panel of our example. The effect of the restrict operation was to exclude from the download panel all visit elements that are not selected, that is to exclude all visits where viewer 2.0 was not downloaded. Selections made after a restrict, select from the subset of the visits where viewer 2.0 was downloaded. The query text field would now show: viewer 2.0 and FPGen. However, local restrict is more difficult to use than standard queries where AND selection from different panel is used, as the restrict button must be explicitly pressed by a user. In some cases, the alternative of placing a dimension in multiple panels may be more acceptable, though there is a tradeoff of operation complexity (use of an explicit restrict) versus interface screen space (for multiple panels).

Portions of a dimension can also be deselected which allows AND NOT queries to be made. A user places the cursor in deselect mode, then clicks the dimension portion that will be excluded. For example, to set the downloads panel to visits where FPGen but not viewer 2.0 was downloaded, a user first selects viewer 2.0 then deselects FPGen.

7 Discussion

Our interface has been demonstrated many times and used by a number of graduate and undergraduate students. The informal feedback has been a small amount of training is needed to operate the interface. Both the tree visualisation used to present each dimension view and the coordination between dimension views is new. Learning the view coordination was the easiest. Once a user had clicked in different panels a few times they could see the effect on other panels via animated transitions. Learning the zoom controls was less discoverable even though zooming also uses animated transitions. Refining this will be future work.

Each dimension view has two functions (i) to support selection of dimension values and (ii) to present intermediate and overall query result overviews. These functions are best met when users are familiar with the dimension hierarchies, so they have a rough knowledge of not only of what is at each level of the hierarchy but also what the left to right ordering is. This would allow selections to be made more rapidly and overviews to be read with less need for zooming to for example read the full titles of nodes. For OLAP, an analysist could be expected to have or quickly acquire such familiarity with the dimension hierarchies.

The breadth of the hierarchy at each level will also affect readability. When a node has a large number of children say one hundred, there is not enough space to show child titles even when the parent node takes up the maximum panel width. A further zoom operation is needed. If the average breadth of the tree is low (say 10 to 25) the clustering of results in global views will be more readable at a glance.

8 Related Work

Our nested tree visualisation is a one-dimensional variation of treemaps [7] a 2D space filling nested tree layout. Polaris [10] which is table based, has applied multiscale visualisations to OLAP. Snap [3] is a configurable architecture for visualisation coordination that supports a drill-down coordination, but lacks explicit support for dimension hierarchies and the dynamic flexibility of our coordinations. Query previews [4] used intermediate query result distributions to guide each query step for relation queries done over a network.

A range of systems have supported progressive or global query coordinations without supporting dimension hierarchies. FocusTable [9] used a progressive query coordination within a spreadsheet based interface. Dynamic queries [1] used a slider for each dimension to make selections, and used a separate display such as a scatterplot for showing a global view of the result set. Attribute Explorer [8] also used separate sliders for each dimension, but unlike dynamic queries and like our viewer, provided a separate display for each dimension that showed a global data distribution.

9 Conclusions

In this paper we have introduced our data graph and interface techniques for querying it based on coordinated dimension views. Progressive, global context and result only coordinations were presented. The progressive coordination used unset dimension views as query previews, while retaining the selection context of set views so they could be varied more easily.

Selections in each dimension corresponded to OLAP slice and dice operations. Data aggregation at all levels was provided in each dimension view rather than explicit roll-up and drill-down operations. In both cases, zoom operations supported the selection and reading of detail. Finally, while our interface provides standalone support for data exploration, it could be integrated with a cross-tab, as its strengths in providing overview and retaining query context complement a cross-tab's results view.

References

1. Ahlberg, C. and Shneiderman, B., Visual information seeking: tight coupling of dynamic query filters with starfield displays. In Proceedings of CHI'94, (April, Boston, MA), ACM Press, 1995.
2. Liechti, O., Sifer, M., Ichikawa, T.: Structured Graph Format: XML metadata for describing web site structure. Proc. of the 7'th World Wide Web Conference, (1998) 11-32.
3. North, C., Shneiderman, B.: Snap-Together visualization: a user interface for coordinating visualizations via relational schemata. Proc. of Advanced Visual Interfaces, 2000.
4. Plaisant, C., Bruns, T., Doan, K., Shneiderman, B.: Interface and data architecture for query previews in networked information systems. ACM Transactions on Information Systems, 17, 3, (1999) 320-341.
5. Pedersen, T., Jensen, C.: Multidimensional Database Technology. IEEE Computer 34(12), 2001.

6. Sifer, M.: A visual interface technique for exploring OLAP data with coordinated dimension hierarchies. Proc. ACM Conference on Information and Knowledge Management, New Orleans, Nov. 2003, 532-535.
7. Shneiderman, B.: Tree visualisation with treemaps: a 2-d space-filling approach. ACM Transactions on Graphics, 11(1), Jan. 1992, 92-99.
8. Spence, R., Tweedie, L.: The Attribute Explorer: information synthesis via exploration. Interacting with Computers, 11(2), 1998, 137-146.
9. Spenke, M., Beilken, C. and Berlage T., FOCUS: the interactive table for product comparison and selection. In Proceedings of UIST'96, ACM Press, 1996, 41-50.
10. C. Stolte, D. Tang and P. Hanrahan. Query, Analysis, and Visualization of Hierarchically Structured Data using Polaris. Proc. ACM SIGKDD, July 2002.

Policies, Models, and Languages for Access Control

Sabrina De Capitani di Vimercati[1], Pierangela Samarati[1], and Sushil Jajodia[2]

[1] DTI - Università di Milano, 26013 Crema - Italy
{decapita, samarati}@dti.unimi.it
[2] George Mason University, Fairfax, VA 22030-4444
jajodia@gmu.edu

Abstract. Access control is the process of mediating every request to data and services maintained by a system and determining whether the request should be granted or denied. Expressiveness and flexibility are top requirements for an access control system together with, and usually in conflict with, simplicity and efficiency. In this paper, we discuss the main desiderata for access control systems and illustrate the main characteristics of access control solutions.

1 Introduction

One of the most important features of today's systems is the protection of their resources (i.e., data and services) against unauthorized disclosure (*secrecy*) and intentional or accidental unauthorized changes (*integrity*), while at the same time ensuring their accessibility by authorized users whenever needed (*no denials-of-service*) [30]. Considerable effort is being devoted to addressing various aspects of secrecy, integrity, and availability. However, historically, confidentiality has received the most attention, probably because of its importance in military and government applications. As a result, significant research has focused on achieving more expressive and powerful *access control systems*. Access control is the act of ensuring that a user accesses only what she is authorized to and no more. The development of an access control system requires the definition of the regulations according to which access is to be controlled and their implementation as functions executable by a computer system. This development process is usually carried out with a multi-phase approach based on the concepts of *security policy*, *security model*, and *security mechanism*. A policy defines the (high-level) rules according to which access control must be regulated. An access control model provides a *formal* representation of the access control security policy and its working. The formalization allows the proof of properties on the security provided by the access control system being designed [21]. A security mechanism defines the low level (software and hardware) functions that implement the controls imposed by the policy and formally stated in the model.

The traditional access control models used for describing the enforcement of confidentiality are based on the definition of access control rules, called *authoriza-*

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 225–237, 2005.

tions, which are of the form ⟨subject, object, operation⟩. These authorizations specify which operations can be performed on objects by which subjects. However, in today's systems the definition of an access control model is complicated by the need to formally represent complex policies, where access decisions depend on the application of different rules coming, for example, from laws practices, and organizational regulations. A security policy must then combine all the different regulations to be enforced [34] and, in addition, must consider all possible additional threats due to the use of computer systems. Given the complexity of the scenario, the simple authorization triple ⟨subject, object, operation⟩ is no more sufficient.

The remainder of this paper is organized as follows. Section 2 discusses the main features supported by modern access control policies and models. Section 3 presents recent approaches in the area of access control languages. Finally, Section 4 concludes the paper.

2 Policies and Models for Access Control

The access control service provided by the computer system should be expressive and flexible enough to accommodate all the different requirements that may need to be expressed, while at the same time be simple both in terms of use (so that specifications can be kept under control) and implementation (so to allow for its verification). In the following, we discuss the main features that an access control service should support.

2.1 Conditions and Groups

Even early approaches to authorization specifications allowed *conditions* to be associated with authorizations to restrict their validity. Conditions can make the authorization validity dependent on the satisfaction of some system predicates (*system-dependent* conditions) like the time or location of access. For instance, a condition can be associated with the bank-clerks' authorization to access accounts, restricting its application only from machines within the bank building and in working hours. Conditions can also constraint access depending on the content of objects on which the authorization is defined (*content-dependent* conditions). Content-dependent conditions can be used simply as way to determine whether or not an access to the object should be granted or as way to restrict the portion of the object that can be accessed (e.g., a subset of the tuples in a relation). This latter option is useful when the authorization object has a coarser granularity than the one supported by the data model [11]. Other possible conditions that can be enforced can make an access decision depend on accesses previously executed (*history dependent* conditions).

Another feature usually supported even by early approaches is the concept of *user groups* (e.g., Employees, Programmers, Consultants). Groups can be nested and need not be disjoint. Figure 1 illustrates an example of user-group hierarchy. Support of groups greatly simplifies management of authorizations, since a single authorization granted to a group can be enjoyed by all its members.

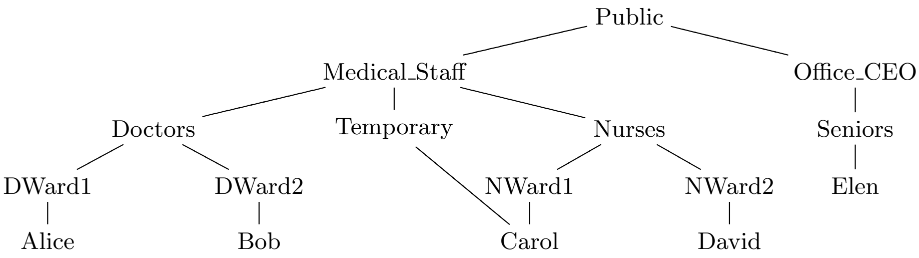


Fig. 1. An example of user-group hierarchy

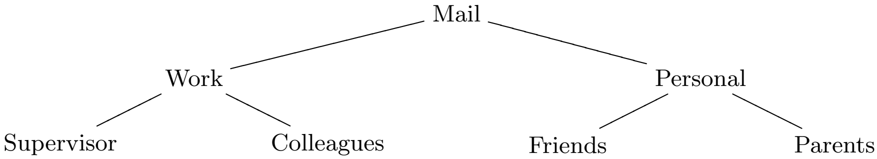


Fig. 2. An example of object hierarchy

Later efforts moved to the support of groups on all the elements of the authorization triple (i.e., subject, object, and operation), where, typically, groups are abstractions hierarchically organized. For instance, in an operating system the hierarchy can reflect the logical file system tree structure, while in object-oriented system it can reflect the class (is-a) hierarchy. Figure 2 illustrates an example of object hierarchy. Even operations can be organized hierarchically, where the hierarchy may reflect an implication of privileges (e.g., write is more powerful than read [29]) or a grouping of sets of privileges (e.g., a "writing privileges" group can be defined containing write, append, and undo [32]). These hierarchical relationships can be exploited *i)* to support preconditions on accesses (e.g., in Unix a subject needs the execute privilege on a directory to access the files within it), or *ii)* to support authorization implication, that is, authorizations specified on an abstraction apply to all its members. Support of abstractions with implications provides a short hand way to specify authorizations, clearly simplifying authorization management. As a matter of fact, in most situations the ability to execute privileges depends on the membership of users into groups or objects into collections: translating these requirements into basic triples of the form ⟨user, object, operation⟩ that then have to be singularly managed is a considerable administrative burden, and makes it difficult to maintain both satisfactory security and administrative efficiency.

2.2 Positive and Negative Authorizations

Although there are cases where abstractions can work just fine, many will be the cases where exceptions (i.e., authorizations applicable to all members of a group but few) will need to be supported. This observation has brought to the combined support of both *positive* and *negative* authorizations. Traditionally, positive and negative authorizations have been used in mutual exclusion corresponding to two classical approaches to access control, namely:

Closed policy: authorizations specify permissions for an access. The closed policy allows an access if there exists a positive authorization for it, and denies it otherwise.

Open Policy: (negative) authorizations specify denials for an access. The open policy denies an access if there exists a negative authorization for it, and allows it otherwise.

The open policy has usually found application only in those scenarios where the need for protection is not strong and by default access is to be granted. Most systems adopt the closed policy, which, denying access by default, ensures better protection; cases where information is public by default are enforced with a positive authorization on the root of the subject hierarchy (e.g., Public).

The combined use of positive and negative authorizations was therefore considered as a way to conveniently support exceptions. To illustrate, suppose we wish to grant an authorization to all members of a group composed of one thousand users, except to one specific member Alice. In a closed policy approach, we would have to express the above requirement by specifying a positive authorization for each member of the group except Alice.[1] However, if we combine positive and negative authorizations we can specify the same requirement by granting a positive authorization to the group and a negative authorization to Alice.

The combined use of positive and negative authorizations brings now to the problem of how the two specifications should be treated:

– what if for an access no authorization is specified? (*incompleteness*)
– what if for an access there are both a negative and a positive authorization? (*inconsistency*)

Completeness can be easily achieved by assuming that one of either the open or closed policy operates as a *default*, and accordingly access is granted or denied if no authorization is found for it. Note that the alternative of explicitly requiring completeness of the authorizations is too heavy and complicates administration.

Conflict resolution is a more complex matter and does not usually have a unique answer [18, 25]. Rather, different decision criteria could be adopted, each applicable in specific situations, corresponding to different policies that can be implemented. Examples of different conflict resolution policies are given below.

[1] In an open policy scenario, the dual example of all users, but a few, who have to be denied an access can be considered.

Denials Take Precedence. Negative authorizations are always adopted when a conflict occurs (it satisfies the "fail safe principle"). In other words, the principle says that if we have one reason to authorize an access, and another to deny it, then we deny it.

Most Specific Takes Precedence. A natural and straightforward policy is the one stating that "the most specific authorization should be the one that prevails"; after all this is what we had in mind when we introduced negative authorizations in the first place (our example about Alice). Although the most-specific-takes-precedence principle is intuitive and natural and likely to fit in many situations, it is not enough. As a matter of fact, even if we adopt the argument that the most specific authorization always wins (and this may not always be the case) it is not always clear what more specific is:

– what if two authorizations are specified on non-disjoint, but non-hierarchically related groups (e.g., NWard1 and NWard2 in Figure 1)?
– what if for two authorizations the most specific relationship appear reversed over different domains? For instance, consider authorizations (Doctors, read+, Mail) and (Medical_Staff, read−, Personal); the first has a more specific subject, while the second has a more specific object (see Figures 1 and 2).

Most Specific Along a Path Takes Precedence. This policy considers an authorization specified on an element x as overriding an authorization specified on a more general element y only for those elements that are members of y because of x. Intuitively, this policy takes into account the fact that, even in the presence of a more specific authorization, the more general authorization can still be applicable because of other paths in the hierarchy. For instance, consider the group hierarchy in Figure 1 and suppose that for an access a negative authorization is granted to Medical_Staff while a positive authorization is granted to Nurses. What should we decide for Carol? On the one side, it is true that Nurses is more specific than Medical_Staff; on the other side, however, Carol belongs to Temporary, and for Temporary members the negative authorization is not overridden. While the most-specific-takes-precedence policy would consider the authorization granted to Medical_Staff as being overridden for Carol, the most-specific-along-a-path considers both authorizations as applicable to Carol. Intuitively, in the most-specific-along-a-path policy, an authorization propagates down the hierarchy until overridden by a more specific authorization [15].

Priority Level. The most specific argument does not always apply. For instance, an organization may want to be able to state that consultants should not be given access to private projects, *no exceptions allowed*. However, if the most specific policy is applied, any authorization explicitly granted to a single consultant will override the denial specified by the organization. To address situations like this, some approaches proposed adopting *explicit priorities*; however, these solutions do not appear viable as the authorization specifications may result not always clear.

Positional. Other approaches (e.g., [32]) proposed making authorization priority dependent on the *order in which authorizations are listed* (i.e., the authorizations that is encountered first applies). This approach, however, has the drawback that granting or removing an authorization requires inserting the authorization in the proper place in the list. Beside the administrative burden put on the administrator (who, essentially, has to explicitly solve the conflicts when deciding the order), specifying authorizations implies explicitly writing the ACL associated with the object, and may impede delegation of administrative privileges.

Grantor- or Time-Dependent. Other possible ways of defining priorities can make the authorization's priority dependent on the *time* at which the authorizations was granted (e.g., more recent authorizations prevails) or on priorities between the *grantors*. For instance, authorizations specified by an employee may be overridden by those specified by her supervisor; the authorizations specified by an object's owner may override those specified by other users to whom the owner has delegated administrative authority.

As it is clear from this discussion, different approaches can be taken to deal with positive and negative authorizations. Also, if it is true that some solutions may appear more natural than others, none of them represents "the perfect solution". Whichever approach we take, we will always find one situation for which it does not fit. Also, note that different conflict resolution policies are not mutually exclusive. For instance, one can decide to try solving conflicts with the most-specific-takes-precedence policy first, and apply the denials-take-precedence principle on the remaining conflicts (i.e., conflicting authorizations that are not hierarchically related).

The support of negative authorizations does not come for free, and there is a price to pay in terms of authorization management and less clarity of the specifications. However, the complications brought by negative authorizations are not due to negative authorizations themselves, but to the different semantics that the presence of permissions and denials can have, that is, to the complexity of the different real world scenarios and requirements that may need to be captured. There is therefore a trade-off between expressiveness and simplicity. For this reason, most current systems adopting negative authorizations for exception support impose specific conflict resolution policies, or support a limited form of conflict resolution. (e.g., see the Apache server [1] where authorizations can be positive and negative and an ordering can be specified dictating how negative and positive authorizations are to be interpreted). More recent approaches are moving towards the development of flexible frameworks with the support of multiple conflict resolution and decision policies.

2.3 Attribute-Based Specifications

In an open system like the Internet, the different parties (clients and servers) that interact with each other to offer services are usually strangers. They have no preexisting relationship and are not in the same security domain. Therefore, on the one side the server may not have all the information it needs to decide whether or not an access should be granted. On the other side, however, the client may not know which information she needs to present to a (possibly just encountered) server to get access. All this requires a new way of enforcing the access control process, which cannot be assumed anymore to operate with a given prior knowledge and return a yes/no access decision. Rather, the access control process should be able to operate without a priori knowledge of the party requesting access and return the information of the requisites that it requires be satisfied for the access to be allowed [2, 17]. Also, the traditional "identity-based access control models" where subjects and objects are usually identified by unique names are not appropriate in this setting. Instead, attributes other than identity are useful in determining the party's trustworthiness. In this context, access restrictions to the data/services should be expressed by policies that specified the properties (attributes) that a requester should enjoy to gain access to the data/services. Some proposals have been developed that use *digital certificates*. Traditionally, the widely adopted digital certificate has been the *identity certificate*. An identity certificate is an electronic document used to recognize an individual, a server, or some other entity, and to connect that identity with a public key [3, 4, 8]. More recent research and development efforts have resulted in a second kind of digital certificate, the *attribute certificate* [14] that can be used for supporting and attribute-based access control. An attribute certificate has a structure similar to an identity certificate but contains attributes that specify access control information associated with the certificate holder (e.g., group membership, role, security clearance). One of the most important aspects that attribute-based access control policies should support is the ability to specify accesses to a *collection* of services based on a *collection* of attributes. In this context, logic programming provides a convenient, expressive, and well-understood framework in which to work with authorization policy. Jajodia et. al [33] propose a framework that models an attribute-based access control system using logic programming with set constraints of a computable set theory. More precisely, the set theory used in this approach is CLP($\mathcal{SET}$), the *hereditarily finite* and computable set theory developed by Dovier et al. [13]. Here, sets are constructed out of a finite universe by applying operators such as $\cap$, $\cup$, and so on. A policy can refer to both attributes and services, and a two sorted first order language with set variables is then used. The terms are constructed in the usual way by means of variables and functions. Also, the approach supports two kinds of predicates: those used to specify the computation domain and those used to specify its sub-domain of constraints. To reduce the runtime inefficiency of constrained logic programs, which is due to the backtracking through program clauses, two techniques are used. The first technique consists in transforming any attribute-based access control policy into one with less backtracking but the same semantics. The second technique consists in materializing commonly accessed predicates instances.

Bonatti and Samarati in [6] propose a uniform framework for regulating service access and information disclosure in an open, distributed network system like the Web. Access regulations are specified as logical rules, where some predicates are explicitly identified. Attribute certificates are modeled as *credential expressions* of the form $\texttt{credential\_name(attribute\_name}_1\texttt{ = value\_term}_1\texttt{),...,}$ $\texttt{attribute\_name}_n\texttt{ = value\_term}_n$, where $\texttt{credential\_name}$ is the attribute credential name, $\texttt{attribute\_name}_i$ is the attribute name, and $\texttt{value\_term}_i$ is either a ground value or a variable. Besides credentials, the proposal also allows to reason about declarations (i.e., unsigned statements) and user-profiles that the server can maintain and exploit for taking the access decision. Communication of requisites to be satisfied by the requester is based on a filtering and renaming process applied on the server's policy, which exploits partial evaluation techniques in logic programs.

Although attribute-based access control polices allow the specifications of access control rules with reference to generic attributes or properties of the involved parties, they do not fully exploit the semantic power and reasoning capabilities of emerging web applications. The next step in the development of expressive and powerful access control models and policies should then be the support of access control rules based on the rich ontology-based metadata associated with both the subjects accessing the resources and the resources themselves [9].

3 Languages for Access Control

Languages for access control aim to support the expression and the enforcement of policies [30]. Many of these languages are used for expressing generic assertions about subjects (principals) such as the association of a principal with a public key, the membership of a principal in a group, or the right of a principal to perform a certain operation at a specified time [27]. They also serve for combining policies from many sources, and thus for making authorization decisions [5]. More broadly, the languages sometimes aim to support trust management [6, 31, 35]. Also, with the increasing number of applications that either use XML as their data model, or export relational data as XML data, it becomes critical to investigate the problem of access control for XML. To this purpose, many XML-based access control language have been proposed [7, 10, 20, 26]. In particular, one of the most relevant XML-based access control language is the eXtensible Access Control Markup Language (XACML). The purpose of XACML is the expression of authorization policies in XML against objects that are themselves identified in XML. XACML covers both an access control policy language and a request/response language. So besides defining who can do what and when by generating the corresponding policies, it is also possible to express access requests and responses in XACML. The eXtensible Access Control Markup Language (XACML) version 1.0 [26] has been an OASIS standard since 2003. Improvements have been made to the language and incorporated in version 2.0 [28].

Several of the most recent language designs rely on concepts and techniques from logic, specifically from logic programming: Li et al.'s D1LP and RT [22, 23, 24], Jim's SD3 [19], and DeTreville's Binder [12]. The expressive power and the formal foundations of logical formalisms are appealing in this context. Some researchers and practitioners object that logic-based specifications may be complicated or even intimidating to some users. Security administrators and end users need simple and user-friendly approaches that allow them to easily understand the system behavior and maintain control over security specifications. It is tempting to conclude that logic-based approaches are not applicable, but then one would also give up all the advantages of logic-based formulations that enjoy a combination of clean foundations (hence, formal guarantees), flexibility, expressiveness, and declarativeness (so that users are not required to have any programming ability). On the contrary, we believe that a careful choice of syntax makes logic-based specifications accessible to a wide spectrum of users. In the following section, we present one of the most representative logic-based languages for access control.

3.1 A Flexible Authorization Framework

Jajodia et al. [18] worked on a proposal for a logic-based language that attempted to balance flexibility and expressiveness on one side, and easy management and performance, on the other. This language is a good representative for this line of work. It allows representing different policies and protection requirements, while at the same time providing understandable specifications, clear semantics, and bearable data complexity. Their proposal for a *flexible authorization framework* (FAF) corresponds to a polynomial (quadratic) time data complexity fragment of default logic.

In FAF, policies are divided into four decision stages, corresponding to the following policy components (Figure 4).

- *Authorization Table.* This is the set of explicitly specified authorizations.
- The *propagation policy* specifies how to obtain new derived authorizations from the explicit authorization table. Typically, derived authorizations are obtained according to hierarchy-based derivation. However, derivation rules are not restricted to this particular form of derivation.
- The *conflict resolution policy* describes how possible conflicts between the (explicit and/or derived) authorizations should be solved. Possible conflict resolution policies include *no-conflict* (conflicts are considered errors), *denials take precedence* (negative authorizations prevail over positive ones), *permissions-take-precedence* (positive authorizations prevail over negative ones), and *nothing-takes-precedence* (the conflict remains unsolved). Some forms of conflict resolutions can be expressed within the propagation policy, as in the case of overriding (also known as *most-specific-takes precedence*).
- A *decision policy* defines the response that should be returned to each access request. In case of conflicts or gaps (i.e. some access is neither authorized nor denied), the decision policy determines the answer. In many systems, decisions assume either the open or the closed form (by default, access is granted or denied, respectively).

Starting from this separation, the authorization specification language of FAF takes the following approach:

- The authorization table is viewed as a database.
- Policies are expressed by a restricted class of stratified and function-free normal logic programs called *authorization specifications*.
- The semantics of authorization specifications is the stable model semantics [16]. The structure of authorization specifications guarantees stratification and hence, stable model uniqueness and PTIME computability.

The four decision stages correspond to the following predicates. (Below s, o, and a denote a subject, object, and action term, respectively, where a term is either a constant value in the corresponding domain or a variable ranging over it).

`cando`(**o**,**s**,±**a**) represents authorizations explicitly inserted by the security administrator. They represent the accesses that the administrator wishes to allow or deny (depending on the sign associated with the action).

`dercando`(**o**,**s**,±**a**) represents authorizations derived by the system using logic program rules.

`do`(**o**,**s**,±**a**) handles both conflict resolution and the final decision.

Moreover, a predicate `done` keeps track of the history of accesses (for example, this can be useful to implement a Chinese Wall policy), and a predicate `error` can be used to express integrity constraints. In addition, the language has a set of predicates for representing hierarchical relationships (`hie`-predicates) and additional application-specific predicates, called `rel`-predicates. Hierarchical predicates represent hierarchical relationships within the different components of the system (objects, subjects, or access modes). For instance, in most realistic systems, data items are organized hierarchically. For example, in a file system, the basic objects are files, but these files are typically organized in a hierarchical directory structure. Similarly, in an object-oriented database, the objects being accessed are organized into an object hierarchy. In an analogous way, authorization subjects can be basic users or hierarchical groups in which they are organized. Application-specific predicates capture the possible different relationships, existing between the elements of the data system, that may need to be taken into account by the access control system. Examples of `rel`-predicates are `owner(user, object)`, which models ownership of objects by users, or `supervisor(user1, user2)`, which models responsibilities and control within the organizational structure.

Authorization specifications are stated as logic rules defined over the above predicates. To ensure stratifiability, the format of the rules is restricted as illustrated in Figure 3. Note that the adopted strata reflect the logical ordering of the four decision stages.

The authors of [18] present a materialization technique for producing, storing, and updating the stable model of the policy. The model is computed on the initial specifications and updated with incremental maintenance strategies.

Note that the clean identification and separation of the four decision stages can be regarded as a basis for a policy specification methodology. In this sense,

Stratum	Predicate	Rules defining predicate
0	`hie`-predicates `rel`-predicates `done`	Base relations. Base relations. Base relation.
1	`cando`	Body may contain `done`, `hie`- and `rel`-literals.
2	`dercando`	Body may contain `cando`, `dercando`, `done`, `hie`-, and `rel`- literals. Occurrences of `dercando` literals must be positive.
3	`do`	When head is of the form $\texttt{do}(\_, \_, +a)$ body may contain `cando`, `dercando`, `done`, `hie`- and `rel`- literals.
4	`do`	When head is of the form $\texttt{do}(o, s, -a)$ body contains just one literal $\neg\texttt{do}(o, s, +a)$.
5	`error`	Body may contain `do`, `cando`, `dercando`, `done`, `hie`-, and `rel`- literals.

Fig. 3. Rule composition and stratification of the proposal in [18]

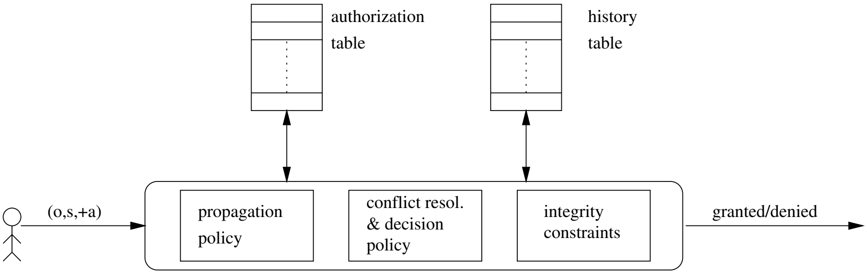


Fig. 4. Functional authorization architecture in [18]

the choice of a precise ontology and other syntactic restrictions (such as those illustrated in Figure 3) may assist security managers in formulating their policies.

4 Conclusions

Access control models, policies, and languages are constantly under development to obtain frameworks flexible and expressive enough so as to handle the specification and enforcement of security requirements of many emerging applications and real-world scenarios. In this paper, we presented the main features that modern access control models and policies should support and discussed recent proposals in the area of access control languages.

References

1. Apache http server version 2.0. http://www.apache.org/docs-2.0/misc/tutorials.html.
2. C. Bettini, S. Jajodia, S. Wang, and D. Wijesekera. Provisions and obligations in policy rule management and security applications. In *Proc. 28th International Conference on Very Large Data Bases*, Hong Kong, China, August 2002.
3. M. Blaze, J. Feigenbaum, J. Ioannidis, and A.D. Keromytis. The role of trust management in distributed systems security. *Secure Internet Programming: Issues in Distributed and Mobile Object Systems. Springer Verlag LNCS State-ofthe- Art series,*, 1998.
4. M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized trust management. In *Proc. of the 1996 IEEE Symposiumon Security and Privacy*, Oakland, CA, USA, May 1996.
5. P. Bonatti, S. De Capitani di Vimercati, and P. Samarati. An algebra for composing access control policies. *ACM Transactions on Information and System Security*, 5(1):1–35, February 2002.
6. P. Bonatti and P. Samarati. A unified framework for regulating access and information release on the web. *Journal of Computer Security*, 10(3):241–272, 2002.
7. D. Box et al. *Web services policy framework (WS-Policy) version 1.1.*, May 2003. http://msdn.microsoft.com/library/en-us/dnglobspec/html/ws-policy.asp.
8. Y-H. Chu, J. Feigenbaum, B. LaMacchia, P. Resnick, and M. Strauss. Referee: trust management forweb applications. *WorldWide Web Journal*, 2(3):706–734, 1997.
9. E. Damiani, S. De Capitani di Vimercati, C. Fugazza, and P. Samarati. Extending policy languages to the semantic web. In *Proc. of the International Conference on Web Engineering*, Munich, Germany, July 2004.
10. E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. A fine-grained access control system for XML documents. *ACM Transactions on Information and System Security (TISSEC)*, 5(2):169–202, May 2002.
11. C.J. Date. *An Introduction to Database Systems.* Addison-Wesley, 6th edition, 1995.
12. J. DeTreville. Binder, a logic-based security language. In *Proc. of the 2001 IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 2002.
13. A. Dovier, C. Piazza, E. Pontelli, and G. Rossi. Sets and constraints logic programming. *ACM Transactions of Programming Languages and Systems*, 22(5):861–931, September 2000.
14. S. Farrell and R. Housley. An internet attribute certificate profile for authorization. RFC 3281, April 2002.
15. E.B. Fernandez, E. Gudes, and H. Song. A model for evaluation and administration of security in object-oriented databases. *IEEE Transaction on Knowledge and Data Engineering*, 6(2):275–292, 1994.
16. M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *Proc. of the 5th International Conference and Symposium on Logic Programming*, pages 1070–1080, Cambridge, Massachusetts, 1988. The MIT Press.
17. S. Jajodia, M. Kudo, and V.S. Subrahmanian. Provisional authorizations. In Anup Ghosh, editor, *E-Commerce Security and Privacy*, pages 133–159. Kluwer Academic Publishers, Boston, 2001.
18. S. Jajodia, P. Samarati, M.L. Sapino, and V.S. Subrahmanian. Flexible support for multiple access control policies. *ACM Transactions on Database Systems*, 26(2):214–260, June 2001.

19. T. Jim. Sd3: A trust management system with certified evaluation. In *Proc. of the 2001 IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 2001.
20. M. Kudoh, Y. Hirayama, S. Hada, and A. Vollschwitz. Access control specification based on policy evaluation and enforcement model and specification language. In *Symposium on Cryptograpy and Information Security, SCIS'2000*, 2000.
21. C.E. Landwehr. Formal models for computer security. *ACM Computing Surveys*, 13(3):247–278, 1981.
22. N. Li, B.N. Grosof, and Feigenbaum. Delegation logic: A logic-based approach to distributed authorization. *ACM Transactions on Information and System Security*, 6(1):128–171, February 2003.
23. N. Li and J.C. Mitchell. Datalog with constraints: A foundation for trust-management languages. In *Proc. of the Fifth International Symposium on Practical Aspects of Declarative Languages (PADL 2003)*, New Orleans, LA, USA, January 2003.
24. N. Li, J.C. Mitchell, and W.H. Winsborough. Design of a role-based trust-management framework. In *Proc. of the IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 2002.
25. T. Lunt. Access control policies: Some unanswered questions. In *IEEE Computer Security Foundations Workshop II*, pages 227–245, Franconia, NH, June 1988.
26. OASIS. *eXtensible Access Control Markup Language (XACML) Version 1.0*, 2003. http://www.oasis-open.org/committees/xacml.
27. OASIS. *Security Assertion Markup Language (SAML) V1.1*, 2003. http://www.oasis-open.org/committees/security/.
28. OASIS. *eXtensible Access Control Markup Language (XACML) Version 2.0*, 2004. http://www.oasis-open.org/committees/xacml.
29. F. Rabitti, E. Bertino, W. Kim, and D. Woelk. A model of authorization for next-generation database systems. *ACM TODS*, 16(1):89–131, March 1991.
30. P. Samarati and S. De Capitani di Vimercati. Access control: Policies, models, and mechanisms. In R. Focardi and R. Gorrieri, editors, *Foundations of Security Analysis and Design*, LNCS 2171. Springer-Verlag, 2001.
31. K.E. Seamons, M. Winslett, T. Yu, B. Smith, E. Child, J. Jacobson, H. Mills, and L. Yu. Requirements for policy languages for trust negotiation. In *Proc. of the 3rd International Workshop on Policies for Distributed Systems and Networks (POLICY 2002)*, Monterey, CA, June 2002.
32. H. Shen and P. Dewan. Access control for collaborative environments. In *Proc. Int. Conf. on Computer Supported Cooperative Work*, pages 51–58, November 1992.
33. L. Wang, D. Wijesekera, and S. Jajodia. A logic-based framework for attribute based access control. In *Proc. of the 2004 ACM Workshop on Formal Methods in Security Engineering*, Washington DC, USA, October 2004.
34. D. Wijesekera and S. Jajodia. A propositional policy algebra for access control. *ACM Transactions on Information and System Security*, 6(2):286–325, May 2003.
35. T. Yu, M. Winslett, and K.E. Seamons. Supporting structured credentials and sensitive policies through interoperable strategies for automated trust negotiation. *ACM Transactions on Information and System Security*, 6(1):1–42, February 2003.

Modelling Peer-to-Peer Data Networks Under Complex System Theory

Cyrus Shahabi and Farnoush Banaei-Kashani

Computer Science Department, University of Southern California,
Los Angeles, California 90089
{shahabi, banaeika}@usc.edu

A Peer-to-peer Data Network (PDN) is an open and evolving society of peer nodes that assemble into a network to pool and share their data (or more generally, their resources represented by data) for mutual benefit. By an interesting analogy to a democratic human society, when nodes join the PDN society, while they agree to follow a restricted set of common rules in interaction with their peers (i.e., the social rules governing the PDN society), they preserve their autonomy as individuals. For example, as part of their social obligations all PDN nodes (or at least those who are good PDN citizens) create and maintain connection with a set of neighbor nodes and participate in cooperative query processing (e.g., forwarding search queries for data discovery). Aside from the social rules, the PDN leaves the behavior of the individual nodes unregulated and flexible, to be managed by their users based on their individual preferences and/or to allow for natural uncertainties and constraints. For instance, nodes may join and leave the PDN society as they decide (by user decision or due to unwanted node/link failure), they control their own resources, and they select their neighbors according to their own administrative policy or physical constraints (e.g., connecting to the nodes that are both accessible and physically close as neighbors). In this sense, individual nodes are self-governed, autonomous, and independent. There is a trade-off between the extent of the social rules and the autonomy of the individual PDN nodes; the more extensive and interfering the social rules, the autonomy of the nodes is more restricted.

Modelling Peer-to-Peer Data Networks. PDNs are distributed query processing systems with an open architecture. The first step toward realizing these systems is to select an appropriate approach to model such systems. As a direct consequence of the computing model described above, a PDN is 1) a self-organizing system, i.e., there is no central entity to organize the PDN and any kind of structural and functional organization emerges from the distributed interaction among PDN nodes; 2) a dynamic system, i.e., the node-set, data-set, and link-set of the PDN are dynamic and in continuous renewal; and 3) a large-scale system, because as an open and beneficial society it tends to attract numerous nodes that intermittently join the society. The combination of these three characteristics makes PDN a "complex system", i.e., a system that is hard to represent/describe information theoretically (considering the large amount of information required to represent the state of the system), and hard to analyze computation theoretically (considering the complexity of computing the state

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 238–243, 2005.

transition of the system). An appropriate modelling approach for such complex PDNs must 1) be compatible with the PDN computing model as a democratic society, and 2) provide a framework with a set of conceptual, experimental, and analytical tools to contemplate, measure, and analyze PDNs; a framework which is neither oversimplified nor overcomplicated to remain both accurate and applicable to such complex systems. We propose the "complex system theory" as the modelling framework for PDNs.

State-of-the-Art. Currently, distributed computing is the framework adopted to model PDNs. With this modelling approach, in line with the traditional system-engineering routine, the system designer implicitly assumes almost full control over the system components and resources. This assumption allows reducing the complexity of the system by imposing fabricated restrictions, and consequently, enables designing efficient mechanisms and architectures. Such an assumption may be valid with typical engineered systems that are managed by a unique authority that governs the entire system. However, it is incompatible with the democratic PDN computing model, where autonomy of the nodes is an essential requirement. Hence, with this modelling approach the resulting solutions are unrealistic and inapplicable for the real PDN applications. Such theoretical solutions that enforce the controlling assumption give rise to dictatorial PDN societies, which are unattractive for prospective citizens, and intolerant and/or fragile to disobedience of their members that want to maintain their autonomy.

The main representative of such solutions is a family of lookup systems, the Distributed Hash Tables (DHTs) [9, 13, 10], which are designed for efficient search in PDNs. DHTs regulate both the data placement and the network topology of the PDN. With the regulated data placement, it is as if the entire data-set of the PDN is owned by a single authority that collects the data from the nodes (the actual owners) and re-distributes the data among them (as a set of slave data storage units/nodes) according to a certain data placement policy to achieve efficient access. Enforcing the data placement violates the autonomy of the PDN nodes in controlling their own data, and for example, is inapplicable to the PDN applications where nodes must maintain their own and only their own data because of security concerns. Moreover, such an unnatural data distribution is an instance of over-engineered design and raises significant practical issues. For example, the communication overhead of transferring the data (or pointers to the data) from the actual owner of the data to where the data is placed can be overwhelming. This important cost factor, which is due whenever the node joins the PDN or its data is updated, is often overlooked in the analysis of the efficiency of the DHTs.

Similarly, with the regulated network topology, among all possible choices of neighborhood, each node is required to connect to a particular pre-defined set of nodes as neighbors. Enforcing the neighborhood of a node violates the autonomy of the node in selecting its neighbors according to its own administrative policy or physical constraints. For example, it is quite possible that none of the designated neighbors for a node are physically accessible to the node when it joins the PDN;

hence, leaving the node isolated. Considering such problems with DHTs, it is not surprising that despite significant efforts of the research community in enhancing and promoting DHTs as the only academic solution for efficient search in PDNs, DHTs are not adopted as practical solutions for any real PDN applications such as file-sharing systems. Instead, these systems have unstructured network topology and prefer to use naive search mechanisms such as flooding, which is not efficient but compatible with the PDN computing model, and hence, practical.

A New Modelling Framework: Complex System Theory. The complex system theory is a unifying meta-theory for collective study of the "complex" systems. Various fields of study, such as sociology, physics, biology, chemistry, etc., were established to study different types of initially simple systems and gradually matured to analyze and describe instances of incrementally more complex systems. The complex system theory is an interdisciplinary field of study which is recently founded based on the observation that analytical and experimental concepts, tools, techniques, and models developed to study an instance of complex system in one field can be adopted, often almost unchanged, to study other complex systems in other fields of study [5]. This meta-theory provides a common modelling framework consisting of a rich set of tools adopted from various fields to study all complex systems under one umbrella.

In this framework, complex systems are modelled as large-scale networks of functionally similar (or peer) nodes, where the links represent some kind of system-specific node-to-node interaction. For example, a social network is a network of people who communicate in a society, a biological network (at the cellular scale) is a network of cells which exchange mass and energy in a biological organ, and a molecular network is a network of molecules that interact by exchanging kinetic and potential energy. Most of the complex systems studied under the complex system theory are natural systems, where nodes are autonomous while they also follow certain natural principles/laws (e.g., the second law of Newton governs kinetic interactions among molecules in a molecular network). Moreover, most of the natural complex systems are also self-organizing, dynamic, and large-scale. Considering the similarity between these features and those of PDNs, we argue that PDNs should also be promoted from the domain of traditional distributed computing systems to the realm of natural complex systems. Consequently, PDNs will be studied alongside their peer systems under the complex system theory, within a modelling framework which is both compatible with PDN's open/autonomous computing model and rich to capture PDN's complexity. With a rich set of tools specially designed to analyze complex systems, the complex system theory is a promising modelling framework for PDNs.

Previously, this modelling approach is successfully applied to the Internet. For example, Ohira et al. [8] used self-organized criticality (i.e., a self-similarity model from the complex system theory [11]) to explain the self-similar scaling behavior of the Internet traffic flows, and Albert et al. [1] employed concepts from statistical mechanics (which was originally developed by physicists to study the collective behavior of the molecular networks, such as temperature and pressure of a mass of gas) to understand the reasons for the power-law connectivity in the Internet topology. However, to the best of our knowledge, modelling PDNs under the complex system theory is novel.

Research Agenda. We categorize PDNs as instances of complex systems and apply the complex system theory as a modelling framework to study PDNs. Our general research agenda is to extend application of the complex system theory to PDNs by:

1. Adopting models and techniques from a number of impressively similar complex systems (e.g., social networks) to design and analyze PDNs; and
2. Exporting the findings from the study of PDNs (which are ”engineered” complex systems, hence, more controllable) to other complex system studies.

We study usefulness of this modelling framework by pursuing two case studies, both focused on the problem of efficient search in PDNs. Observing the similarity between PDNs and social networks, we adopt two models from the study of social networks to develop efficient search mechanisms for two types of PDNs. Search is a generic primitive for query processing in PDNs: a mechanism that locates the required data in response to one or more types of queries is a search mechanism. Developing efficient search mechanisms for the self-organizing, dynamic, and large-scale PDNs is a challenging task. We recognize two different types of PDNs that require significantly different search approaches: unindexable PDNs and indexable PDNs.

Traditionally, index structures are used for efficient search in large-scale distributed object repositories such as distributed databases. By indexing, the repository is organized/structured into a distributed data structure that allows real-time search with minimum cost and short response time. With unindexable PDNs, the extreme dynamism of the PDN node-set, data-set and link-set renders any attempt to self-organize the network to an index-like structure (for efficient query processing) impossible and/or inefficient. Without indexing, efficient search is only possible by efficient scanning of the network nodes. For unindexable PDNs, we introduce the STEPS (Search with Tunable Epidemic Sampling) search mechanism [3] that enables efficient processing of partial selection queries (i.e., selection queries that can be satisfied by a partial result-set rather than the entire result-set). STEPS is inspired by the SIR (Susceptible-Infected-Removed) epidemic disease propagation model for social networks [6]. We also employ the percolation theory [12], a common analytical tool in the complex system theory, to formalize and analyze STEPS.

On the other hand, with the indexable PDNs, the dynamism of the PDN is such that the benefit of indexing the PDN still exceeds the overhead of maintaining/updating the index. For indexable PDNs, we propose a self-organizing mechanism that structures the PDN to SWAM (Small-World Access Method) [4], a search-efficient structure that enables efficient processing of various similarity queries (namely, exact-match, range, and kNN queries). SWAM is a distributed index structure that organizes the PDN nodes in order to index the data content of the nodes while it avoids changing the natural placement of the data. For the design of SWAM as well as its search dynamics, we were inspired by small-world models [15, 2, 7, 14]. Small-worlds are models proposed to explain efficient communication in social networks. These two case studies strongly confirm applicability and appropriateness of the complex system theory as a modelling framework for PDNs.

Acknowledgment

This research has been funded in part by NSF grants EEC-9529152 (IMSC ERC) and IIS-0238560 (PECASE), and unrestricted cash gifts from Microsoft. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

1. R. Albert and A.L. Barabasi. Topology of evolving networks: Local events and universality. *Physical Review Letters*, 85:5234–5237, 2000.
2. L.A.N. Amaral, A. Scala, and M. Brathélémy. Classes of small-world networks. *Proceedings of National Academy of Science of the USA (PNAS)*, 97(21):11149–11152, October 2000.
3. F. Banaei-Kashani and C. Shahabi. Epidemic sampling for search in unstructured peer-to-peer networks. Submitted for review.
4. F. Banaei-Kashani and C. Shahabi. SWAM: A family of access methods for similarity-search in peer-to-peer data networks. In *Proceedings of the Thirteenth Conference on Information and Knowledge Management (CIKM'04)*, pages 304–313, November 2004.
5. Y. Bar-Yam. *Dynamics of Complex Systems.* Westview Press, 1997.
6. H. Hethcote. The mathematics of infectious diseases. *SIAM Review*, 42(4):599–653, Otober 2000.
7. J. Kleinberg. The small-world phenomenon: an algorithmic perspective. In *Proceedings of the 32nd ACM Symposium on Theory of Computing (STOC'00)*, pages 163–170, May 2000.
8. Toru Ohira and Ryusuke Sawatari. Phase transition in a computer network traffic model. *Physical Review E*, 58:193–195, 1998.
9. S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content-addressable network. In *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM '01)*, pages 161–172, August 2001.
10. A. Rowstron and P. Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *Proceedings of ACM International Conference on Distributed Systems Platforms (Middleware'01)*, pages 329–350, November 2001.
11. D. Sornette. *Critical phenomena in natural sciences: chaos, fractals, self-organization and disorder.* Springer, 2000.
12. D. Stauffer and A. Aharony. *Introduction to Percolation Theory.* Taylor and Francis, second edition, 1992.

13. I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM '01)*, pages 149–160, August 2001.
14. D.J. Watts, P.S. Dodds, and M.E.J. Newman. Identity and search in social networks. *Science*, 296:1302–1305, 2002.
15. D.J. Watts and S.H. Strogatz. Collective dynamics of small world networks. *Nature*, 393(6684):440–442, 1998.

Attribute-Based Access to Distributed Data over P2P Networks

Divyakant Agrawal, Amr El Abbadi, and Subhash Suri

Dept. of Computer Science, UCSB, Santa Barbara, CA 93106, USA
{agrawal, amr, suri}@cs.ucsb.edu
http://www.cs.ucsb.edu/~agrawal
http://www.cs.ucsb.edu/~amr, http://www.cs.ucsb.edu/~suri

Abstract. Peer-to-peer (P2P) networks are distributed data sharing systems with no dedicated and centralized infrastructure. These systems are attractive because they deliver on the Internet's promise of true decentralization, offering scalability, availability, fault tolerance, robustness, and low barriers to entry. While P2P systems have been used so far mainly for file sharing, their true potential lies as a vast, loosely connected world wide infrastructure for sharing resources, data and information. However, many challenging research problems must be addressed and solved before this vision can materialize. This paper addresses a natural step in the evolution of P2P: going beyond simple file sharing based on exact-name based lookups to data and information sharing where data is accessed based on its attributes or properties. We have identified diverse applications such as network monitoring, astronomy data applications, event-notification systems, and Grid computing that can benefit directly from attribute-based access to distributed data over P2P systems. Based on the application requirements, we propose three new models for both data distribution and data accesses. For each of these models, we propose CAN and CHORD like structures for data storage and information retrieval. A novel aspect of our development is that one of the models identified is directly applicable for building *content-based publish/subscribe* systems over P2P networks.

1 Introduction

Peer to peer (P2P) computing has attracted enormous interest recently, both from the commercial and the academic communities. The underlying principle of P2P systems is very simple. A user wishing to participate in a P2P system registers his/her machine and, once registered, becomes a peer node. When a user wants to search for a file, he submits a query string (name of a file), and the system returns the name of one or more peers that contain the file (if it is available in the system). Napster became an overnight sensation as millions of users found it useful to share their music files. Its centralized index, however, was technically deficient and not designed to scale to the large population that it found itself serving. Soon thereafter, other more decentralized file-sharing

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 244–263, 2005.

systems like Gnutella and Freenet [7] came along that eliminated the need for a centralized index. The popularity of P2P systems has also resulted in several research projects [20, 9, 21, 8, 10, 13, 1, 26, 24, 23, 27] addressing issues such as scalability, fault-tolerance, and security.

In their current form, however, P2P systems are still primarily used for sharing files (or media objects). Yet they possess the potential to become much more than file sharing systems. A grand vision of P2P computing is to combine all the informational resources of the world wide web (data, storage, computing power) into a loosely connected but highly available, reliable, robust system [11]. We envision P2P systems to evolve into something far more significant than a simple file sharing infrastructure. In particular, we identify a natural step in this progressive evolution: moving from accessing files by name to accessing data based on its attributes or properties [15, 17, 16]. In order to motivate this vision better, let us first consider some examples:

1. Internet routers maintain extensive packet logs of network traffic, which are used or monitor network conditions, to detect traffic anomalies, or plan network capacity. A telecommunication network manager may want to query this data: *how much TCP data went across a backbone router R that was sent from a subnet S to subnet D between* 1 *and* 4 *PM yesterday?*
2. The World Wide Telescope is an ambitious project to link several large telescopes and make their data repositories, such as the Sloan Digital Sky Survey (SDSS) [28], available to the global community of scientists and scholars. An astrophysicist may wish to query: *find all objects with brightness* $\geq \beta$ *in the color spectrum range* γ *and positional range (right ascension, declination) R.*
3. One of the key data processing challenges facing the homeland security is to link together the multitude of databases at various airports and install continuous queries like: *alert FBI whenever a new arrival fits one of the given profiles.*
4. The grid computing [12] initiative aims to harness a large number of loosely coupled computers (or storage devices) into a single "transparent" supercomputer. This distributed supercomputer requires a "directory service" that lets users discover what resources are available. For instance, a user may want to know: *which computers in a certain IP address space have at least 2 GHz processor, at least 1 GB memory, and run Linux version X or later.*

An important theme common to all these examples is that data is distributed and is often too large to reside in a single place. For example, the Sloan Digital Sky database alone contains more than 200 million objects, and each year 5 TB of new raw data is collected. Similarly, Internet routers maintain logs for millions of packets per second. Instead, the P2P paradigm can be used to access data and execute queries obviating the need to consolidate the information at a central repository. In addition, in nearly all of these applications, users also form a natural community of common interest, making P2P architectures quite attractive for managing data. The main observation that can be made from the above examples is that in all cases user queries need to be executed over data that is potentially distributed over multiple repositories. A P2P paradigm can be used here for accessing and executing such queries obviating the need for consolidating the data and information at a central repository. Unfortunately, current developments in P2P computing, in general, have focussed on providing name-based access to distributed objects. As can be seen from the above examples, in order to retrieve information relevant to these queries, we need to extend the P2P paradigm so that data can be accessed based on its attributes or properties.

Even within the four applications mentioned, there are subtle but significant differences in the nature of data and queries. Another observation that can be made from the above set of examples is that the nature in which these queries access distributed data aries significantly from each other. For instance, in examples 1 and 2, a large amounts of *raw data* are collected, and a wide variety of users ask queries. Such data is likely to be stored for a long time (astrophysics data) or at least days or weeks (network traffic). Example 3, on the other hand, deals with a model where users may pre-install queries (FBI profiles) and new data generates notification events. Thus, queries may be posed *even before data arrive.* Example 4 may be a mixture of these two models: a user may ask to be notified when a certain combination of resources become available, or ask queries about the current availability of resources. One can envision many other potential applications requiring this form of query functionality where data and objects are accessed based on their attributes instead of being accessed by name. For example, applications involving sensor networks where data is constantly generated by geographically dispersed sensors can benefit from attribute-based access to data over P2P systems [18]. Another class of applications is specialized data analysis where data and derived data products are generated in distributed manner. Examples of such applications include weather and climate pattern analysis, demographic information analysis, and geological data pattern analysis. In summary, there is a large number of applications that can benefit from attribute-based data access over P2P systems.

In this paper, we envision P2P data sharing architectures that can be deployed for attribute-based access to distributed data. We start by identifying four different models of data distribution and data access in distributed systems in Section 2. The first model under this classification corresponds to the current exact name-based lookup functionality of P2P systems. In the remaining sections we outline research challenges in the context of the remaining three models identified in Section 2. The paper concludes with a research vision in the context of these models.

2 Data Distribution and Attribute-Based Access Models

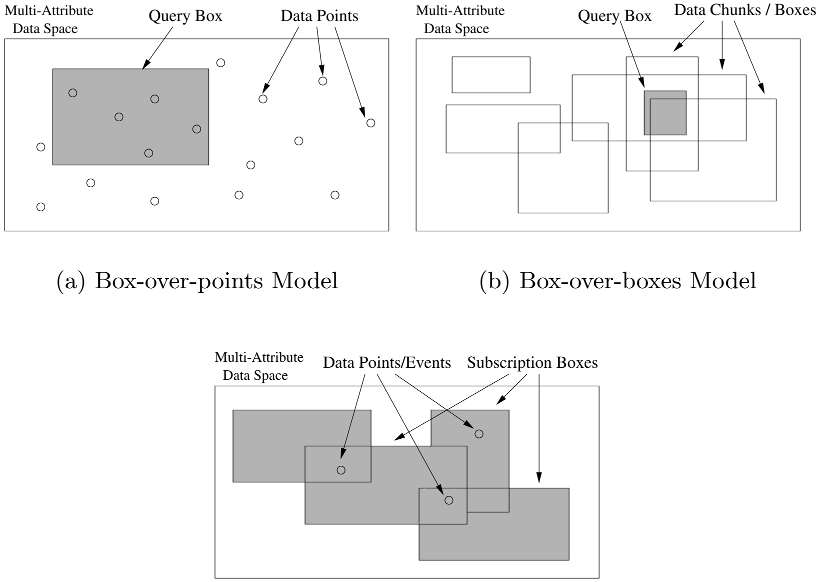
The P2P storage model is a compelling one for the applications we have suggested. However, as mentioned earlier, data accesses in current P2P systems are largely limited to “name–based exact match” queries, which is inadequate for the kind of applications identified in the introduction. There remains a large gap between the name-based exact lookup functionality of current P2P systems and the complex attribute-based access functionality of even the most primitive database systems. In this paper, we design and develop P2P architectures in which users can access data based on its properties or *attributes*. A key step in this direction is to use the *relational model* in which data is retrieved from storage systems based on *selection predicates*. Typically, these selection predicates are formulated either as *point queries*, where the exact value of attribute(s) is specified, or as *range queries*, where the attribute value ranges are specified. In general, selection predicates are used by the database engine in conjunction with index and access structures to efficiently retrieve relevant data. Even in the context of complex database operations involving SQL queries, selection operations are the most fundamental operations. Our goal, therefore, is to develop P2P architectures and access mechanisms that will enable efficient execution of selection queries over distributed data.

There are four natural models for access to distributed data. The first one, which we call *point-on-point*, is the basic exact match query: the access is at the level of individual objects or files. The user specifies all necessary attributes of the desired object, and the system determines which, if any, peer contains that object. This is equivalent to the name-based "point" lookup, which is currently available in P2P.

The second model, which we call *box-over-points*, is the basic range query model (Figure 1(a)). A user's query is a range, which can be thought of as a multi-dimensional *box*, and the system returns all the objects (points) that match this query. This model assumes that the objects of interests are scattered over multiple peers. While such a fine-grained distribution of data may not be attractive or feasible for all applications, there are some cases where this model is a natural fit. Sensor networks, for instance, offer a good example: each sensor acts like the source of a single data value, and typical user queries are ranges, for instance, asking for "average temperature within a geographical region," "all locations where temperature exceeds a certain value" etc.

Next, we consider the *box-over-boxes* access model, where we assume that *subsets* of the objects are distributed among the peers, and users submit range queries (Figure 1(b)). This model is motivated from the observation that points based object distribution is too fine-grained and therefore is applicable to few specific applications such as sensor data networks. In general, we envision that a large number of application will group objects together based on certain properties and hence the need for box based object distribution. Each peer stores a set of boxes, representing data in certain ranges, and given a query box, the goal is to retrieve enough stored boxes to be able to answer the query. The data stored at peers could be either carefully chosen "chunks" of the source data, or cached copies of answers to previously asked range queries. In either case, the scattering of data is at a much higher level than individual tuples. This model significantly generalizes the box-over-points model, and it is one of the main focus points of our work.

Finally, we consider the *point-over-boxes* model, where peers store boxes, representing ranges, and the query processing involves finding all boxes that



(a) Box-over-points Model

(b) Box-over-boxes Model

(c) Point-over-boxes Model

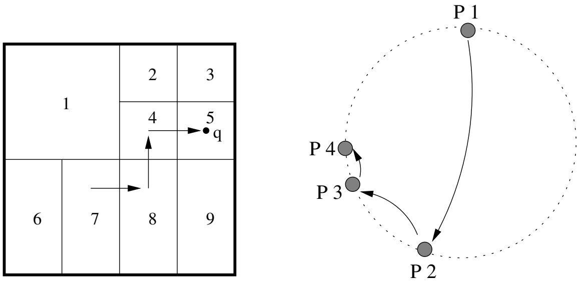
Fig. 1. Data Distribution and Attribute-based Access Models

match a point (Figure 1(c)). This model is motivated by the emerging "publish/subscribe" model of data service, where interest profiles of users are modeled as ranges (boxes) in an attribute space, and when an event (point) occurs, the goal is to quickly identify all ranges that contain the point.

3 Box-over-Points Distribution and Access Model

One of the main challenges in a P2P system is providing efficient, fault-tolerant, and scalable lookup operations. Early P2P systems either used a centralized [22] or a flooding approach [14]. Although widely used, these solutions suffer from the scalability problem. Either the centralized site is overloaded, or the system is flooded with messages. Several academic efforts are underway to design structured P2P systems to ensure scalable performance as the number of peers and the number of data objects increase. These systems typically use Distributed Hash Tables (DHT) [3] to uniformly distribute the location information about the objects among all the peers. Furthermore, they typically ensure that lookup operations locate the desired objects in a bounded number of hops.

Two of the early and most popular structured P2P systems are CAN [23] and CHORD [27]. We will use these two systems as representative examples for structured P2P systems and will illustrate our extensibility models within these frameworks. In these systems, both the peer identifiers (i.e., IP address) and the



(a) CAN Routing (b) CHORD Routing

Fig. 2. Routing Schemes in Structured P2P Systems

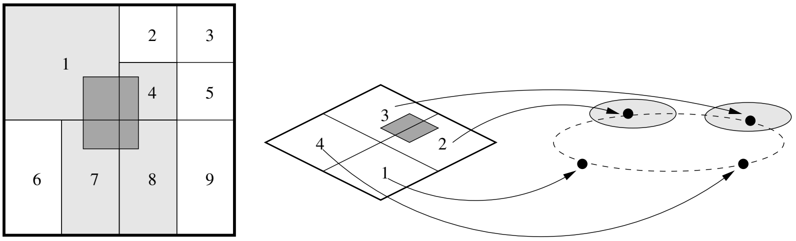
object identifiers (e.g., song name) are uniformly hashed into the same identifier address space. Typically, the hash function is LSA-1, which uniformly hashes the identifier to a 128 bit space.

In CAN [23], the identifier address space is a d-dimensional space. This space is partitioned into *zones*, each *owned* by a peer. Initially, the entire space is a single zone owned by some initial peer. As nodes join the system, they are uniformly hashed to a point in the multi-dimensional space. The system then splits the zone in which this point is located with the current owner. Likewise, the name or identifier of each new object is hashed into the multidimensional CAN space and the owner of the zone with this point either stores the new object or, more realistically, stores the IP address of the peer storing the object. Lookup simply involves hashing the query identifier to a point in the CAN space and locating the owner of the corresponding zone, who is responsible for returning the requested data, or a pointer to the peer where the requested data is stored as shown in Figure 2(a).

In CHORD [27], the identifier space is a ring, with 2^{128} locations. When a peer joins the system, it is hashed to one of these locations. A new object identifier is also hashed to this circular ring, and the file, or a pointer to the file is stored at the closest peer following the hash point. Lookup simply involves hashing the query identifier to a point in the CHORD ring, and checking the peer following this point (Figure 2(b)).

Since the distribution and access model for current P2P systems fall under the point-on-point model, both CAN and CHORD only support exact match queries, e.g., find the album entitled "Blood on the Tracks". However, we are interested in more complex *box-over-points* queries, e.g., find all songs by Bob Dylan from 1975–1979. Both of these systems (as described above) would face scalability problems in that they would entail breaking a single range query to an enumeration of multiple point queries. We, therefore, now propose simple and preliminary extensions for adapting these systems to support box-over-points queries.

Although the original CAN proposal associated no semantics with various dimensions, it is relatively straightforward to associate each dimension with one of the attributes or properties of the domain of discourse. For example, one dimen-



(a) Range Query over CAN

(b) Range Query over CHORD

Fig. 3. Adapting Structured P2P Systems for Range Query Processing

sion corresponds to names (alphabetically ordered) and another corresponds to dates of publication (numerically ordered). Consider an application where each object is associated with *d* attributes. We create a *d*-dimensional CAN structure. Each new object is mapped to a point in the space corresponding to the values associated with its d attributes (we assume every point has all attribute values completely specified). The rest of the CAN structure, and methods for peer joins and departures do not change. Now consider a box query with a range in each of the d dimensions (if some dimension is not specified, we assume the range to be the entire domain). We need to locate all zones that overlap with this box since they may contain some data objects of interest to the query. One simple method to implement this would involve locating the zone corresponding to the bottom left corner (in d dimensions) of the query box (see Figure 3(a)). The peer owning this zone identifies all objects that were hashed to points in the intersection of the zone and the query box. These form part of the answer set which is returned to the client. However, the peer needs to also propagate the box query to all zones above it and to the right (in the multidimensional space). The query box is thus recursively propagated in this way until it reaches zones which do not overlap with the query.

Alternatively, we could consider CHORD as the underlying P2P structure to support box-over-points queries. In this case we need to map the d-dimensional attribute space to the one dimensional CHORD ring. One simple approach would involve dividing the d-dimensional attribute space into *partitions* (see Figure 3(b)). These partitions may be predefined based on the domain range or based on the actual data distribution. Each partition is given a unique identifier; this could be arbitrarily chosen or could be based on, say, the lower left coordinate of the partition. This identifier is used by CHORD to hash the partition to a single peer in the CHORD ring. This peer is now responsible for all the data points which are located in that partition. When a box query needs to be evaluated, the querying peer uses CHORD to locate the peers that own all partitions which overlap with the box query. These peers collectively either store or have pointers to the locations of all data points in the query box.

Although simple and in some way naive, the box-over-points model has many applications. In fact in sensor networks, alternative solutions have been proposed [19]. One main challenge that arises in the context of real applications is the fact that data as well as accesses to the data tend to have non-uniform distribution or equivalently are skewed. This can potentially lead to some peers being responsible for most of the data. Analogously, due to skewed accesses, some peers will have a higher burden for processing queries.

4 Box-over-Boxes Distribution and Access Model

We now consider a data distribution and attribute-based data access model in which data is appropriately “chunked” or partitioned and is stored over multiple peers. The reason such “chunking” or partitioning may arise is either due to the physical organization of the system or perhaps as a result of *caching* prior range queries at the peers. For example, in the context of sensor networks, the “chunking” may arise as a consequence of fine-grained readings from individual sensors being collected at a base station responsible for a group of sensors. Similarly, in the case of astronomy data, the partitioning may result from different repositories being responsible for different parts of the sky. We refer to such partitions in a d-dimensional space as d-boxes. Hence the main problem under the box-over-boxes model is that whenever a new query is issued, the peers are searched to determine if the query can be answered from the data boxes stored at the peers. Our goal is to develop techniques that will enable efficient evaluation of range queries over d-boxes that are distributed (and perhaps replicated) over the peers in a peer-to-peer system. The problem of performing range queries in a peer-to-peer system has recently been investigated [2, 16]. Andrzejak and Xu [2] use Hilbert curves to partition data among peers in a way that contiguously distributes the data at the peers. Gupta et al. [16], on the other hand, use locality sensitive hashing to retrieve d-boxes ”similar” to a query box. Both these approaches, however, are fairly restrictive for supporting box-over-boxes model in full generality.

In order to adhere to the peer-to-peer design methodology, the proposed solution for range lookup should also be based on distributed hashing. A nice property of the DHT-based approach is that the only knowledge that peers need is the function that is used for *hashing*. Once this function is known to a peer, given a lookup request the peer needs to compute the hash value locally and uses it to route the request to a peer that is likely to contain the answer. Given this design goal, a naive approach would be to use a linear hash function over the range query schema $\langle low, high\rangle$, i.e., a linear hash function over *low*, *high*, or both *low* and *high*. A simple analysis reveals that such a hash function will enable *only* the exact matches of given range requests, but not set containment matches. As an example, suppose the range $\langle 20, 35\rangle$ for a given attribute is stored at a peer. While the query $\langle 25, 30\rangle$ does not have an exact match with the stored range, it can be answered using the stored range: we can extract the objects relevant to the query from the superset range $\langle 20, 35\rangle$. The set containment query functionality is powerful, but harder to provide. We now propose two designs design to solve the range containment problem. The first one utilizes a CAN like structure over the peers and the second one uses the CHORD structure.

4.1 CAN-Based Design

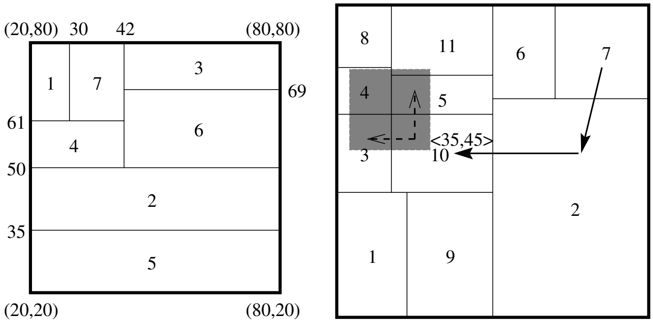
Consider a set of data objects each specified with d attributes. The main challenge is to map d-boxes representing a subset of these objects into a CAN-like address space. Rather than associating each dimension of CAN with a single attribute as discussed in Section 3, we associate two dimensions for each attribute. The first dimension captures the start value of the range specified for the attribute, while the second dimension captures the end value of that range. Hence, we use a $2 \cdot d$-dimensional CAN and refer to it as 2CAN. A box, hence, is mapped as a single point in this space. Given the domain $[a, b]$ of an attribute, the corresponding virtual hash space is a two dimensional square bounded by the coordinates (a, a), (b, a), (b, b), and (a, b) in the Cartesian coordinate space. Figure 4(a) shows the corresponding virtual hash space for a range attribute whose domain is [20,80]. The corners of the virtual space are $(20, 20)$, $(80, 20)$, $(80, 80)$, and $(20, 80)$. A range $\langle q_s, q_e\rangle$ is hashed to point (q_s, q_e) in the virtual hash space.

The virtual hash space is partitioned into zones as was described in Section 3. In this case, a zone can be identified by a pair $\langle(x_1, y_1), (x_2, y_2)\rangle$ where (x_1, y_1) is the bottom left corner coordinates whereas (x_2, y_2) is the top right corner coordinates. Figure 4(a) shows the partitioning of the virtual space. The virtual space is partitioned into 7 zones : *zone-1* $\langle(20, 61), (30, 80)\rangle$, *zone-2* $\langle(20, 35), (80, 50)\rangle$, *zone-3* $\langle(42, 69), (80, 80)\rangle$, *zone-4* $\langle(20, 50), (42, 61)\rangle$, *zone-5* $\langle(20, 20), (80, 35)\rangle$, *zone-6* $\langle(42, 50), (80, 69)\rangle$, and *zone-7* $\langle(30, 61), (42, 80)\rangle$. Each zone is assigned to a peer in the system.

For the purpose of routing requests in the system, each peer maintains a *routing table* with the IP addresses and zone coordinates of its neighbors, which are the owners of adjacent zones in the virtual hash space. For example in Figure 4(a), the routing table of the owner of *zone-4* contains information about its four neighbors: *zone-1, zone-7, zone-6* and *zone-2.*

In the following, for simplicity of exposition, we assume that d-boxes are generated as a result of caching answers to prior range queries. Alternatively, d-boxes stored at different zones may be generated via a separate mechanism determined by the underlying application. The point to which a query is mapped to is referred to as the *target point* of the query range. The target point is used to determine where to store the information about the answer of a range query as well as where to initiate range lookups when searching for the result of a range query. The zone in which the target point lies and the node that owns this zone are called the *target zone* and the *target node*, respectively. Therefore, the information about the answer of each range query is stored at the target node of this range.

Initially, the peers in the system may not be populated and therefore a range query must retrieve the results from some initial source of information, e.g., the SDSS repositories. In general, the answer is the d-box. Once a peer node



(a) Mapping boxes to Points

(b) Routing of Queries

Fig. 4. Range Query Processing using CAN as the underlying P2P structure

retrieves the answer for its range query, if the peer is willing to share its computed answer and has available storage space, it caches the d-box and informs the corresponding target node about it. The target node stores a pointer to this querying node. If the target node has available storage, it caches the result itself. In either case, we say that the target node stores the result of this query. For example, according to Figure 4(a), the range query $\langle 50, 60\rangle$ is hashed into *zone-6*, so the set of objects that form the answer to this query may be stored at the node that owns *zone-6* or the node will store a pointer to the peer that caches the objects in that range.

When searching for the answer of a range query, the first place to look for cached results is the target zone of this range. Therefore whenever a range query is issued, it is routed toward its target zone through the virtual space. Starting from the requesting zone, each zone passes the query to an adjacent zone until it reaches its target zone. The target node is checked to determine if it stores a d-box that contains the results of the range query. If so, the result is returned to the client. However, if no such d-box is found the search is forwarded to all zones that may contain a superset of the answer set. Given the simple geometric properties of 2CAN, such zones are restricted to the upper-left region of the target zone. This is easy to observe since such zones contain d-boxes with earlier starting values and later ending values than the target zone. Figure 4(b) shows how a query is routed in the system. The range query $\langle 35, 45\rangle$ is initiated at *zone-7* and then routed through *zone-6* to its target zone, *zone-10*. If *zone-10* does not contain any d-box $\langle X, Y\rangle$ such that $X \leq 35$ and $45 \leq Y$ then the search is forwarded to the peers that correspond to zones to the upper left side of zone-10 since if the answer exists it must be restricted in that part of the attribute space. The search region is illustrated as the shaded region in Figure 4(b).

An important challenge in the design of 2CAN is to limit the length of the routing path and to reduce the space and time overheads of retrieving a qualifying d-box. An estimation of the average routing distance for processing range queries in the proposed model is presented in [25]. The analysis shows that the average routing path length in an equally partitioned hash space is $O(\sqrt{n})$, where n is the number of zones in the system. The above analysis easily generalizes to $2d$-dimensional space representation. In addition, we have developed techniques for zone maintenance for a dynamic environment where peers are allowed to join and depart from the system freely and frequently. As discussed above, 2CAN results in directed flooding if a query cannot be processed at its target zone. Although this flooding is controlled, a naive approach can give rise to many duplicate messages. We have developed techniques to minimize the duplicate messages when query requests are forwarded. Another aspect of 2CAN is that the upper-left corner of the attribute space may need to bear a greater burden of answering queries. We have also developed load-balancing strategies based on *zone replication* (instead of zone partitioning) to distribute this load.

4.2 CHORD-Based Design

Our second approach to support box-over-boxes model leverages the $O(\log n)$ search complexity of CHORD and has two components: a *spatial structure* that organizes the underlying multidimensional attribute space, and a *mapping rule* that associates each d-box with a unique region of the spatial structure. Let the d-boxes be in a multidimensional hypercube $\mathcal{D}$ of side length L, called the *domain*[1]. We could use the simple d-dimensional partitioning approach used in Section 3 for point data. However, with d-boxes the mapping is not as straightforward as with point data. Furthermore, since d-boxes have different extents, a uniform partitioning may not be efficient. We therefore propose a hierarchical partitioning. We subdivide $\mathcal{D}$ into 2^d identical boxes, which are the *regions* at *level 1*. Each of these 2^d boxes is then recursively subdivided, until we reach *unit size boxes*.[2] It is easy to see that all the regions are also d-dimensional hypercubes.

The recursive partition of $\mathcal{D}$ is naturally modeled by a tree, called the *partition tree*, where each node corresponds to a region produced during the partition. The root (level zero) corresponds to the entire domain $\mathcal{D}$. The 2^d children of the root correspond to 2^d boxes of level 1, and so on. Figure 5 shows an example of the spatial partition and its corresponding partition tree. The partition tree defines a parent-child relationship among different regions. With each node, we associate an *id* between 0 and $\eta - 1$, where η is total number of regions. The root node has *id* 0. If a node's *id* is i, then the *id*s of its children are $i2^d + 1, i2^d + 2, \cdots, (i+1)2^d$. Call two regions r and r' *neighbors* if they are at the same level of the partition tree and are *spatially adjacent*; that is, the two

[1] We choose a hypercube for convenience; with a careful modification, the scheme extends also to rectangular domains.

[2] The unit of space resolution is a user defined parameter. For simplicity, we assume that the original domain is a box of integer dimension L. Thus, the recursive partitioning stops after $\log_2 L$ levels.

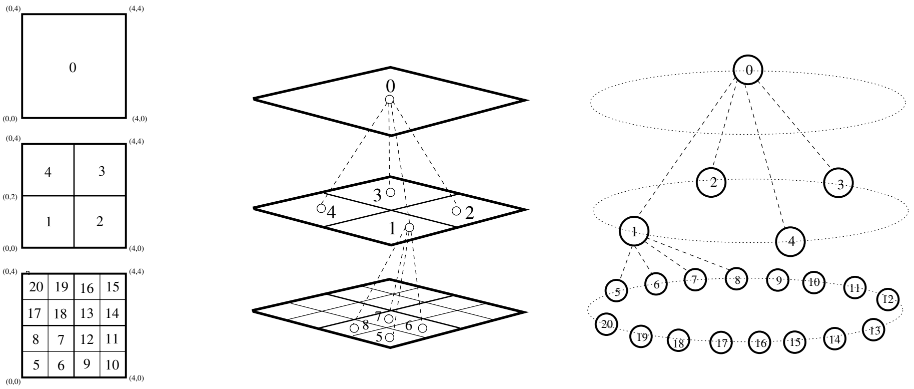


Fig. 5. CONE's Overlay Structure

regions have at least one point in common. In Figure 5, for instance, regions 5 and 7 are neighbors but 5 and 13 are not.

The main challenge now is to map each d-box to the different regions of the partition tree. This mapping depends on two attributes of a d-box: a *reference point*, and a *measure*. The reference point can be any consistently defined point in the box. In our scheme, we use the *lower left* corner. More precisely, the reference point of a d-box $[x_1, x_1'] \times [x_2, x_2'] \times \cdots \times [x_d, x_d']$, is the corner $(x_1, x_2, \cdots, x_d)$. In Figure 5, for example, the reference point of region 5 is the point $(0, 0)$. Note that a reference point can be mapped to a region at each of the different levels in the partition tree, e.g., $(0, 0)$ in the regions 5, 1, and 0. In order to determine which region a d-box B is mapped to, we define the *reference family* of B, denoted $\mathcal{F}(B)$, to be the set of regions in the partition tree that contain the reference point of B. It is easy to see that $\mathcal{F}(B)$ contains exactly one region from each level of the partition tree and that the region at level i is the parent of the region at level $i + 1$. Thus, the nodes corresponding to regions of $\mathcal{F}(B)$ lie on a single path in the partition tree. This scheme will store B at one of the regions of $\mathcal{F}(B)$, but that choice depends on the second attribute: the measure of B.

The *measure* of B, denoted $meas(B)$, captures the *size* of B. There are many natural ways to define the measure: the volume, the longest dimension, the shortest dimension, the average dimension, etc. In our scheme, we use *the longest dimension* as the measure. That is, $meas(B) = \max_{1 \leq i \leq d} |x_i - x_i'|$. Our *mapping rule* is as follows: A d-box B is mapped (uniquely) to the highest region of its reference family whose level is at least $\log L - \log(meas(B))$. For convenience, we use the term *native region* to denote the region to which a box is mapped. Thus, if $meas(B)$ is in the (semi-open) range $(2^{m-i-1}, 2^{m-i}]$, where $m = \log L$, then the native region of B is the region in $\mathcal{F}(B)$ at level i. The key property of our mapping rule is that all boxes matching a query Q can be found by searching only the regions that are neighbors of Q's reference family. Since $\mathcal{F}(Q)$ has at most $\log L$ regions, and each region has a constant number of neighbors (2^d in d-space), our lookup scheme will be able to answer the query by searching $O(\log L)$ identifiers. We would like to point out that several other natural definitions of the measure do not lead to efficient search. For instance, mapping rules based on the *volume* or the *shortest dimension* can scatter boxes in irregular ways, making it impossible to find boxes matching a query in a small number of regions.

We now describe a generalization of CHORD, which we call CONE, based on the generic scheme described so far. CONE uses the spatial partition of the previous section to build an efficient overlay structure. It generalizes the CHORD structure developed for box-over-points model in a natural way. CONE groups the d-boxes by their measure: boxes of small measure are associated with regions near the bottom, while boxes of large measure go to regions near the top of the partition tree. CONE is organized in levels too—the name CONE is suggestive of the *conical* shape of the structure, where each level looks like a ring, with rings getting smaller near the top. CONE consists of $O(\log L)$ CHORD rings. The number of levels or rings is a tunable parameter. The ith ring manages the set of regions at level i in the partition tree. For convenience, we use the *id*s of the regions also as the *id*s in the *identifier space* of CHORD. Thus, the *id*s of the regions at level i are the *identifier space* for the ith ring. See Figure 5 for illustration. In CONE's overlay structure, two identifiers are said to have a parent-child relationship if the corresponding regions have one in the partition tree. Similarly, two identifiers are neighbors if the corresponding regions are neighbors.

A peer in CONE belongs to exactly one chord ring. Each peer has an *id* between 0 and $\eta - 1$. The identifier corresponding to the peer's *id* is referred as its *master id*. A peer *owns* all the identifiers whose *id*s lie between its *id* and its predecessor's *id*. The predecessor of p is the closest peer in its ring whose *id* is smaller than p's *id*. A peer's *zone* is the set of identifiers owned by that peer. In Figure 5, for instance, if peers 15 and 19 are the neighboring peers, then the zone of peer 19 includes the identifiers 15–18.

We use the parent-child relationship among the identifiers of the partition tree to define the parent-child relationship among the peers. Specifically, a peer p' is called the parent of p if p' owns the identifier that is the parent of p's master id. We refer to p as a child of p'. Clearly, each peer has only one parent peer, but multiple children peers. We call two peers neighbors if they own identifiers that are neighbors.

The distribution of peers across rings is tunable. We use the default rule where peers are assigned to rings in proportion to the ring size. One way to implement this rule is to let a new joining peer choose its *id* uniformly between 0 and $\eta - 1$, which ensures that it is assigned to the ith ring with probability $2^{-d(m-i)}$, where $m = \log L$ is the height of the partition tree. CONE extends the basic pointer structure of CHORD. Each peer maintains two additional types of pointers: *level pointers* and *spatial neighbor* pointers. A peer p (at level i) maintains two level pointers, an *up pointer*, which references the parent of p (at level $i - 1$), and a *down pointer*, which references a child of p (at level $i + 1$). The level pointers help the search to navigate across levels in constant time, and they are critical in reducing the search complexity from $O(\log n \log L)$ to $O(\log n + \log L)$. The second type of pointers reference p's spatial neighbors (as defined by our spatial structure). These pointers are needed to efficiently search the spatial neighbors of each peer visited during the search.

A query lookup in CONE is done in two phases. We first determine the native region of the query box Q (from its reference point and measure), and then find the peer p who owns this native region. In order to locate p, we use the up/down pointers to reach the correct ring, and then use CHORD to perform the lookup using the native region's *id*. Having found p, the second phase then recursively searches the neighbors of Q's reference family at higher rings. The first phase of the search takes $O(\log n)$ steps (dominated by the CHORD search), and the second phase climbs $O(\log L)$ levels of CONE, but each level takes only a constant time; i.e. the up/down pointers allow us to avoid doing a new CHORD lookup search at each level. Thus, the overall search time to locate a matching box is $O(\log n + \log L)$.

The basic design of CONE gives us a sound starting point for an attribute-based distributed data access, but it can be improved and extended in several directions. A natural way to improve the "hit rate" of a range cache system is to locate multiple boxes whose union covers the query. Such a scheme can also improve the "quality" of the answer, because one could find boxes whose union covers the query "more tightly" than a single box, hence minimizing the extraneous data. Since locating each additional box also incurs a search cost, we need to balance the total search against the quality of the answer. Both CAN and CHORD use hash functions to "uniformly" distribute the data (as well as the peers) in the search space. However, in many applications, data are highly skewed, and the uniform hashing is likely to create hot spots. In such a case, a more "data adaptive" hashing is needed, and we plan to investigate such hash functions and partitions. In CONE, for instance, we can adapt the assignment of peers to different chord rings to match the load. Analogously, load-balancing strategies need to be developed to handle the case of skewed accesses, which leads to some peers sharing a greater burden of load to process queries.

5 Point-over-Boxes Distribution and Access Model

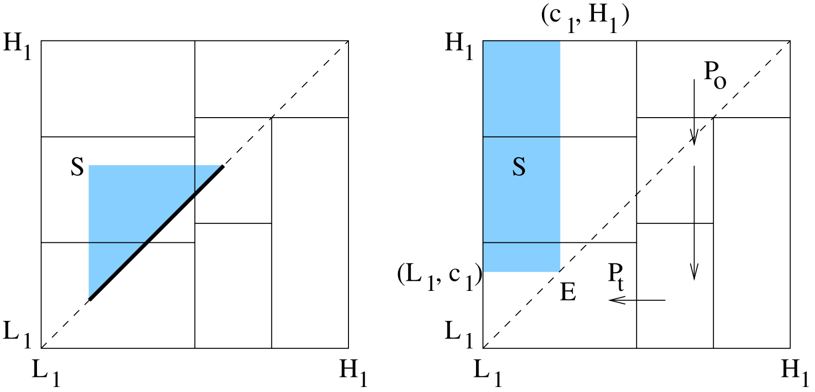
A large class of data access over P2P systems is likely to be in the context of *future data*. Such systems are often referred to as *publish/subscribe* systems. Publish/Subscribe systems are utilized to deliver data events from *publishers* (data/event producers) to *subscribers* (data /event consumers) in a decoupled fashion. Publishers can be completely unaware of the subscribers and simply introduce data events into the system. Subscribers can register their interests with the system in the form of *subscriptions* which act as filters that are used by the system to deliver relevant events to the subscribers. The publish/subscribe system is required to manage the subscriptions and on the occurrence of an event find the matching subscriptions and deliver the event to relevant subscribers.

Existing solutions [5, 4, 6, 29] for publish/subscribe systems employ routing agents for the dissemination of events to the subscribers. Typically, these agents are distributed across the network based on the network locations of the subscribers. These agents form an overlay routing tree which is used to disseminate the events. The agents install filters based on the subscriptions on the overlay links to only send relevant events across. This solution has scalability problems because of the routing bottlenecks at the roots of the routing trees, as well as it is not easy to add more agents to the routing trees.

We employ a peer-to-peer solution for the publish/subscribe system. We utilize the Distributed Hash Table based P2P overlay network for storage of subscriptions and event delivery. This system can be employed directly over the routing agents to improve their scalability. Subscribers can themselves contribute to this system by introducing their machines as peers into the system. A peer-to-peer design for the publish/subscribe system makes it more scalable by utilizing the distributed resources of the agents as well as contributing client machines. In addition, the peer-to-peer design provides the flexibility of adding resources to the system dynamically to scale the system as the demand on the system increases.

We consider a content-based publish/subscribe system with multiple attributes. The schema for the system can be described in the following way: $\mathbb{S} = \{A_1, A_2, \ldots, A_d\}$, where each A_i corresponds to an attribute. Each attribute has a name, type and domain, and can be described by the tuple {`Name: Type, Min, Max`} where `Min` and `Max` is the domain range of the given attribute. The schema for the publish/subscribe system is known to all the peers participating in the system. A *subscription* is a conjunction of predicates over one or more attributes. Each predicate specifies a constant value (using =) or a range (using $<, >, \leq, \geq$) for an attribute. An attribute cannot appear in more than one predicate. The subscription should specify a continuous range over an attribute A_i. An example subscription is $\mathcal{S} = (A_1 \geq v_1) \wedge (v_2 \leq A_3 \leq v_3)$. An *event* is a set of equalities over all the attributes in the schema. Therefore, an event can be represented as $\mathcal{E} = \{A_1 = c_1, A_2 = c_2, \ldots, A_d = c_d\}$. Using this nomenclature a subscription can be visualized as a *box* whereas an event corresponds to a *point*. An event $\mathcal{E}$ matches a subscription $\mathcal{S}$ if each predicate of $\mathcal{S}$ is satisfied by the value of the corresponding attribute specified by the event $\mathcal{E}$. The publish/subscribe system is required to store the subscriptions specified by the users and given an event, find all subscriptions matching the event and deliver the event to the subscribers. Thus the name *point-over-boxes* model.

We now describe the construction of the logical space used for maintaining the subscription boxes and subsequent routing of event points. We build on the 2CAN P2P system developed for box-over-boxes model in Section 4.1. Given a schema $\mathbb{S}$ with d attributes $\{A_1, A_2, \ldots, A_d\}$, we create a Cartesian space with $2d$ dimensions. Attribute A_i with domain range $[L_i, H_i]$ corresponds to the dimensions $2i - 1$ and $2i$ of the Cartesian space. Intuitively, a subscription specifies ranges of interest over the attributes. The starting point and the end point of the range over the ith attribute can be mapped to the two dimensions



(a) Regions of Events affecting a subscription

(b) Region of subscriptions affected by an event

Fig. 6. Publish/Subscribe Overlay Structure for Single Attribute Space

corresponding to attribute A_i. Therefore the domain of the $2i-1$ and $2i$ axes in the Cartesian space is bounded by $[L_i, H_i]$. This logical space is partitioned into zones among the peers present in the system, and each peer owns a zone. As was the case in Section 4.1, the peers maintain information about the coordinates of its own zone as well as their neighboring zones. Figure 4(a) in Section 4.1 represents the subscription space for a single attribute.

When a user wishes to subscribe for some events, the user submits the subscription to a peer in the system. We call this peer the *origin* peer P_o. The origin peer P_o maps the subscription to its corresponding subscription point in the $2d$-dimensional space. The peer whose *zone* contains this point is referred to as *target* peer P_t. P_o needs to route the subscription to the target peer P_t. In order to route the subscription to the target, P_o selects one of its neighbors, which has the closest Euclidean distance in the $2d$-dimensional space to the target point, and forwards the subscription to it. This process of forwarding is continued until the subscription reaches the target zone P_t. When P_t receives the subscription, it stores the subscription along with an identifier (e.g.. IP address, user name etc.). Figure 4(b) illustrates the routing of subscription $\langle 35, 45 \rangle$ from *zone-7* to *zone-10*.

When an event is introduced into the system, the publish/subscribe system is required to find all the matching subscriptions installed in the system, and deliver the event to the subscribers. Consider an event $\mathcal{E} = \{A_1 = c_1, A_2 = c_2, \ldots, A_d = c_d\}$. A subscription $\mathcal{S} = (l_1 \leq A_1 \leq h_1) \wedge (l_2 \leq A_2 \leq h_2) \wedge \ldots \wedge (l_d \leq A_d \leq h_d)$ is affected by event $\mathcal{E}$ if the following property holds:

$$\forall i \in \{1, 2, \ldots, d\} \quad l_i \leq c_i \leq h_i$$

Event $\mathcal{E}$ is mapped to the point $\langle c_1, c_1, c_2, c_2, \ldots, c_d, c_d \rangle$ in the $2d$-dimensional space, and is referred to as an *event point*. The shaded area in Figure 6(a) shows the region of event points in a $2d$ Cartesian space corresponding to a single attribute schema that can affect the subscription $\mathcal{S}$, because all the event points in the shaded region will satisfy the above property. Notice that all events that affect a subscription $\mathcal{S}$ in the system are located in the bottom right region of the subscription point.

When an event is introduced in the system at a peer P_o referred to as *origin* peer, P_o maps the event to its corresponding *event point* and routes the event to the *target* peer P_t which contains the event point. Figure 6(b) shows the routing path of an event E and the affected region of E in a $2d$ space. The event is then propagated starting from P_t to all peers which are in the region affected by the event. P_t sends the event to its immediate neighbors in the affected region, which in turn propagate the event to their neighbors in the affected region. This process continues until all peers in the affected region have been notified of the event.

A particular challenge in P2P systems is the uniform distribution of load among the different peers in the system. Traditional P2P systems are oblivious to the content of the data and hence use a uniform hash function to distribute the data among the different peers. However, in a content-based publish/subscribe system, we distribute the subscriptions and events based on their content. Most real world datasets tend to be skewed and hence will cause a non-uniform distribution of load on the peers. Hence, unlike previous work, we need to use the characteristics of the load to determine how to distribute the load. We have explored alternative approaches. For example, zones that are overloaded due to a heavy subscription load could split their zones with new coming peers. Alternatively zones that are loaded due to heavy event propagation could replicate their information with new peers, thus distributing the event propagation load. These are interesting and novel challenges, which were not studied in earlier P2P systems due to their uniform distribution characteristics. Once, the content of the subscriptions and events is taken into consideration, these issues become crucial for the success of P2P in such data-intensive applications.

Another aspect of 2CAN is that subscriptions with large attribute extents tend to cluster in the upper-left region of the attribute space. In the point-over-boxes model, however, we have the flexibility of representing large subscriptions in terms of subscriptions with smaller extents. Note that this does not cause the end-user any problems since this mapping is completely transparent to the users. In contrast, in the context of box-over-boxes model, this is not possible since breaking-up d-boxes with large extents will result in either some query boxes not being answered or the query box being flooded to a large number of peers.

The P2P publish/subscribe system can be utilized in critical systems for event monitoring applications, for example power distribution system. The power distribution system consists of *Power Stations* which generate power and send it to *Transmission Substations*. These transmission substations use high voltage transmission lines to convey power to various *Power Substations* which are located in different geographical areas. The power substations step down the power voltage and distribute it to the residential locations. The power system has sensors that measure the amount of power generated (in MW), transmitted and consumed (in KW) at various points within the system. Sensors also measure the voltage in the transmission lines. Monitoring agents can specify continuous queries that detect anomalies, like sudden drop in voltage over transmission lines or a trip in power generation. This can lead to early detection of problems that can lead to catastrophe. We are currently involved with an energy related company which manufactures hardware for monitoring electrical systems. We plan to work with this company to develop an event notification architecture built on top of these hardware devices which are full-fledged Linux based computing and communication platforms.

6 Conclusion

In this paper, we point out that distributed data access at a fundamental level can be classified in four distinct categories. The first one (point-over-point) is the model that has been addressed and solved in existing P2P systems. The second one (box-over-points) represents the basic range searching functionality of traditional database systems. This model can be solved using simple extensions of current P2P systems. However, we argue that this model has certain weaknesses as a general model for distributed data access, and is unlikely to be efficient. We, therefore, propose a natural generalization, the box-over-boxes model, which can handle data distribution at an arbitrary granularity. The box-over-boxes model also allows a mixed use of data replication and caching: the boxes stored at peers can be either data chunks or cached answers to previous range queries. The system tries to find a superset box containing the query box. If no box contains the query box, we can find partial matches. Finally, our point-over-boxes model is a new direction in distributed data service. This is a particularly interesting new model suggested by the new types of "event-notification" services that are emerging around the Internet.

Acknowledgments

We would like to thank the following students for their contributions to the work described in this paper: Abhishek Gupta, Anshul Kothari, Ozgur D. Sahin, Shyam Anthony, and Chiranjeeb Bourgohain.

We would also like to acknowledge our colleagues Michel Raynal (IRISA, France) and Achour Mostefaoui (IRISA, France) for their feedback on this work.

This work has been supported by the NSF under grants CNF-04-23336, IIS-02-23022, IIS-02-20152, IIS-02-09112, INT-00-95527, and EIA-0080134.

References

1. Ganesh A., Rowstron A., Castro M., Druschel P., and Wallach D. Security for structured peer-to-peer overlay networks. In *Proceedings of the Fifth Symposium on Operating Systems Design and Implementation (OSDI'02)*, 2002.
2. Artur Andrzejak and Zhichen Xu. Scalable, efficient range queries for grid information services. In *Proceedings of the 2nd IEEE P2P*, pages 33–40, 2002.
3. Hari Balakrishnan, M. Frans Kaashoek, David Karger, Robert Morris, and Ion Stoica. Looking up data in P2P systems. *Communications of the ACM*, 46(2):43–48, Feb. 2003.
4. Guruduth Banavar, Tushar Chandra, Bodhi Mukherjee, Jay Nagarajarao, Robert E. Strom, and Daniel C. Sturman. An efficient multicast protocol for content-based publish-subscribe systems. In *Proceedings of the 19th IEEE International Conference on Distributed Computing Systems*, pages 262–272, 1999.
5. Antonio Carzaniga, David S. Rosenblum, and Alexander L. Wolf. Design and evaluation of a wide-area event notification service. *ACM Transactions on Computer Systems*, 19(3):332–383, 2001.
6. M. Castro, P. Druschel, A. Kermarrec, and A. Rowstron. SCRIBE: A large-scale and decentralized application-level multicast infrastructure. *IEEE Journal on Selected Areas in communications (JSAC)*, 20(8):100–110, 2002.
7. I. Clarke, O. Sandberg, B. Wiley, and T. Hong. Freenet: A distributed anonymous information storage and retrieval system. In *Proc. of the ICSI Workshop on Design Issues in Anonymity and Unobservability*, July 2000.
8. Brian Cooper and Hector Garcia-Molina. Peer-to-peer data trading to preserve information. *Information Systems*, 20(2):133–170, 2002.
9. Francisco Matias Cuenca-Acuna, Christopher Peery, Richard P. Martin, and Thu D. Nguyen. PlanetP: Using Gossiping to Build Content Addressable Peer-to-Peer Information Sharing Communities. Technical Report DCS-TR-487, Department of Computer Science, Rutgers University, May 2002.
10. Roger Dingledine, Michael J. Freedman, and David Molnar. The free haven project: Distributed anonymous storage service. In *Workshop on Design Issues in Anonymity and Unobservability*, number 2009 in LNCS, pages 67–95, 2000.
11. I. Foster and A. Iamnitchi. On death, taxes, and the convergence of grid and p2p computing. In *Proceedings of the International Workshop on P2P Systems*, 2003.
12. I. Foster and C. Kesselman. Globus: A Metacomputing Infrastructure Toolkit. *The International Journal of Supercomputer Applications and High Performance Computing*, 1997.
13. Michael J. Freedman and Robert Morris. Tarzan: A peer-to-peer anonymizing network layer. In *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS 2002)*, Washington, D.C., November 2002.
14. Gnutella. http://gnutella.wego.com/.
15. Steven Gribble, Alon Halevy, Zachary Ives, Maya Rodrig, and Dan Suciu. What can peer-to-peer do for databases, and vice versa? In *Proceedings of the Fourth International Workshop on the Web and Databases (WebDB 2001)*, Santa Barbara, California, USA, May 2001.
16. A. Gupta, D. Agrawal, and A. El Abbadi. Approximate range selection queries in peer-to-peer systems. In *Proceedings of the First Biennial Conference on Innovative Data Systems Research*, Asilomar, California, January 2003.
17. Matthew Harren, Joseph M. Hellerstein, Ryan Huebsch, Boon Than Loo, Scott Shenker, and Ion Stoica. Complex queries in DHT-based peer-to-peer networks. In *Proceedings of the first International Workshop on Peer-to-Peer Systems*, 2002.

18. J. Hellerstein. Sensor networks. The Gong Show, First Biennial Conference on Innovative Data Systems Research, 2003.
19. Xin Li, Young Jin Kim, Ramesh Govindan, and Wei Hong. Multi-dimensional range queries in sensor networks. In *Proceedings of the ACM SenSys 2003*, 2003.
20. C. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker. Search and replication in unstructured peer-to-peer networks. In *Proceedings of 16th ACM International Conference on Supercomputing(ICS'02)*, June 2002.
21. Dahlia Malkhi, Moni Naor, and David Ratajczak. Viceroy: A scalable and dynamic emulation of the butterfly. In *Proceedings of the 21st ACM Symposium on Principles of Distributed Computing*, 2002.
22. Napster. http://www.napster.com/.
23. Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard Karp, and Scott Shenker. A scalable content-addressable network. In *Proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications*, pages 161–172. ACM Press, 2001.
24. A. Rowstron and P. Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In Proceedings of IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), Heidelberg, Germany, pages 329-350, 2001.
25. O. D. Sahin, A. Gupta, D. Agrawal, and A. El Abbadi. Query processing over peer-to-peer data sharing systems. Technical Report UCSB/TR-2002-28, University of California at Santa Barbara, 2002.
26. Emil Sit and Robert Morris. Security considerations for peer-to-peer distributed hash tables. In *Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS)*, Cambridge, MA, March 2002.
27. Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek, and Hari Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *Proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications*, pages 149–160. ACM Press, 2001.
28. Alexander S. Szalay, Jim Gray, Ani Thakar, Peter Z. Kunszt, Tanu Malik, Jordan Raddick, Christopher Stoughton, and Jan vandenBerg. The SDSS skyserver: public access to the Sloan digital sky server data. In *Proceedings of the ACM International Conference on Management of Data*, pages 570–581, 2002.
29. Shelley Q. Zhuang, Ben Y. Zhao, Anthony D. Joseph, Randy H. Katz, and John D. Kubiatowicz. Bayeux: an architecture for scalable and fault-tolerant wide-area data dissemination. In *Proceedings of the 11th international workshop on Network and operating systems support for digital audio and video*, pages 11–20. ACM Press, 2001.

Governing the Contract Lifecycle: A Framework for Sequential Configuration of Loosely-Coupled Systems

Harumi Kuno[1], Kei Yuasa[1], Kannan Govindarajan[1], Kevin Smathers[1],
Bernard Burg[1], Paul Carau[2], and Kevin Wilkinson[1]

[1]Hewlett-Packard Laboratories, 1501 Page Mill Rd., Palo Alto, CA 94304
[2]HP Technology Solutions Group (TSG), 3404 E Harmony Rd.,
Fort Collins, CO 80528-9599

Abstract. Sequential configuration is a fundamental pattern that occurs when integrating systems that span domains and levels of abstraction. This task involves not only the integration of heterogeneous autonomous information systems, but also the integration of processes and applications. Challenges include the lack of a common information model, the lack of explicit correlations, and the lack of common taxonomies between the systems. We propose here an extensible system for correlating sequential configurations across loosely-coupled systems. Our system includes a framework that defines fundamental abstractions and interfaces that enable the implementation of domain-specific models. We also provide a suite of tools that work upon objects that implement our framework's interfaces and abstractions. The tools can then be used to support applications that manage the configuration lifecycle. We have implemented a prototype application, called the Deal Configurator, on top of our framework. The Deal Configurator integrates the processes, tools, and data involved in the first two stages of the contract lifecycle.

1 Introduction

This work is motivated by the problem of how an outsourcing provider can efficiently manage its contract lifecycle. In outsourcing, a provider manages both technical components and business aspects of an enterprise customer's IT operations, or alternative critical but non-core business processes. For example, when HP agrees take over desktop management for a large customer, this means that HP contracts to manage the complete lifecycle for desktops for all of that customer's employees all over the world.

From the perspective of the provider, the service lifecycle can be modeled as a sequence of configurations across loosely coupled systems that involve human processes. As shown in Figure 1, the Sales team selects and configures the terms of a contract to meet customer needs. The Solution Architects must select and configure service features that fulfill the terms of the contract. The Delivery Manager must select and configure delivery steps that implement the configured

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 264–279, 2005.

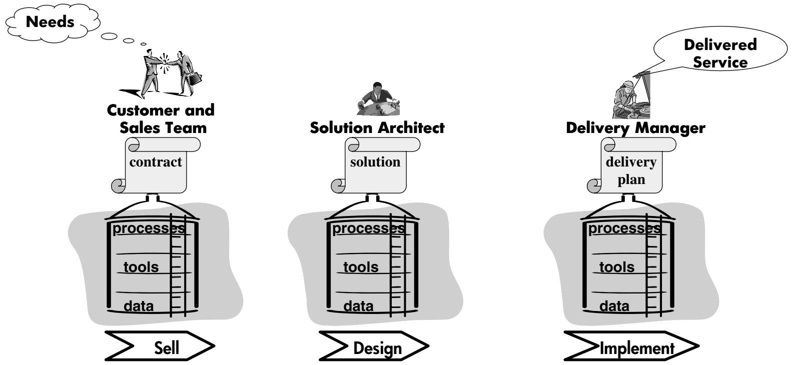


Fig. 1. The outsourcing contract lifecycle spans silos of processes, tools, and data

service features. The ongoing Delivery Manager must select and configure an ongoing service delivery plan that will meet the customer's need. Finally, at each stage, the resulting configuration must reflect the capabilities of the next lifecycle stage. Without both forward and backwards coordination, the provider will not be able to deliver on the contract terms in an efficient and cost-effective manner.

Like other applications that manage system configuration, any solution to this problem must model properties of system components and configured systems, capture the requirements and constraints that apply to configurations, manage dependencies between components, and handle the optimization tradeoffs. In addition, we must also address the unique needs of the outsourced service context. First, we must support continuity of the configuration process across silos of organizations, processes, and data. Tracing customer needs from requirements through to implementation helps both ends understand the impact of newly proposed changes. This is difficult because the organizations that support the various stages of the lifecycle are loosely-coupled, and do not share processes, applications, or data. Second, continuity is two-way. We need to implement a feedback loop. This is important because the Sales Team needs to sell solutions that reflect the capabilities of the Delivery Team to the best advantage. Third, customization is very important because most solutions are require some degree of customization. Ideally, we would like to implement the mass customization of services. That is, we would like to enable the configuration of standardized products and their components into unique combinations - thus producing highly customized offerings. Finally, in addition to hardware and software, we also need to handle other resource types, such as humans, real estate, currency, etc. For example, if the outsourced service provider wishes to constract with a customer to deploy a solution in Lithuania, the provider may need to determine whether the right people, tools and resources are available to meet the customer's needs. That is, the provider may need to know whether any current employees are experienced with this particular customer (perhaps in a past job), or whether the standard solutions can be deployed in Lithuania.

We introduce here an extensible platform for correlating sequential configurations across systems. We begin by placing our efforts in the context of related work (Section 2). Our approach, described in Section 3, enables the correlation of arbitrary data objects across configurations. Our system, whose architecture we discuss in Section 4, is unique in that we provide an explicitly extensible framework that supports the integration of additional autonomous domains, as well as in that our approach judiciously automates of easily computable tasks while leveraging human input for tasks requiring semantic judgements. We have implemented an initial prototype application, called the Deal Configurator on top of our platform, and discuss the validation of our approach by testing the Deal Configurator with actual data collected during an actual deal with a major customer, which we discuss briefly in Section 5. We conclude with observations about this work, plus some directions for future work (Section 6).

2 Related Work

One perspective of our work is that we address the problem of how to provide an integrated view over a set of autonomous information systems, each with its own data sources, tools, and processes. The underlying information systems are loosely coupled. They operate on different levels of abstraction, which results in different perspectives of shared concepts. These differing perspectives are expressed using different ontologies, models, and data structures. Furthermore, the information systems are subject to change–we must be able to accommodate changes to the existing systems and to add support for new systems easily.

The problem of integrating heterogenous data sources is known to be difficult. Most work has focused on the general problem of supporting arbitrary queries over the integrated systems. The Stanford-IBM Manager of Multiple Information Sources (better known as TSIMMIS) supports the querying and browsing of hetergeneous information systems [4]. IBM's Garlic project provides a query processor that can optimize and execute arbitrary SQL queries over diverse data sources [5]. InfoSleuth focuses on the problems of locating, evaluating, retrieving, and merging information from heterogeneous locally-autonomous sources in dynamic environments [9].

Our problem is more constrained than this general problem. In particular, unlike systems that provide integrated access to multiple distributed heterogeneous information sources, we do not need to support arbitrary queries. Our goal is instead to manage correlations between key concepts.

In that, we share some common goals with work on semantic metadata. The Resource Description Framework (RDF) and Web Ontology Language (OWL) support the modeling of and reasoning over sophisticated relationships [6, 7]. However, they do not specifically support system configuration. Melnick et al use RDF to implement the Generic Interoperability Framework, which facilitates the integration of heterogeneous information systems [8]. They propose a system for model management and define an algebra to manipulate models and mappings. They address problems about matching models, merging models, selecting and extracting a subset of a model. They also propose generic mechanisms to maintain the mappings between models. However, the Generic Interoperability Framework focuses on the generic representation of interfaces–protocols, languages, and data, and does not attempt to model the process layer.

In fact, we can consider the configuration artifacts themselves to represent a mediation layer between the systems to be integrated. For example, customers may express their needs in a different language than the language that a Delivery Team technologist can understand, and vice versa. The statement "I need 98% completed sessions" may be meaningful to a customer in a business sense, but is meaningless to the technologist, who does not control whether or not customers complete their sessions. The terms of the service level agreement represents an intermediate formulation expressed in terms that both the customer and the Delivery Team can understand: "Service shall maintain less than 2s latency." The business is able to evaluate this requirement in light of their business needs. If some hypothetical service had a parameter that could be tweaked that let a new instance be allocated every N sessions, the technologist could translate the latency requirement into a meaningful configuration.

That is to say, unlike other systems, whose goal is to automate the integration process for the purpose of supporting arbitrary queries, our goal is support the configuration process. Our objective is thus to support, not replace, human processes. This is significant because it allows us to exploit the input of knowledgeable humans for tasks requiring semantic interpretation.

Similarly, recent interest in Utility Computing has led to efforts to create automated resource management systems for the provisioning, configuration, and lifecycle management of computing resources [10]. Utility Computing addresses the dynamic outsourcing of applications and business processes. Most researchers in this area are focused on automating the process by which service consumers identify service providers and service attributes [3, 10]. That is, they address issues such as service interoperability requirements, how to programmatically establish trust between service providers and consumers, and how to automate the management of the service level agreement (SLA) lifecycle [3].

Our work addresses a more general outsourcing problem than utilty computing. Figure 2 compares Utility Computing to general outsourced services along the axises of technology (standard solutions v.s. custom solutions), resources (shared resources v.s. dedicated resources), and proceses (automated processes v.s. manual processes). The future envisioned by utility computing builds has standard technology, shared resources, and automated IT processes. The reality of outsourced services today, is that customers require custom solutions, that it is often not practical to share technology, processes, and people across customers, and that many of the processes involved in the outsourcing lifecycle are still manual. Our goal is to enable outsourcing to move towards the ideal of utility computing, but we recognize that some outsourced services, such as service desk, will always have an inherent human component. Unlike utility computing, which focuses on computing rather than human resources, we believe that ultimately

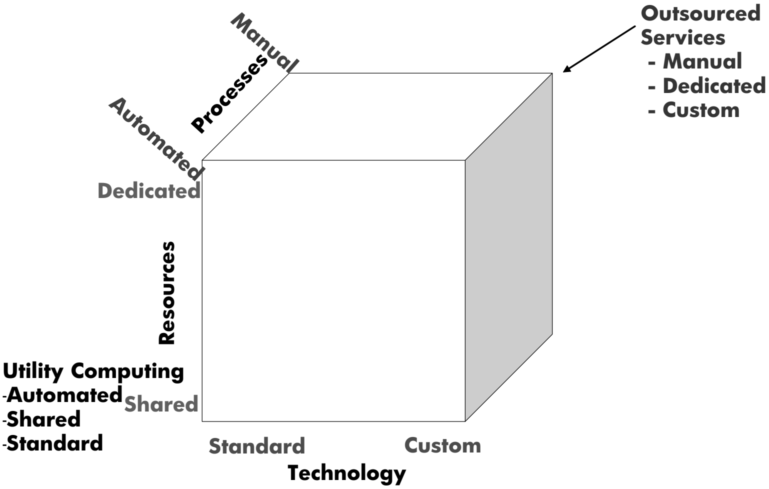


Fig. 2.

we are integrating human processes across the organizational silos involved in configuring and deploying outsourced services.

Although some academic work addresses mass customization for outsourced services, to the best of our knowledge there is no competition extant in the outsouring provider domain for configuration. Many companies are in the arena of IT outsourcing provisioning. These include IT service providers who compete directly with HP, companies that provide tools for IT service management, companies that sell infrastructure for procurement and spend management, and enterprise application suite vendors. We believe that our efforts are unique in that we are focusing on an industry (IT outsourcing service providers) and a lifecycle stage (service design and outsourcing) that other service vendors do not address.

3 Approach

We define a configuration to generally consist of the selection of terms from a candidate set, and the assignment of values for those terms. Our goal is to manage the continuity of correlation across configurations. This goal is complicated by the requirement that the configured systems are autonomous and continually evolving in design and implementation. To this end, we have designed an extensible platform to support a series of applications designed to enable a continuous information flow between the stages of the outsourcing contract lifecycle.

Current knowledge management systems do not support the translation of information into appropriate constructs and formats as it moves from stage to stage. That is, current knowledge management systems either provide information that is of use to a specific stage, or provide very generic information that can be used by more than one stage. Our approach is a hybrid; we translate information into the terms needed by each team as the stages of delivery progress so that data is linked both within and across silos.

To support data transformation, we go beyond data storage and retrieval, and incorporate cross-functional processes and analysis tools for decision-making and correlation. We define a set of generic interfaces and constructs that support general functionality. Given an existing sequence of configurations, our system supports the definition of additional configuration steps. Once such an additional step has been defined, our system's correlation capabilities will operate upon the new step's constructs, automatically inferring relationships with pre-existing configuration components.

There are three main aspects to our approach.

1. We provide a framework of constructs and interfaces that enable the management of correlation and information flow.
2. We use these constructs and interfaces to create a domain-specific information layer that serves as an integration point for the structure and semantics of key concepts involved in the configuration steps. The model objects implementing our interfaces can be immediately plugged into our integration framework and be manipulated with our tools. This layer insulates our platform from implementation changes to the underlying data sources (the silos from Figure 1).
3. We provide tools that support the configuration processes and manage (visualize and manipulate) the relationships between configuration components. The tools are leveraged by applications built on top of our platform.

The key to our approach is that we automate the computable tasks of relationship information management and configuration process management, exploit human input for the difficult tasks of term/value configuration and mapping validation, and integrate the relationships discovered during the human input phase back into the system. This enables us to manage correlations more accurately and efficiently than either a fully-automated system or a non-automated system could.

4 System Architecture

Our goal is to provide a high fidelity information stream that spans the processes, tools, and data of the systems involved in the outsourced service contract lifecycle, transforming data into critical information that flows between stages. To this end, we are building an enterprise platform to support the outsourced services contract lifecycle.

We have designed a four layer architecture, sketched in Figure 3. The Framework Layer defines generic interfaces and constructs that facilitate correlation. The Domain Layer manages domain-specific objects that implement interface from the Framework Layer. The Tool Layer provides a collection of tools for visualizing and managing relationships between Domain Layer objects. Because these tools are written to use the generic Framework Layer interfaces, Domain

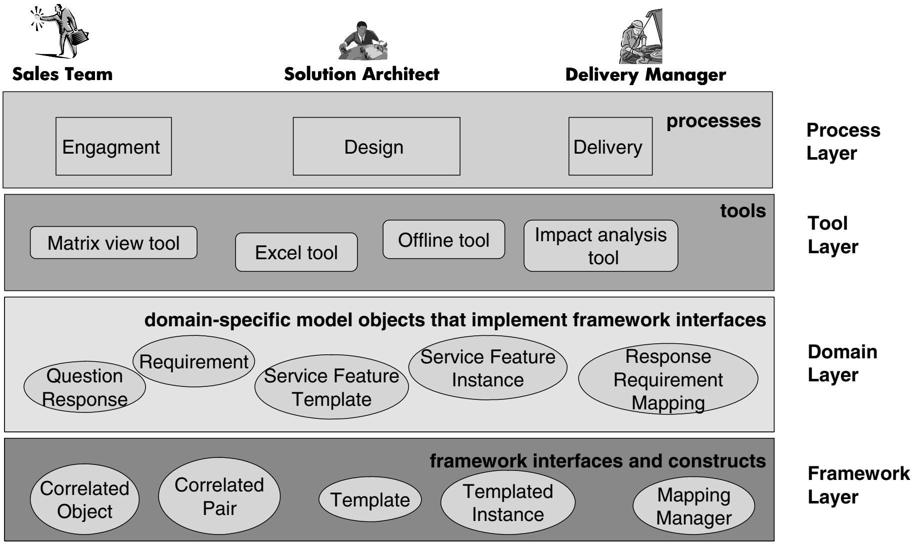


Fig. 3. Our four layer architecture abstracts from domain-specific concepts to processes that span the outsourced service lifecycle

Layer objects can be simply plugged into the tools. The Process Layer uses the tools to provide interfaces to configuration processes, for use by knowledgeable humans.

4.1 Framework Layer

In order to support the flexibility needed to incorporate new domain-level constructs into the system, we provide framework interfaces and constructs that facilitate the management of relationships between domain objects. We use these constructs to support a variety of correlation patterns between the objects.

For example, we define generic interfaces for configuration templates and instances of configuration templates. The template interface supports a generic findOrCreate method that takes some identifier that can be used to find or create an appropriate instance of the template. We explicitly define a set of constructs to manage the configuration parameters (as well as their ranges) that are associated with a given template.

Similarly, we define a number of constructs and interfaces to capture mappings between template and instance objects as well as a generic interface for domain-level objects that can be correlated. The interface for correlated objects dictates some general properties for use in displays (such as display name, popup name, full name, etc.), as well as some generic operations (e.g., the ability to save changes). We also define interfaces to capture correlations of two or more correlated objects, as well as collection objects that represent lists of both correlations and correlated objects. These lists support access to the various properties of its contents.

These correlated object constructs are used by a Mapping interface, which is designed to manage the correlation between sets of correlated objects. The Mapping interface implements methods to access the the correlations managed by the mapping object; defines indices that map object identifiers to correlation axis indices; and methods that persist changes to the correlations represented by the mapping to the appropriate data sources. Mapping objects can also analyze their correlations–for example identifying elements that do not participate in any mappings.

Given two instances of Mapping objects representing sequential stages of configuration (e.g., MappingAB correlates SystemA to System B and Mapping BC correlates System B to System C) and a pivot point (e.g., System B), the two Mapping instances can be automatically merged to create a third instance of Mapping correlating System A to System C. This facilitates both root cause and impact analysis, as well as insurance of coverage (e.g., ensuring that a service delivery plan meets all of a customer's requirements).

We also define interfaces that support the management of these correlations. In particular, for each pair of systems to be correlated, we create a mapping manager. the Mapping Manager interface is responsible for generating appropriate Mapping objects.

4.2 Domain Layer

The Domain Layer serves as an integration point for the structure, semantics, and content of the key domain-specific concepts integrated by our system. By structure, we mean that this layer defines a canonical model. For example, this layer includes constructs we use to capture customer needs and requirements. By semantics, we mean the relationships between the concepts and the functions and operations related to them. For example, when we correlate customer needs to requirements to service feature configurations, we capture coverage and tracability between customer needs and service features. Together, the structure and semantics help us design parameterized standard components for each lifecycle stage. By content, we mean that the Domain Layer is responsible for integrating the various data sources. For example, our prototype integrates a number of repositories that capture collective knowledge about resources, tools, and processes.

If we were dealing with a network management system, we could have leveraged standards such as the Common Information Model (CIM), which specifies how management information databases should represent information about logical and physical objects on a managed network for access by CIM-compliant network management systems [2]. However, there is currently no equivalent of CIM for the domain of outsourced service providers. The content for this layer thus represents a significant effort that we are approaching incrementally, leveraging as many existing models as we can.

To give an idea of the potential scale of this layer, SAP defines tens of thousands of data entities. Our system currently defines only about a hundred domain-specific entities related to outsourced service provisioning. For example, these include entities to support the engagement evaluation, requirement gathering, service feature configuration, and delivery plan creation processes.

The Domain Layer leverages the constructs and interfaces defined by the Framework Layer to implement classes that model domain-specific concepts. A given class can implement both the CorrelatedObject and Template interfaces, or both the CorrelatedObject and Instance interfaces. The system can thus leverage associations between a CorrelatedObject from System A and a Template object from System B, and use the Template object to automatically prompt a knowledgeable human to create a configured instance of System B that is appropriate for the Correlated Object from System A.

This enables our system to manage (visualize and manipulate) the relationships between configuration components. The generic interfaces provide a simple means to correlate arbitrary information across configurations. For example, given two systems, System A and System B, whose configurations should be correlated, we can easily initialize our system to manage the configurations of these systems:

1. Identify constructs representing the configuration object for each system. For example, in the case of configuring a service feature set that meets customer needs, we might use the construct of a response to a questionnaire question to model a customer need.
2. If the two systems do not share a common vocabulary, then identify or create an intermediate configuration system. For example, customer needs are captured using free form text and business terms but the service features are configured via application-specific technical parameters. We therefore create an intermediate construct, Requirements, to mediate between the customer needs and service features.
3. Optionally, create a class implementing the Template interface to manage the configuration of instances of each system.
4. For each construct to be correlated, create a class implementing the CorrelatedObject interface.
5. Create a class implementing the CorrelatedPair interface for the actual correlation.
6. Create a class implementing the Mapping interface (e.g., that reflects existing mappings between CorrelatedObjects and that can update the stored mappings).
7. Implement a method capable of creating and returning an appropriate Mapping object for a given configuration from System A. E.g., we might implement a manager that can create and return a mapping object for managing which objects from System A are associated with which objects from System B.

These steps can be repeated for each additional configuration stage. It is thus easy to integrate a new configuration stage into our system. Note that the objects in our domain layer may be backed by objects from any number of underlying data sources.

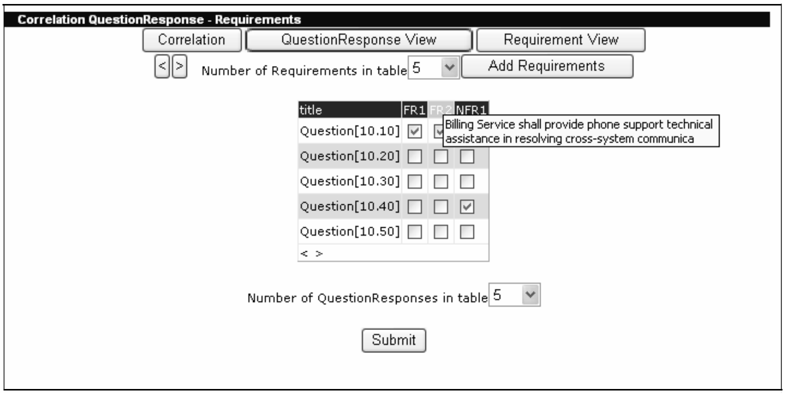
4.3 Tool Layer

The tool layer is critical because we must leverage existing tools and processes as much as possible. These include common tools such a correlation tool, workflow engines, etc; and the personal productivity tools upon which our users rely. We have made a point of explicitly supporting Excel as an interface to the Deal Configurator. In addition, we need our system to integrate with HP-internal tools (which standardizes the infrastructure and operational processes within delivery). Finally, we intend for our platform to act as a vehicle to deliver other solutions from HP Labs to the business unit, so this layer must support the plug-in of other solutions.

The Matrix Tool component manages user interface functionality. It is capable of manipulating and visualizing mapping objects. For example, the Matrix can produce a clickable matrix for display on a Web browser that a human user can use to manipulate the correlations between pairs of CorrelatedObjects. The Matrix also provides generic functions for manipulate mappings, for example computing the correlation between elements of two mappings or identifying elements that participate in contradictory mappings.

That is to say, given an instance of a Mapping, Matrix Tool can automatically create a data grid that shows the associations between the Correlated Objects in an intuitive manner, and that a knowledgeable human can use to correct correlation between the two systems. For example, Figure 4 shows a user interface automatically constructed by the User Interface manager automatically constructed from a Mapping that correlates QuestionResponses to Requirements. In the figure, the user has requested to see the correlations using a five-by-five matrix. The number of cells shown is configurable by drop down menu. As the user moves their mouse across the column and row headings, as well as over the individual cells, pop-up windows appear with detailed information about the correlated objects. The user can click in the cells to turn correlations on and off, and click on the Submit button to persist their changes.

Our tool layer is also designed to support personal productivity applications. For example, we provide an Excel Tool that uses the off-the-shelf Microsoft Excel spreadsheet application to implement the operational closure and hold the data closure. We employ Visual Basic macros to implement our closure of operations (including both user interface and data manipulation functions), and use Excel's spreadsheet capabilities to hold the data closure. Having identified these macros and the dataset, our server application can dynamically create an appropriate Excel spreadsheet and load the current dataset into it. The user can then download this spreadsheet file onto their local computer and work with it offline. When the user re-establishes their network connection, they can upload the modified spreadsheet file back onto the server, which is then responsible for synchronizing the data in the cells of the modified dataset back into the system. By using the spreadsheet format, it is possible to put operations and logics in the spreadsheet cells.



Correlation QuestionResponse - Requirements

Number of Requirements in table 5

title	FR1	FR2	NFR1
Question[10.10]	☑	☑	
Question[10.20]	□	□	□
Question[10.30]	□	□	□
Question[10.40]	□	□	☑
Question[10.50]	□	□	□
< >			

Billing Service shall provide phone support technical assistance in resolving cross-system communica

Number of QuestionResponses in table 5

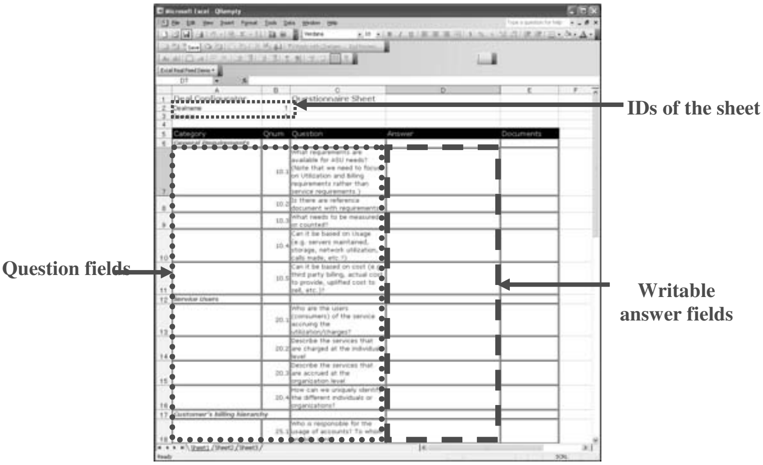
Fig. 4. The Matrix Tool can automatically construct interfaces for managing correlation between Domain layer objects that implement interfaces defined by the Framework layer

That is to say, this tool can perform the following tasks:

1. Identify core operations and data.
2. Identify the closure of operations (operations needed directly or indirectly by the core operations).
3. Identify the data closure (data needed either directly or indirectly to run the core operations upon the core data).
4. Identify or create a client application that supports the closure of operations.
5. Load the client application with the data closure. This client represents a functional closure that can run on a disconnected client machine.
6. Download the loaded client application onto the user's machine, which the user can then use to work offline.
7. Ingest of the closure into the data in the server side when the connection is reestablished

Figure 5 shows an example of downloaded data in Excel file. The web-based application with the presented invention has a step of questions where the user answers to the number of questions. The offline operation is provided for this step so that the user can download an answer sheet in Excel format in Figure 5. In the top 2 lines, there are cells for identifying this answer sheet. The table below is the list of question and answer pairs. The questions are stored in the left 3 columns, and the 4th column is for answer text. By using locking function of the spreadsheet application, the user can only fill in the answer cells. When this answer sheet is uploaded to the server, the identifiers are used to check the identity of the data. The identifiers, question texts, and the form of the table are not changeable.

In the future we plan to extend our Framework to support more sophisticated relationships. For example, we might define an exclusive-or relationship between components, indicating that one service feature precludes another. The Matrix



Deal Configurator | Questionnaire Sheet

Dealname 1

Category	Qnum	Question	Answer	Documents
General Requirements				
	10.1	What requirements are available for ASU needs? (Note that we need to focus on Utilization and Billing requirements rather than service requirements.)		
	10.2	Is there are reference document with requirements		
	10.3	What needs to be measured or counted?		
	10.4	Can it be based on Usage (e.g. servers maintained, storage, network utilization, calls made, etc.?)		
	10.5	Can it be based on cost (e.g. third party billing, actual cost to provide, uplifted cost to sell, etc.)?		
Service Users				
	20.1	Who are the users (consumers) of the service accruing the utilization/charges?		
	20.2	Describe the services that are charged at the individual level		
	20.3	Describe the services that are accrued at the organization level		
	20.4	How can we uniquely identify the different individuals or organizations?		
Customer's billing hierarchy				
	25.1	Who is responsible for the usage of accounts? To whom		

Fig. 5. The Excel Tool supports disconnected operation without the need to pre-install dedicated applications on the client machine

Tool could then determine when an early stage configuration leads to conflicts in a later stage configuration.

4.4 Process Layer

Ultimately, outsourcing represents the delegation of IT operations. The business process is the fundamental unit of IT operation. It is thus critical that our system enable the correlated configuration of business processes between organizational silos, as well as the correlation of the domain objects to business processes.

Business processes are typically organized using hierarchical levels of abstraction. For example, the Supply Chain Operations Reference (SCOR) Model addresses three levels of process detail [12]. Level 1 (Process Types) defines the scope and content for the reference model and sets the basis of competition performance targets. Level 2 (Process Categories) determines the configuration of the supply chain. Level 3 (Decompose Processes) fine tunes the operation strategy, defining process elements, their inputs and outputs, and identifying best practices.

Similarly, [11] defines five levels of abstraction for business processes:

- Level 0: Process domain (company strategy/business goals)
- Level 1: Process types (program) (opportunity development phase between 1 and 2)
- Level 2: Configuration level (process categories e.g., from SCOR) (scoping phase between 2 and 3)
- Level 3: Process element level (analysis phase between 3 and 4)
- Level 4: Decomposed process element level (organization-specific process steps defining how things are performed) (design phase between 4 and 5)
- Level 5: Work steps (clearly identifies steps to meet objective)

The number of elements at any level increases by a branching factor of approximately 10 at each level. For example, a company with ten strategic goals might have approximately hundreds of programs, which would potentially map to thousands of SCOR process categories, tens-of-thousands of process elements, etc.

The challenge is how to integrate these elements between levels of abstraction and across silos of organizations, data models, and tools. Without mechanisms to help manage this integration, IT Management can only supply IT services and products to meet current needs, as opposed to governing the course of both present and future business operations by linking business processes to IT operations.

The framework layer of our system currently includes constructs that help capture the lower levels of abstraction – e.g., process configuration and delivery steps. We hope to extend it in the future to handle Level 0/1 abstractions.

5 Prototype Implementation

Teams from HP Managed Services and HP Labs collaborated to create the Deal Configurator, the first in a series of applications designed to enable a continuous information flow between the stages of the outsourced service contract lifecycle. This stream ensures that information gathered during the customer engagement phase is transformed accurately and appropriately into the delivery planning phase, and that changes in delivery planning are reflected back to the customer [1]. For example, the Sales Team and the Solution Architects (who design solutions for customers) can use the Deal Configurator to capture the needs of the customer accurately, and transform those needs into a structured form that the implementation teams can use to deliver the service once a contract is signed.

Figure 6 provides a functional overview of the Deal Configurator. There are two anchor points in modeling IT services outsourcing: at one end is the customer's needs, and at the other end is the delivered service. The Deal Configurator captures the customer's needs and transforms them into a service delivery plan by introducing a sequence of translation steps and intermediate constructs, as well as tools that operate upon these. The Service Manager defines the service qualification questionnaire, the features that can be delivered, their costs, and the service can be configured. The Solution Architect completes the Questionnaire to collect customer needs, and then together with the Delivery Manager, translates those needs into requirements. The Delivery Manager then translates the requirements into a deliverable configuration of service features and develops a cost estimate for the delivery plan.

Because the domain knowledge captured by the system is critical, we worked extremely closely with the domain experts so as to validate all assumptions as early as possible. We used iterative, rapid development cycles to build an initial prototype of the Deal Configurator (using ASP.NET), and tested it with actual data collected during an ongoing deal with a major customer. We used the data to mirror a Solution Architect's actions in designing and configuring a solution

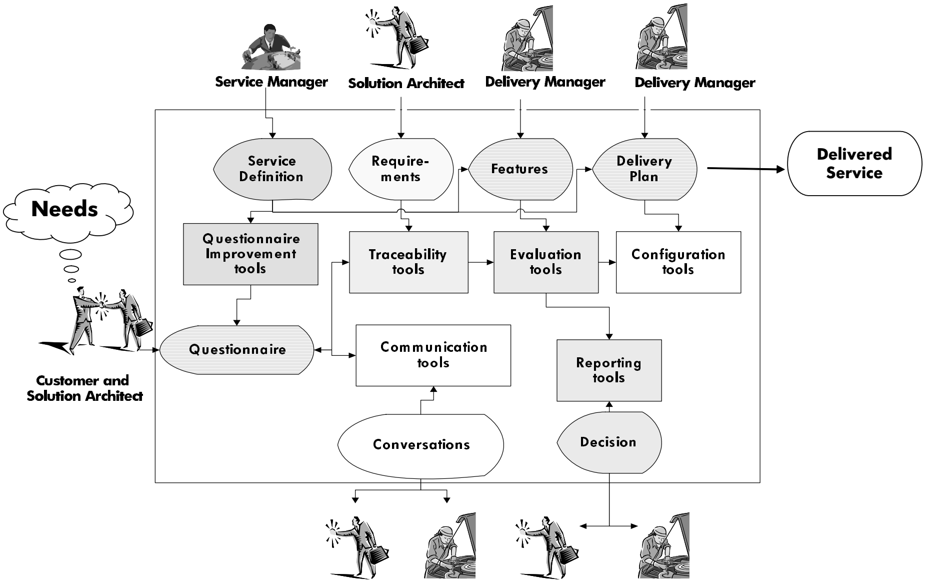


Fig. 6. The Deal Configurator implements a system for configuring service features to meet customer requirements

for a customer. In the course of this test, we validated the following benefits to the Deal Configurator's approach:

- Information Flow Continuity: The Deal Configurator tied all the various information (question, answers, requirements, and features) together, and shows the direct linkages between stages and the applicability of data to each stage as the information moves through the managed service contract lifecycle.
- Data transformation: Data are transformed into the appropriate scope and details for each of the phase within the managed service contract lifecycle (e.g., customer requirements are collected during the sell or customer engagement stage, requirements are mapped into service features and parameters during the design/configuration stage, service features are mapped into step-by-step processes during the implementation stage, etc).
- Automation: We automate the computable tasks of relationship information management and configuration process management, exploit human input for the difficult tasks of term/value configuration and mapping validation, and integrate the relationships discovered during the human input phase back into the system. This enables us to manage correlations more accurately and efficiently than either a fully-automated system or a non-automated system could. For example, the Deal Configurator can help a Solution Architect write more concrete and actionable requirements because each requirement

had to be mapped to service features and parameters. Also, vague requirements or conflicting requirements were addressed with customers earlier and more easily.

6 Conclusions and Future Work

We validated our approach by using the Deal Configurator to mirror a Solution Architect's actions using data from a major HP deal. Our testers validated that the Deal Configurator's approach provided information flow continuity and facilitated data transformation. By automating the computable tasks of relationship information management and configuration process management, exploiting human input for the difficult tasks of term/value configuration and mapping validation, and integrating the relationships discovered during the human input phase back into the system, we can manage correlations more accurately and efficiently than either a fully-automated system or a non-automated system could. By mapping each requirement to service features and parameters, the Deal Configurator led to concrete and actionable requirements, and resulted in the early identification of both vague and conflicting requirements that might otherwise have gone undetected until a later stage of the deal lifecycle.

We are currently extending our prototype to support proposal generation and cost estimation. This integration will extend the Deal Configurator to cover additional stages of the lifecycle currently not addressed by the prototype. We are piloting the combined application for one of the outsourced services that HP provides.

References

1. B. Burg, K. Govindarajan, H. Kuno, K. Smathers, and K. Yuasa. An enterprise applications platform for the managed services business. Technical Report HPL-2004-119, Hewlett-Packard Laboratories, 2004.
2. Common Information Model (CIM) Standards, 2004. http://www.dmtf.org/standards/cim/.
3. A. Dan, D. Davis, R. Kearney, A. Keller, R. King, D. Kuebler, H. Ludwig, M. Polan, M. Spreitzer, and A. Youssef. Web services on demand: WSLA-driven automated management. *IBM Systems Journal*, March 2004.
4. H. Garcia-Molina, Y. Papakonstantinou, D. Quass, A. Rajaraman, Y. Sagiv, J. Ullman, V. Vassalos, and J. Widom. The tsimmis approach to mediation: Data models and languages. *J. Intell. Inf. Syst.*, 8(2):117–132, 1997.
5. V. Josifovski, P. Schwarz, L. Haas, and E. Lin. Garlic: a new flavor of federated query processing for db2. In *SIGMOD '02: Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, pages 524–532. ACM Press, 2002.
6. O. Lassila and R. Swick. Resource description framework (RDF) model and syntax specification, W3C recommendation. http://w3.org/TR/1999/RED-rdf-syntax-19990222, February 1999.

7. D. L. McGuinness and F. van Harmelen. OWL Web Ontology Language overview. http://www.w3.org/TR/owl-features, August 2003.
8. Generic Interoperability Framework, Accessed December 2004. http://www-diglib.stanford.edu/diglib/ginf/WD/ginf-overview/.
9. J. R. J. Bayardo, W. Bohrer, R. Brice, A. Cichocki, J. Fowler, A. Halal, V. Kashyap, T. Ksiezyk, G. Martin, M. Nodine, M. Rashid, M. Rusinkiewicz, R. Shea, C. Unnikrishnan, A. Unruh, and D. Woelk. The infosleuth project. In *SIGMOD '97: Proceedings of the 1997 ACM SIGMOD international conference on Management of data*, pages 543–545. ACM Press, 1997.
10. A. Sahai, S. Singhal, R. Joshi, and V. Machiraju. Automated policy-based resource construction in utility computing environments. In *IEEE 5th International Workshop on Policies for Distributed Systems and Networks (POLICY 2004)*, June 2004. Also appears as HPL Technical Report HPL-2003-176.
11. M. Salle. It service management and it governance: Review, comparative analysis and their impact on utility computing. Technical Report HPL-2004-03, Hewlett-Packard Laboratories, June 2004.
12. Supply Chain Council. SCOR, Supply Chain Operations Reference Model, 2002. http://www.supply-chain.org.

GridFS: Ensuring High-Speed Data Transfer Using Massively Parallel I/O

Dheeraj Bhardwaj[1] and Manish Sinha[2]

[1] Department of Computer Science & Engg., Indian Institute of Technology, Delhi, India
[2] GridLogics Technologies Pvt Ltd, IndiaCo iCenter, 214, L.B.S Road, Pune, India

Abstract. I/O has always been performance bottleneck for applications running on clusters. Most traditional storage architectures fail to meet the requirement of concurrent access to the same file that is posed by most high-performance computing applications. While many parallel and cluster file systems meet this requirement, they are still plagued by metadata overheads and associated management complexities that prevail in read/write intensive scenarios. In this paper we introduce GridFS, a next generation I/O solution that can scale to hundreds or thousands of nodes and several hundreds of terabytes of storage with very high I/O and metadata throughput. It is besed on Object based Storage Architecture (OSA) model and goes a step further to eliminate runtime file access overheads as compared to other implementations on the same model. By eliminating most access overheads and optimizing metadata, GridFS outperforms other solutions in read/write intensive scenarios and this makes it better suited for I/O intensive applications like seismic analysis, weather forecasting, genomics and 3D/4D design simulations.

1 Introduction

The Linux clusters have become mainstream for high-performance computing (HPC) solutions and more than one third of the Top 500 supercomputers are now based on cluster configurations [1]. While performance may have steadily increased with the number of nodes, the technology for enhancing speeds of the I/O system has not kept the pace. The problem is all the more aggravated by the sharp-rise in the node count. So although parallel-computers are quick in resolving highly-computation tasks, they do not perform well with applications, which are almost always I/O intensive.

The I/O problems are also better solved in parallel processing and parallel I/O is needed for higher performance [2]. Most clustering applications employ a scale-out approach for parallelization in which core problem is broken into thousands of independent tasks and is distributed across a set of nodes. For such applications data management poses significant operational challenges especially in large cluster and grid computing environments where core datasets change regularly. Present approaches such as MPI-IO provide optimizations at an application-layer and are still plagued by problems inherent to the underlying file system and storage topology [3].

This paper introduces **GridFS** a high performance distributed file system that eliminates all I/O performance and scalability problems present in current file systems.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 280–287, 2005.

2 Background

To simplify data management, most distributed applications make use of shared storage wherein the data is accessible ‘on-demand’ by each entity in the system. The shared-storage solution however must satisfy the following requirements posed by applications:

High Concurrency. The storage system must scale to accommodate thousands of concurrent sessions.

High Aggregate I/O. Although the per-node I/O requirements may be modest (tens of MB/s), the aggregate throughput between the source and the storage system can be very high – 10 GB/s in current applications is not unusual. Throughputs in excess of 100 GB/s will be required to support peta-scale computing applications.

Balanced Scaling. As applications grow in complexity – through increased data acquisition capabilities, higher fidelity models, and more aggressive processing timeframes, larger datasets accompany the increased computational capabilities. This requires that storage systems scale in both capacity and I/O in a balanced fashion with data-generation capabilities.

Scalable Management. As these storage systems scale in capacity, it is critical that they be managed in a scalable fashion. This includes the ability to easily provision additional storage as capacity requirements increase, and to support automatic capacity and load distribution to best leverage the available capacity and throughput of the system.

Current shared-storage solutions such as SAN and NAS fail to meet one or more of these requirements. SANs were designed to provide a modest number of application servers with high performance, highly reliable access to a shared pool of storage devices – e.g. for enterprise transactional databases. SANs allow disks to be moved among application servers to readily address changes in capacity requirements, but lead to application-server-based islands of storage. NAS systems, on the other hand, afford widespread data sharing on heterogeneous platforms, with relatively modest per-user I/O requirements – e.g. for user home directory storage. Neither SAN nor NAS architectures support the aggressive concurrency and high per-client throughput requirements of high-speed distributed applications.

3 Existing Shared File System Storage Model

Most clusters run more than one application at a time and with technologies like processing virtualization the mapping of application to server is not fixed. To tackle this, the clusters deploy a shared storage solution that stores all the datasets and can be accessible to all the nodes of the cluster. For this many clusters use mountable distributed file systems such a Network File System (NFS), Andrew File System (NFS) and Distributed File System (DFS) and Shared File Systems. File systems such as Parallel

File Systems such as PFS and PVFS [4] and object-based cluster file systems as very popular among scientific community for high I/O throughput.

In Distributed File System or NAS model, the configurations make use of NAS devices get mounted onto the nodes using NFS. The major drawbacks of this model are – 1. NAS filer becomes the fundamental bottleneck. 2. NAS systems itself have traditionally suffered from scalability issues of the NAS filer.

The Shared File Systems perform well for configurations that have limited performance requirements and scalability needs.

3.1 Problems with Existing Shared File System Architecture

Following are the main problems with exiting shared files system architecture:

1. Not designed for parallel I/O, hence incurs high levels of blocking in aggressive read/write scenarios.
2. Management of file meta-data adds considerable overhead to the server.
3. Meta-data management also limits the scalability of the configuration.
4. File level Locking is not possible.
5. Suffers from problems of lock ownership that require drastic fencing strategies such as STOMITH.
6. Fault tolerance is low.

The result is that such systems do not scale efficiently for systems that demand aggregate bandwidth in excess of 100 MB/s.

The scientific Community keeping applications performance in mind has designed Parallel File Systems. They typically have a 2-tier architecture with I/O servers decoupling meta-data management from the application server cluster. A dedicated meta-data manager is used for handling file and directory namespace management and maintaining the file-stripe mapping for every file apart from the administrative functions.

A common I/O architecture used comprises PFS (or PVFS) at compute nodes with I/O Servers running a Shared File System over FC or iSCSI SAN storage. This system is illustrated in Figure 1.

1. Does not support direct parallel I/O.
2. Limited by the speed of the RAID controller or the I/O server, which come in between direct data paths.
3. Additional Meta-Data Manager used to maintain File-stripe mapping creates overhead in file access. In read/write configurations, every file access requires the compute node to query the meta-data manager for the file-stripe mapping in order to contact the correct I/O server.
4. Incur high-level of blocking when a large number of source nodes are performing a simultaneous read-write. Cannot support aggressive concurrency.

The result is that such systems do not scale efficiently beyond 80-100-node configuration, which demand aggregate bandwidth in excess of 1GB/s.

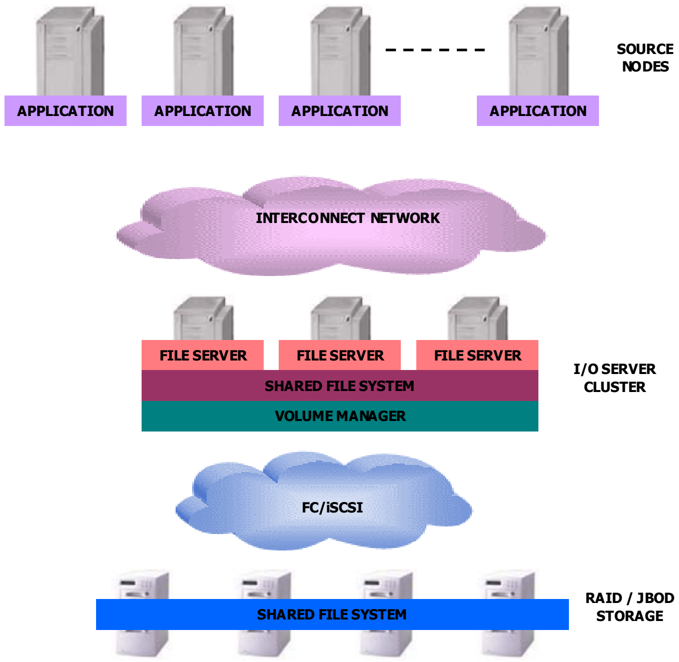


Fig. 1. Parallel File System Architecture

4 GridFS

GridFS is a next generation I/O solution that can scale to tens or thousands of nodes and petabytes of storage with groundbreaking I/O and metadata throughput. It is based on the Object based Storage Architecture (OSA) model and goes a step further to eliminate runtime file access overheads as compared to other upcoming OSA implementations. GridFS has multiple advantages over earlier distributed file systems such as AFS [5], Coda [6], InterMezzo [7], GFS [8], and GPFS [9].

4.1 Object-Based Storage Architecture (OSA)

The Object based Storage architecture combines the best properties of SAN and NAS storage environments. Like NAS, it allows for a file-level abstraction that allows secure data-sharing across various platforms and like SAN, it allows for fast, block-level, scalable access to the shared data. Figure 2 illustrates the storage topology.

OSA solutions contain split the storage functionality into two entities:

Object-Based Storage Devices (OSD). An OSD is an intelligent evolution of existing RAID/JBOD storage devices and has the following distinguishing features:

- Data stored in form of objects. An object encapsulates user data with its attributes.
- Handles inode and block layout management for stored objects internally.
- Handles QoS and access permissions for stored object.
- Includes local processing capabilities and local memory for data and attribute caching.
- Each OSD contains disk storage of capacity 150-500 GB.

Meta-data Servers. They provide file-system services to clients and can have following functions:

- File-directory membership, namespace management.
- File ownership and permission management.
- Provide NFS/CIFS Interface to LAN/WAN clients.

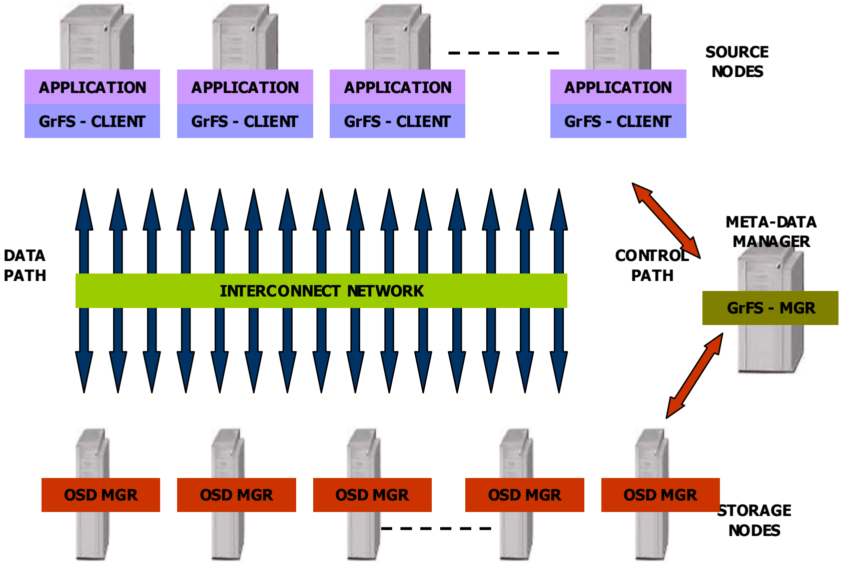


Fig. 2. GridFS Storage Topology

5 Two Core Advantages of GridFS

1. Direct Parallel I/O. Client nodes can now directly access storage without the I/O Server or the RAID controller bottleneck found in traditional systems. This enables building a massively parallel I/O storage system with extremely high aggregate I/O throughput rates.

Figure 3 compares a Store/Retrieve scenario in a Traditional Shared Storage with GridFS Object based Storage:

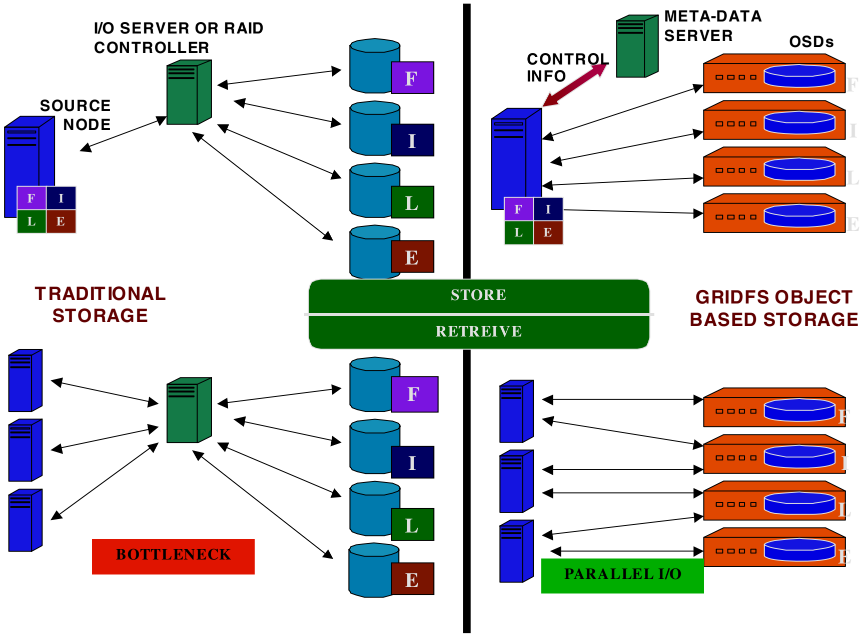


Fig. 3. Tradition Storage Vs GridFS Object Based Storage

2. In 'Write' Intensive Environment File Accesses Are 100% More Efficient. GridFS employs unique models that completely eliminate the need for maintaining a file-object mapping. GridFS incorporates advanced algorithms that eliminate most of the overheads in access to files spread across in a clustered storage configuration. The model improves the efficiency of every file I/O operation by 100% in the most 'write' intensive scenarios i.e., file access becomes twice as fast in GridFS as compared with existing cluster and shared file systems.

3. Dynamic Adaptability. GridFS has been designed to tune the access mechanism to suit the I/O pattern of the application giving faster application performance. For instance if scientific application processing enters a write-intensive stage and the application writes/appends data across 1000's of files, GridFS quickly adapts to this new access pattern and keeps access latencies (caused by metadata processing and locking) at a minimum.

6 Other Benefits of GridFS

1. Distributed Cache-coherency Management. Cache-coherency responsibility is given to the storage node storing the object. Since each storage node is responsible for the file-objects it houses, the overall responsibility of coherency of the data is distributed across storage devices.

2. Automatic Capacity Management. GridFS ensures that the load is balanced across each storage device and no single disk is filled up. Depending upon the configuration, a new storage device added to the system automatically relieves the heaviest loaded device with part of its objects so as to balance the load. Additionally if a storage device goes offline for some reason, data present on it is reconstructed on another storage device (similar to RAID 5 parity reconstruction). In other models, cache-coherency and capacity management are centered at the meta-data server and hence do not scale well.

3. Lower Costs. Within GridFS, both meta-data nodes and I/O nodes are low-cost 1U servers with inbuilt S-ATA disk storage or with attached RAID storage. Consequently, the cost of the system is much lower than scaling a typical SAN based storage solution.

7 Conclusion

By combing direct parallel I/O with a upto two times improvement in file I/O, GridFS can provide the fastest I/O performance needed by Application Clusters. Without the meta-data server bottleneck, the GridFS solution can more efficiently take to petabytes of storage, hence its highly scalable. By eliminating the need for file-object mapping and its related information, meta-data management in GridFS is reduced to providing only directory namespace and security management for the data maintained in the system. Instead of a high-availability cluster configuration of high-end machines for meta-data control, a simple 2-node fail over configuration of mid-sized machines can suffice.

GridFS is designed for robust solution with better fault-tolerance and resilience. It automatically handles failures of individual OSD devices with immediate recovery. Simplified policy implementation such as Hierarchical Storage Management (HSM), data backup and archival. GridFS allows for setting automatic polices where data stored or not accessed for a configurable period of time is automatically migrated to secondary storage areas or backup devices like Tape.

References

1. Top 500 supercomputers. http://www.top500.org
2. May, John M., Parallel I/O for High Performance Computing, Morgan Kaufmann Publishers, (2001)
3. Thakur, R., Gropp, W., Lusk, E., On implementing MPI-IO portably and with high performance, Procs of Workshop on I/O in Parallel and Distributed Systems. (1999) 23-32
4. PVFS: www.parl.clemson.edu/pvfs/
5. J. H. Howard. An overview of the andrew file system. In In Proceedings of the USENIX Winter Technical Conference, 1988.
6. J. J. Kistler and M. Satyanarayanan. Disconnected operation in the coda file system. In Thirteenth ACM Symposium on Operating Systems Principles, volume 25.5, pages 213–225, Asilomar Conference Center, Pacific Grove, U.S., 1991. ACM Press.
7. http://www.inter-mezzo.org

8. Steven R. Soltis, Thomas M. Ruwart, and Matthew T. O ´ Keefe. The global file system. In Proceedings of the Fifth NASA Goddard Conference on Mass Storage Systems, College Park, MD, 1996. IEEE Computer Society Press.
9. Frank Schmuck and Roger Haskin. GPFS: A Shared-Disk File System for Large Computing Clusters, In Proceedings of the Conference on File and Storage Technologies (FAST'02), 28–30 January 2002, Monterey, CA, pp. 231–244. (USENIX, Berkeley, CA.)
10. SCSI Object-Based Storage Device Commands (OSD), T10 (http://www.t10.org) Working Draft, Date: 2004/02/19, Rev: 09

Improving Customer Experience via Text Mining

Choudur Lakshminarayan, Qingfeng Yu, and Alan Benson

Global Operations and Information Technology, Hewlett-Packard Company
{choudur.lakshminarayan, qingfeng.yu, alan.benson}@hp.com

Abstract. Improving customer experience on company web sites is an important aspect of maintaining a competitive edge in the technology industry. To better understand customer behavior, e-commerce sites provide online surveys for individual web site visitors to record their feedback with site performance. This paper describes some areas where text mining appears to have useful applications. For comments from web site visitors, we implemented automated analysis to discover emerging problems on the web site using clustering methods and furthermore devised procedures to assign comments to pre-defined categories using statistical classification. Statistical clustering was based on a Gaussian mixture model and hierarchical clustering to uncover new issues related to customer care-abouts. Statistical classification of comments was studied extensively by applying a variety of popular algorithms. We benchmarked their performance and make some recommendations based on our evaluations.

1 Introduction

Analysis of textual data known as text mining has assumed tremendous importance due to the realization that there is a plethora of untapped information that can be leveraged for an organization's advantage. Textual information is referred to variously as unstructured data, semi-structured non-standard data etc. Some rich sources of text data is the free form customer comments available from on-line surveys. On-line survey is a medium by which a company gathers information about customer experiences, proclivities and sentiments. Although much effort is expended in gathering the information, it is largely unread due to limited resources of personnel and time as well as sheer monotony of reading. It is however desirable to harness the power of the rapidly evolving power of text mining technologies to glean salient content from the customer comments.

In the brief history of the Internet, customers' attitudes toward company web sites have gone from considering the sites a curiosity to finding the sites invaluable in terms of expected resources. This attitude shift is particularly true for technology companies, where we have increasingly observed customers expecting an easy-to-use site. At the same time, competitors' sites have increased in quality. Continuously improving customers' experience with the site through customer research is essential to stay competitive in the technology industry. Recognizing the importance of an outstanding customer experience, we take two approaches to track customer experience.

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 288–299, 2005.

One way is to provide an online survey that produces stable, quantitative, satisfaction ratings over time. Online survey data including demographic data, satisfaction ratings, and comments have been available to web site owners. In general, customers are not shy about describing the problems that they find with web sites. These comments generate a wealth of knowledge that flows into the online survey continuously, but this rich source of customer experience has not been well utilized for two reasons.

1. The number of comments can be overwhelming. Without some facility for sorting the comments based on a problem with a specific web site, the process of reading comments can be tedious and haphazard. Cluster analysis may solve this problem.
2. Comments are sometimes difficult to interpret without knowing the pages the customer visited before completing the survey. We believe that integrating comments data with click stream data would address this problem.

The goal of this paper is to present some ways in which text mining can be utilized to extract valuable information from the untapped textual information resources. We focus mainly on analysis of on-line customer comments and finding similarities in documents via text matching. Secondarily, we study the performance a variety of classification procedures and recommend that the support vector machine or the naive Bayes classifier technique is viable for text categorization.

On-line customer comments are mined using statistical clustering methods. Clustering performed periodically reveals new customer care-abouts that can be addressed by web site management. Another way to exploit comment data is by statistical classification. In this approach new in-coming comments are assigned to one of a set of pre-defined set of categories. This methodology allows the web site owners to track rate of change of customer sentiments about specific issues.

The paper is organized as following. In sections 2, we discuss the data characteristics and provide some examples of typical text customer comments and the steps involved in pre-processing text prior to analysis. In Sections 3, we discuss in detail clustering and classification and present some results. In Section 4, we present our conclusions and discuss future directions for research.

2 Data

The data used for analysis is the free-form text from customer comments. Randomly chosen visitors to a web-site are delivered a survey form that captures information about their experience via a set of questions with corresponding ratings on a well defined ordinal scale. Also, customers are allowed to express their experience in a section of the on-line survey. Customer comments tend to range between very terse to very prolix. We removed comments that were less than 25 characters in length. Customer comments give web-site management valuable insight into the concerns of a customer that is used to improve the customer's experience. Although this is a rich source of information, the data is often very dirty in that its replete with typos, obscure characters, and tends to be conversational. In the next paragraph and ensuing sections, we will present in detail our analysis using statistical clustering and classification.

2.1 Challenges from Text Comments

Comprehensibility – Comments are sometimes difficult to understand.

- "I LOOKED FOR LEPTOP COMPUTER NOT FOR BUISSINESS, FOR PRIVATE HOME NEEDS AND I DID NOT FIND" (sic)
- "my cuputer freezes up .it would not run right most of the time . error messages, defeacted microsoft products in my computer. two since i have owened it. i have several case numbers with you people .and my computer is only like 2 years old now, but when i firt started having problems was just shortly after a year of use . and still experience problems.i was told by your people it was my tuff luck i should of taken out a extended warranty. please a year afer you buy something should not go crazy after one year . it should last longer than that .i have had nothing but problems with your computer and your products.not to mention your people you have working for you" (sic)

Relevance – Some comments focus on problems customers are having that are not directly related to the HP.com website

- "I want to get on the internet. I don't want hp advertising. How do I get on the internet and rid of hp advertising?"
- "You guys suck, learn to make a good product!!!"
- Specificity – Some comments do not include enough information to diagnose a specific problem
- "The looks. Not intuitive enaugh." (sic)
- "info was very difficult to find"
- "It seems difficult to find and buy items."

Multiple themes – some comments address more than a single issue relating to web site

- "There are many levels (in other words clicks and keep going down the hierarchy). For eaxmple: Start from main hp page and look for extended battery for IPaq 4155. It is not there in Home... but available in Small business. Why not ???? It's the same battery I believe for 4150 and 4155. Secondly - goto the spooping cart page and hit refresh, it keeps adding the qty (something very basic to handle for a ecommerce site). After that try calculate shipping - select both state and zip (although only one is required as mentioned) but the page crashes there. Your site need some work" (sic)
- "Looking for Memory upgrade for Designjet 1050c. I had to make several nonintuitive clicks (and go back) to find the memory selector option. Plus, the ""1-2-3"" interface for memory didn't function properly for printers. The options for 2 and 3 remained blank. For reference, I'm using IE 5 on Windows 2000 PC." (sic)

Idiosyncratic language – some comments use infrequently used terms

- "Too many channels, or, pages to arrive (finally) what I'm really looking for.. It should be simplified somehow.. I know it is difficult to address everyones' problem, but at the start "oranges" must be seperated from "apples", mean; seperate the one "konws a little" than those totally green." (sic)

- "Look have someone in your family who doesn't work at hp try to buy one. What I want is the best portable pc that has a wireless capabilities with good video and can act as a tv. why do you make it difficult. Tell us what the difference is on ram and how it will effect our usage. get smart and don't quit being a bunch of smarty pants. You know, you should design your web site in Iowa, where their isn't an over abundence of Stanford jerks who think they know it all. I am buying a sony today but i really wanted and HP" (sic)

2.2 Data Pre-processing

Our first step for both techniques was to preprocess data in order to reduce noise and increase signal. Preprocessing included 1. Creating synonym lists (e.g., computer=PC, notebook=laptop) 2. We Removed low-value terms such as prepositions, articles, adverbs and company names (e.g., HP, DELL) via a stop list. It is well known that 20-30% of the documents contain stop-words. 3. We applied spell checker to all the comments in our dataset to correct miss-spellings. 4. We linguistically parsed words to perform parts-of-speech tagging. 5. Finally, we removed funny characters, tabs and such to clean up the data. Furthermore, we weighted the terms in the documents using the popular tfidf weighting scheme [9] as one would in any information-retrieval indexing endeavor. The tfidf weighting is a product of term-frequency and the inverse document frequency. In our original dataset over 7,000 different terms appeared in the comments. To increase the tractability of the data, we applied latent semantic indexing [5] for term space reduction (TSR). There are multiple approaches to achieve term space reduction such as roll-up terms. In this approach, the first 100 or 200 heavily weighted terms are utilized. This approach was first introduced by [9] and studied later in systematic detail by [15]. We used latent semantic indexing (LSI) for term parsimony, some linguistics related issues and to lessen computational burden. LSI is a method used for dimensionality reduction developed to address problems emanating due to synonymy and polysemy. Polysemy is a condition where the meaning of a word is context dependent. The word "bank" is an example. It may refer to a river bank or a financial institution depending on the situation. This is particularly sticky because unrelated documents tend to get clustered together because the terms themselves, but not their import is used to characterize them. Synonymy relates to a condition where two terms "retire" and "sunset" in a particular parlance convey the meaning that a product or an application is discontinued. However documents with these terms tend to be unrelated although there is a latent semantic relationship between them. LSI compresses document vectors into other vectors of a lower dimension. The lower dimensional vectors are obtained as linear combinations of terms in the original vectors with the property that co-occurring terms are mapped into the same lower dimensional vector. Implementationally, LSI is achieved by applying singular value decomposition ([7], [13]). Singular Value Decomposition (SVD) applied to our data set reduced the number of unique terms from 7,000 to the 250 most important terms. Thus for each comment in our dataset, we had 250 variables that represented the most discriminating terms. Data reduction to 250 salient features captures about 80% of the variation in the data. It has been reported in the literature [10] that LSI yields better performance as opposed to feature selection involving top x number of terms.

3 On-line Customer Comment Analysis

We employ two types of analyses for the comment data. Statistical classification is an algorithm trained to learn features endemic to certain pre-defined categories. Statistical clustering, wherein the algorithm decomposes the data set into natural groupings based on the textual similarities in the comments. Both analyses are based on vector-space models also known as bag-of-words models. As the name implies, vector-space modeling approach ignores the semantic content in the data. In vector-space based approach, each document is expressed as a vector of terms/words which are weighted by a suitable weighing scheme (see section 2.2). Note that the data vector is invariant to the ordering of the terms.

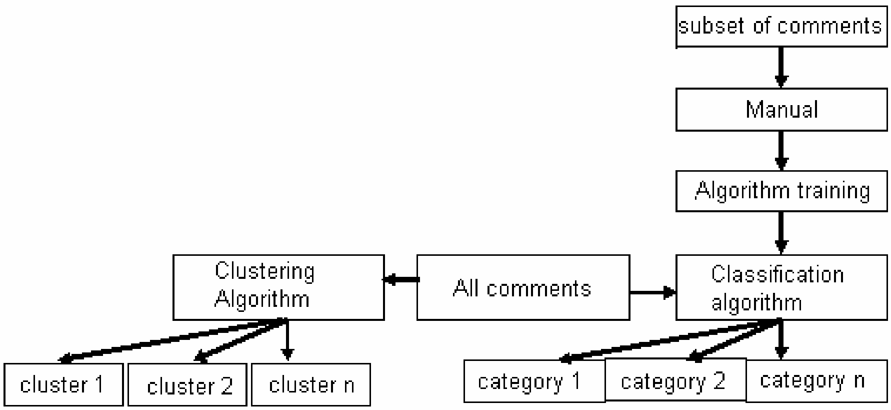


Fig. 1. Process differences between clustering and classification methods

Both clustering and classification require some human intervention as shown in Fig. 1. Successful implementation of semi-automated analysis of text requires the interaction of a domain expert (knowledgeable about the business) and a technical expert (knowledgeable about the techniques involved for text analysis). Statistical classification requires a training data set as a pre-requisite. Training data comprises of pre- defined comment categories. Each category hosts a sample of n comments relating to a specific issue. For clustering, human interaction is needed in order to divine the content of the comments clustered together. A schematic of the clustering or classification of a corpus of documents is given in Fig. 1.

3.1 Statistical Clustering of Customer Comments

We used statistical clustering to segment customer comments into multiple groups. Clustering was implemented with the hope that comments will split into homogeneous groups of comments, each group pointing to a theme regarding customer experiences and care abouts. Gaussian mixture modeling [18] technique was implemented to produce the clusters. Briefly Gaussian mixture modeling (GMM) is a model based method. It assumes a probability distribution for the observed data. The probability distribution function (pdf) of an observed vector x is given as:

$$f\left(x \mid \underset{\sim}{\theta}\right)=\sum_{j=1}^{c} f\left(x \mid c_j,\theta_j\right)\pi_j \tag{3.1}$$

Where θ is a vector of unknown parameters of the mixture density, π_j, $j = 1,2,,....,c$ are known as mixing proportions or prior probabilities of the j clusters and c_j denotes the j^{th} class. Under this formulation, each $f\left(x_i \mid c_j,\theta_j\right)$ is modeled by a specific probability density function (in this application it is the normal distribution). The Gaussian parameters and the mixing proportions are estimated using the famous E-M algorithm ([3], [6]). In short, the E-M algorithm maximizes the expectation of a log-likelihood function instead of the plain likelihood function of the data which is intractable due to the formulation of the data density as a sum of component probability distributions as in equation 3.1.

3.1.1 Comments Clustering Results

Clustering was implemented using the text miner (TM) node with SAS software. We used hierarchical clustering as well, but output from GMM yielded interpretable as well as actionable results. Our methodology included clustering data on a monthly basis. On the average there were approximately 2000 comments from customers per month. Initial customer comment clustering results are encouraging thus far, and are summarized in Table 1. Two positive trends are seen in these data:

1. Clustering identifies stable long term customer experience issues. The first five problem areas shown are stable problems that were not systematically addressed over seven months. The clustering procedure successfully identified of the five stable problems over seven months, only missing the printer supplies problem in May.
2. New and unknown problem clusters are identified. The "Pocket PC upgrade problem" and the "subscription login problem" were identified The existence of the pocket PC upgrade issue and subscription function login problems have been verified, and had not been identified before.

Limitations of this methodology include the need for large sample sizes to achieve stable results. Customer experience issues could be identified more quickly if either the sample size needed for stable results could be reduced (so clustering can performed over data collected in shorter time intervals) or more comments were collected.

3.1.2 Entropy Results

To aid in the comment clusters interpretation we calculated Entropy [4] values for each combination of web site areas and problems identified by the clustering methodology described above. Entropy measures the amount of disorder in a set; in this case the set was the web pages associated with a specific web area and problem. Site/problem combinations with low entropy values reflect problems that are concen-

Table 1. Clustering Results over Time (including cluster keywords and percentage of total)

Cluster description	April	May	June	July	August	September	October
Printer supplies related problems	cartridge, printer, print, color, hp (5%)		color, print, printer, hp, accessory (7%)	cartridge, color, print, printer, hp (7%)	cartridge, color, printer, print, model (7%)	Printer, cartridge, print, list, hp (7%)	color, print, printer, item, give (4%)
Part Number and search problems	numb, part number, part, list, search (6%)	engine, numb, part, search, list (13%)	engine, search, specific , result, (4%)	numb, part number, part, compaq, model (4%)	numb, part number, part, order, compaq (5%)	numb, part number, part, spare, compaq (5%)	numb, part, model, find, specifications (4%)
Driver download problems	download, update, unix software, driver (7%)	download, driver, printer, software, update (10%)	download, driver, patch, update, software (6%)	download, driver, web site, web, hp (18%)	download, driver, update, software (6%)	download, link, driver, page, appear (5%)	download, driver, software, unix, update (9%)
Server configuration problems	server, customize, storage, configure, price (7%)	information, server, find, customize, look (10%)	server, configure, option, model, technique (5%)	server, configure, buy, hard, find (6%)	server, info, storage, customize, configure, (5%)	Server, web site, web, software, unix (14%)	product, server, price, detail, information (12%)
Page rendering performance problems	load, page, rebate, slow, time (15%)	Load, page, slow, time, long (11%)	load, page, slow, time, long (10%)	load, page, slow, time (9%)	load, page, slow, time, long (8%)	load, page, slow, time, appear (9%)	load, long, page, slow, time (10%)
PocketPC upgrade problems			link, pda, purchase, upgrade, order (6%)	link, pda, purchase, upgrade, software (6%)			
Subscription login problems							address, change, email, log, login (14%)

trated on a few pages. Low entropy problems tend to be much more actionable because they reflect specific content or functionality that customers find difficult. Problems with higher entropy values tend to be vague and less actionable. The entropy function is calculated using the formula:

$$Entropy(c_j) = -\sum_{i=1}^{n_j} p_i \log(p_i), i = 1,2,....,n_j \qquad (3.2)$$

Where $p_i = \frac{\#(pages \quad with \quad specific \quad site/problem)}{n_j}$ and c_j is the symbol for the j^{th} cluster.

3.1.3 Satisfaction and Conversion Rate Modeling

Customer comment data allows the modeling of the impact of specific problems in terms of overall satisfaction levels. Because satisfaction goals are reported to top management, identifying specific problem areas is valuable for prioritizing resources to be applied to customer experience problems. Satisfaction impact values shown in Table 2 represent the percent decline in overall satisfaction for the Enterprise segment of the customer website. In this case, customers' inability to find drivers lowered the area's satisfaction scores by 4%. The total influence of all problem clusters on the enterprise site is a 14% reduction in satisfaction.

Table 2. Entropy and satisfaction projections for one specific web area (Enterprise segment)

	Printer supplies related problems	Server configuration problems	Driver download problems	Part Number and search problems	Page rendering performance problems	Sub-scription login problems	Total (includes other problems not shown)
Entropy	2.89	3.91	4.76	3.24	4.09	3.72	
Satisfac-tion impact	0%	-1%	-4%	0%	-1%	-2%	-14%

We also attempted to model purchase conversion rates based on comment clusters. However, because of the relatively small numbers of customers who purchase products online, we were not able to achieve stable results.

3.2 Text Categorization and Customer Comments

Statistical classification or semi-automated text categorization has a long history dating back to the middle of the 20th century. Early text categorization involved manually building a set of rules encoding expert knowledge on how to categorize documents [20]. During that period, the interest in text categorization (TC) at best was tepid and half-hearted. Late in the 20th century (~90's) due to the extraordinary growth in documents (web pages)-thanks to the internet- TC received a shot in the arm, and an approach based on machine learning [14] became the de facto analytical method. In this approach, a set of pre-classified documents known as training data is utilized to train an algorithm (usually statistical in nature). The algorithm is a decision rule that allows the assignment of a future document into one of the pre-defined classes. The decision rule is derived based on an optimality criterion. In this paper we took the machine learning approach to text categorization. Among the methods used are the parametric linear discriminant analysis (LDA), the non-parametric discriminant analysis using Parzen windows [1], The k-nearest neighbor (k-NN) [18], Naïve Bayes classifier [14], and Support vector machines (SVM) [16]. We also used the neural networks classifier (ANN), but due to its poor performance did not include in the comparative study. We considered using the Rocchio classifier ([10],[20]). The Rocchio classifier uses the cosine transformation [21] to compute similarity between a prototype vector (based on an average) and an incoming document vector. As it depends on an average (and the average is only a smoothed version of the data set), it suffers from significant disadvantages.

3.3 Comparison of Classifiers Used in Categorization

We benchmarked classifier performance using five different statistical classifiers to automatically classify comments based on pre-categorized comments. We used the proc discrim routine available in SAS/STAT [22]. We coded 2,000 example comments into 1 of 8 comment categories; we used these comments for five different classification algorithms. The corpus of 2000 comments/documents was split into training, testing, and validation sets. The break up was 70%, 15% each respectively for the three sets. Furthermore, we used the k–fold cross-validation method for evaluating classifier performance. Accuracy in the best case was 70% due to Parzen windows as shown in Table 3. From a business perspective, 70% accuracy would not be high enough to facilitate usage of comment classifications by web site owners. To improve this rate, we either need to lower the number of comment categories, thereby making them less useful, or we need to increase the number of training examples used, thereby increasing the human resources required. While the non-parametric Parzen windows method yielded the best performance, it was subject to wild variations over data sets produced by changing the percent distribution of sample comments across the training, test, and validation sets. This suggested that we apply bagging [11] which is a bootstrap [8] aggregation of classifiers performances over several randomly sampled training sets. As the error-rate of 30% is rather high, it appears boosting [17] may yield reduced misclassification rates. However, we are aware that if the classifier is close to the limits of accuracy attainable for the data set, no amount of boosting will do anything to improve performance. As a next step, we intend to implement bagging and boosting to stabilize and increase performance of the classifiers.

The 70% accuracy achieved by the Parzen window was by using a tri-weight kernel and selecting a value for the control-parameter *r* in the interval [4, 5]. The tri-weight kernel is given by:

$$K_x(x) = \begin{cases} c_3(t)\left(1 - \frac{1}{r^2} x^T V_x^{-1} x\right)^3 & if \quad x^T V_x^{-1} x \le r^2 \\ 0 \quad elsewhere \end{cases} \qquad (3.3)$$

where $c_3(t) = (1 + p/6)(1 + p/4)\frac{1}{v_r(x)}(1 + p/2)$, V_x is the pooled covariance matrix, $v_r(x) = r^p |V_x|^{1/2} \frac{\pi^p}{\Gamma(p/2+1)}$, and $\Gamma(.)$ is a Gamma function ([2]).

The *k-NN* classifier produced the lowest accuracy approximately equal to 61%. This performance was achieved by selecting the nearest neighbors in the range between [30, 40]. Fisher's LDA is a classifier based on normal theory. The data vector *d* is assumed to follow a multivariate normal distribution and a new incoming document is assigned to that class for which the posterior probability $p(c_j \mid d), j = 1,2,...,k$ is the maximum. The document vector is not necessarily Gaussian, but we observe that LDA is quite robust against departures from normality and it invariably finds the class for which the distance between the new document vector *d* and the class represented by the prototype mean vector is the smallest. The SVM classification scheme gave the best consistent performance. The accuracy of the SVM classifier was in the 68%-70% range. All the techniques considered heretofore are based on using the SVD dimensions as the canonical inputs. In the NBC setting, the input data vector was simply a string of word tokens. This classifier computes the conditional probability of a class given a document vector *d* by using the Bayes theorem. The naivete in the classifier comes from the assumption that the terms in a document are conditionally independent within a class. The assumption is clearly incorrect in most practical situations. In a vector space model setting where the order of the terms is ignored, it turns out that such a naïve assumption is not detrimental. In practice, the classifier performs fairly well, although conditional independence does not necessarily hold [12]. The conditional probability of a class given a document *d* is given by:

$$p(c_j \mid d, \theta_j) = \frac{p(d \mid c_j, \theta_j) * p(c_j \mid \theta)}{p(d \mid \theta)} \propto p(d \mid c_j, \theta_j) * p(c_j \mid \theta) \qquad (3.4)$$

Where θ_j is a class conditional parameter vector of the component probability density function, 1and $j = 1,2,...,k$.

Table 3. Accuracy of classification algorithms

Classifiers	Accuracy (%correctly classified)
Parzen Window: Procedure that uses smoothly tapering window function to trap similar comments and measure similarity relative to a candidate comment within a user defined radius	70%
Naïve Bayes Classifier: Assumes independence among terms in the comments and maximizes the probability of joint word occurrences within a category	69%
Fisher Discriminant Analysis: Uses a modification of the Euclidean distance to measure similarity between comments across categories	64%
K-NN (nearest neighbors): Finds the category that is most similar to the candidate comment to classify based on a majority voting rule	61%
Support Vector Machine: Constructing hyperplanes in a multidimensional space that separates cases of different class labels.	69%

Upon applying a variety of classifiers to our data set, we believe that the SVM classifier is the most stable. The classifier performance may itself be improved by using such a methodology as Adaboost. M1 [19]. The naive Bayes classifier also produces comparable performance. The choice of the classifier is therefore between SVM and NBC.

While classification has the advantage of summarizing large sets of comments into agreed upon comment categories, it has no way of identifying new problem categories. From a business perspective, this is important because customer satisfaction or dissatisfaction tends to be related to issues with the web site (see Table 2). Because number of issues experienced by customers tends to be very large, manual categorization is a gargantuan effort to get enough number of categorized exemplars.

4 Conclusions

This paper is an attempt to demonstrate how text mining technologies can be used effectively for constantly driving business improvement and improving customer satisfaction. Furthermore, our comparative study involving many classification procedures provided insight about the classifier performance. Our conclusion is that the support vector machine classifier and the Naive Bayes classifier are clear winners. Statistical clustering so far appears to be most powerful method to discover new issues with the web site. Our observation is that clustering based on Gaussian mixture models produced meaningful and actionable results as opposed to hierarchical clustering schemes. Presently we are studying the application of bagging and boosting methods to improve categorization. Text mining as it is applied presently is a semi-automated process which means that it requires considerable human labor. Our work is now limited to approaching text mining from a statistical perspective revolving around vector space models.

References

1. Parzen, E., On Estimation of a probability density function and mode. Annals of Mathematical Statistics, 33(3):1065-1076, 1962
2. Abramowitz, Milton, Stegun, Irene, A. Handbook of Mathematical functions, Dover, 1972
3. Dempster, A., Laird, N. and Rubin, D. Maximum likelihood from incomplete data via the EM algorithm(with discussion), Journal of the Royal Statistical Society, B. 39: 1-38, 1977
4. Berger, James, A, Statistical Decision Theory and Bayesian Analysis, 2nd edition, Springer-Verlag, 1985
5. Deerswater, S, Dumais, S.T., Furnas, G.W., Landauer, T.K., and Harshman, R. Indexing by Latent Symantic Indexing. Journal of American society for information science, 41(6):391-407, 1990
6. McLachlan, G.J., Discriminant Analysis and Statistical Pattern Recognition, New York: John Wiley & Sons, 1992
7. Johnson, R.A. and D.W. Wichern, Applied Multivariate Statistical Analysis, 3rd edition, Englewood Cliffs, New Jersey: Prentice Hall, 1992
8. Effron, B. and Tibshirani, R. An introduction to the Bootstrap, Chapman and Hall, 1993
9. Apte, C., Damerau, F., and Weiss, S. Automated learning of decision rules for text categorization. ACM transactions on Information Sciences, 12(3): 233-251, 1994

10. Schutze, H., Hull, D.A., and Pederson, J.O. A comparison of classifiers and document representations for the routing problem. Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval, 229-237, 1995
11. Breiman, L., Bagging Predictors, Machine Learning, Vol. 24, No. 2, pp. 123-140. 1996
12. Domingo, P. and Pazzani M. On the optimality of the simple Bayesian classifier under zero-one loss, Machine Learning 29, 103-130, 1997
13. Letsche, T.A. and Berry M.W., Large scale information retrieval with latent semantic indexing, Information sciences, 100, 105-137, 1997
14. Mitchell, T., Machine Leaning, John Wiley, 1997
15. Yang, Y. and Pederson J.O. A comparative study on feature selection in text categorization. Proceedings of the 14th International conference on Machine Learning, 412-420, 1997
16. Vapnik, Vladimir, N. Statistical Learning Theory, Wiley Inter-science, 1998
17. Schapire, R.E., and Freund, Y. Boostexter: a boosting based system for text categorization. Machine Learning 39, 135-168. 2000
18. Duda, R.O., Hart, P.E. and Stork, D.G., Pattern Classification, New York: John Wiley Interscience, 2001
19. Tibshirani, R., Hastie, T. and Friedman, J. The Elements of Statistical Learning Data mining, Inference, and Prediction, Springer, 2001
20. Sebastiani, F. Machine learning in automated text categorization. ACM Comput. Surv. 34, 1-47, 2002
21. Baldi, P., Frasconi, P., and Smyth, P. Modeling the Internet and the Web. Probabilistic Methods and Algorithms, John Wiley, 2003
22. SAS/STAT 9 and 9.1 User's Guide, Copyright © 2003 by SAS Institute Inc., Cary, NC, USA

A Distributed Algorithm for Outlier Detection in a Large Database

Biplab Kumer Sarker[1] and Hiroyuki Kitagawa[1,2]

[1] Graduate School of Systems and Information Engineering
[2] Center for Computational Sciences,
University of Tsukuba, Tsukuba, Japan
{bksarker, kitagawa}@kde.is.tsukuba.ac.jp

Abstract. This paper proposes a distributed algorithm to detect outliers for large and distributed datasets. The algorithm employs the basis of distance-based outliers based on the distance of a point to its k^{th} nearest neighbor. It declares the top n points in the ranking to be outliers. To the best of our knowledge, this is the first proposal of a distributed algorithm for outlier detection for shared-nothing multiple processor computing environments. It has four phases. First, in each processing node, the algorithm partitions the input data set into disjoint subsets, then it prunes entire partitions as soon as it is determined that they cannot contain outliers. Then it applies a global filtering technique to collect the partitions as global candidates from local candidate partitions in each processing node. Further, it introduces a load balancing algorithm to balance the number of local candidate partitions. Finally, it identifies outliers from each processing node.

1 Introduction

Data mining and knowledge discovery (KDD) has been the emerging research area for the past several years. The typical mining tasks generally include discovering associations, sequences, decision tree classification and clustering. Although perhaps the most common task of data mining has been to discover relationship in data, not much work has been done to find exceptions in the data.

Finding "outliers" or exceptions can be useful to detect credit card fraud, telephone calling card fraud, analysis of performance statistic of professional athletes, exploration of satellite and medical images and many more where the occurrence patterns that are exceptions may need special attention. Therefore, the outliers may point out surprising and suspicious activities, extreme or relatively extreme values or observations (or a subset of values/observations) which appear to be inconsistent with the remainder of the set of data.

In this paper, as a contribution we propose a distributed algorithm for detecting outliers on shared-nothing distributed systems. The algorithm is based on a distance-based outlier detection approach which finds the top n outliers using the distance of the k^{th} nearest neighbor for a point. To detect outliers we require huge computation of point wise distance based calculation. Therefore, it is a time consuming task. Keeping

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, pp. 300–309, 2005.

in mind to reduce the execution/running time of the whole process, we use the notion of distributed processing to distribute the task to various nodes of the system and processing them in parallel.

The paper is organized as follows. Section 2 presents a brief overview of related works. The problem of mining outliers based on distance based approach is defined in section 3. The proposed distributed algorithm and the detailed description of its each phase are discussed elaborately in section 4. Finally we conclude with future directions.

2 Related Works

The problem of detecting outliers has been studied widely in the statistical community [1]. The main drawback of the statistical approach is that the user might simply not have enough knowledge about the underlying data distribution. In order to overcome the problem, Knorr and Ng [2] first proposed distance based definition for outliers which is simple and intuitive. The demerit of the approach is that it requires the user to specify a distance *d* which could be difficult to determine. It also does not provide a ranking for the outliers and the cell-based algorithm proposed here is not suitable for the high dimensional data. To overcome these drawbacks a new definition for outliers and developing algorithms has been proposed [3]. In this definition, users do not need to require specifying the distance *d*; instead he/she has to specify the number of outliers *n* that he/she is interested in using the distance of the k^{th} neighbor of the n^{th} outlier to define the neighborhood distance *d*. The above works are considered as distance-based outlier detection approach.

The other approach that earns a lot attention on outlier detection is density-based approach [4]. This approach overcomes some problems of the distance based approach when the data set have both dense and sparse region [5]. There are some other approaches for outlier detection such as Distribution-based and Depth based approaches reported in some literatures. LOCI examines the "outlier-ness" of objects in neighborhoods of different scale [5]. These outlier detection schemes are more complicated than distance based approach.

The definition of outliers used in some clustering algorithms like CLARANS [6], DBSACAN [7], BIRCH [8] and CURE [9] are in a sense subjective and related to the clusters that are detected by these algorithms, which is in contrast to the definition of distance based outlier detection [3]. These algorithms consider outliers, but only to the point of ensuring that they do not interfere with the clustering process.

Our present work uses the definition of the distance-based outlier detection proposed in [3] and proposes a distributed algorithm for detecting outliers for shared nothing distributed systems. We choose distance based outlier detection approach for its great intuitive appeal. Due to its huge computation on calculating the distance of the k^{th} neighbor of the n^{th} outlier to define the neighborhood distance *d*, it is inevitable that employing parallel processing would be a reasonable approach for detecting outliers. As far as our knowledge this is the first attempt to propose such kind of algorithm for detecting outliers.

Table 1. List of symbols and notations

Symbol	Description
K	Number of neighbors of a point that we are interested in
$D^k(p)$	Distance of point p to its k^{th} nearest neighbor
n	Total number of outliers we are interested in
N	Total number of input points
MBR	Minimum Bounding Rectangle
δ	Dimensionality of the input data
Dist(p, q)	Distance between a pair of points (*p*, *q*)
MINDIST(p, R)	Minimum distance between a point and a MBR
MAXDIST(p, R)	Maximum distance between a point and a MBR
MINDIST(R, S)	Minimum distance between two MBRs
MAXDIST(R, S)	Maximum distance between two MBRs
minDkDist	The smallest value of D^k in each processing node
global_minDkDist	The smallest value of D^k in global filtering phase

3 Problem Definition

As given in [3], the definition of outlier is as follows:

Definition 3.1: *Given an input data set with N points, parameters n and k, a point p is a* $D^k{}_n$ *outlier if there are no more than n-1 points p′ such that* $D^k(p')> D^k(p)$.

In other words, the top n points in the descending order of D^k values are considered as outliers. These outliers are referred as the $D^k{}_n$. The definition basically uses the distance of the k^{th} neighbor of the n^{th} outlier to define the neighborhood distance *d*. The points are ranked according to their $D^k(p)$ distance, and the top n points in this ranking are considered to be outliers. To measure the distance between a pair of points Euclidean distance metrics are used.

3.1 Distances Between Points and MBRs

The key technical tool that we used in this paper is the approximation of a set of points using their Minimum Bounding Rectangle (MBR) [3]. Then, by computing lower and upper bounds on $D^k(p)$ for points in each MBR, it is possible to identify and prune entire MBRs that cannot possibly contain outliers. The computation of bounds for MBRs requires defining the minimum and maximum distance between two MBRs. Outliers detection is also aided by the computation of the *minimum* and *maximum* possible distance between a point and a MBR.

In this paper, we use the square of the Euclidean distance as the distance metric since it involves fewer and less expensive computations [3]. We denote the distance between two points p and q by *dist (p, q)*. Let us denote a point p in δ-dimensional space by $(p_1, p_2, \ldots, p_\delta)$ and a δ dimensional rectangle R by the two endpoints $r = (r_1, r_2, \ldots, r_\delta)$ and $r' = (r'_1, r'_2, \ldots, r'_\delta)$ of its major diagonal such that $r_i \leq r'_i$ for $1 \leq i \leq n$. Let us denote the minimum distance between point p and rectangle R by *MINDIST* (p, R). Every point in R is at a distance of at least *MINDIST* (p, R) from p.

Definition 3.2:

$$MINDIST(p,R) = \sum_{i=1}^{\delta} x_i^2, \; where$$

$$x_i = \begin{Bmatrix} r_i - p_i & if\ p_i < r_i \\ p_i - r'_i & if\ r'_i < p_i \\ 0 & otherwise \end{Bmatrix}$$

We denote the maximum distance between point p and rectangle R by *MAXDIST (p, R)*. That is, no point in R is at a distance that exceeds *MAXDIST* (p, R) from point p.

Definition 3.3:

$$MAXDIST(p,R) = \sum_{i=1}^{\delta} x_i^2, \; where$$

$$x_i = \begin{Bmatrix} r'_i - p_i & if\ p_i < \frac{r_i + r'_i}{2} \\ p_i - r_i & otherwise \end{Bmatrix}$$

Next, we define the minimum and maximum distances between two MBRs. Let R and S be two MBRs defined by the endpoints of their major diagonals (r, r' and s, s' respectively) as before. We denote the minimum distance between R and S by *MINDIST* (R, S). Every point in R is at a distance of at least *MINDIST* (R, S) from any point in S (vice-versa). Similarly, the maximum distance between R and S, denoted by *MAXDIST* (R, S) is defined.

Definition 3.4:

$$MINDIST(R,S) = \sum_{i=1}^{\delta} x_i^2, \quad where$$

$$x_i = \begin{Bmatrix} r_i - s'_i & if\ s'_i < r_i \\ s_i - r'_i & if\ r'_i < s_i \\ 0 & otherwise \end{Bmatrix}$$

Definition 3.5:

$$MAXDIST(R,S) = \sum_{i=1}^{\delta} x_i^2, \quad where\ x_i = \max\left\{ |s'_i - r_i|, |r'_i - s_i| \right\}$$

4 Distributed Outlier Detection

Our proposed distributed algorithm for outlier detection has the following phases:

1. Local Candidate Identification Phase (LCIP)
2. Global Filtering Phase(GFP)
3. Load Balancing Phase(LBP)
4. Outlier Detection Phase(ODP)

It is assumed that our working platform is a shared-nothing distributed system i.e. each processing node has its own processor and local memory and a local disk. The dataset is assumed to be distributed equally in size to the local disks of each node without overlapping. Communication between the processing nodes can be done using message passing.

We use manager-worker style paradigm suitable for a shared-nothing environment to solve our problem. In this paradigm, one process, called manager, is responsible for keeping track of assigned and unassigned data. It assigns tasks to other processes, called workers, and retrieves results back from them. We dedicate each process to each processing node (PN). The algorithms described below in each phase, termed with 'Algorithm_Manager' are to be performed by the processing node regarded as 'manager' and 'Algorithm_Worker' are to be performed by the processing nodes regarded as 'workers'. Now we briefly present the working principles and the algorithms of each phase.

4.1 Local Candidate Identification Phase (LCIP)

In this phase, considering that the dataset is distributed among all the processing nodes, we make use of the notion of partition based algorithm proposed in [3] to identify the possible local candidates i.e. the partitions containing points which are the candidates for outliers. The key idea is to prune the partitions as soon as it is determined that they cannot contain outliers. We perform the following steps locally for each processing node in parallel.

a) Generate partitions using the clustering algorithm (in this case BIRCH algorithm [8]) to cluster the data and treat each cluster as a separate partition. Let the total set of partitions be P_{set}.(Procedure *BIRCH*)

b) Compute bounds on D^k for points in each partition P in P_{set}, using the procedure *computeLowerUpper*. The idea behind this is to approximate a set of points by its MBR. Then, by computing lower and upper bounds on $D^k(p)$ for points in each MBR, we are able to identify and prune entire MBRs that cannot possibly contain D^k_n outliers. The input parameters for this procedure are the root of the index containing all the partitions and *minDkDist* which is a lower bound on D^k. (Procedure computeLowerUpper[3]).

c) Identify partitions that can potentially contain outliers, and prune the remaining partitions. If *P*.upper for a partition *P* is less than minDkDist, none of the points in *P* for which *P*.upper ≥ minDkDist are candidate partitions.(Procedure computeCandidatePartitions[3]).

4.2 Global Filtering Phase (GFP)

In this phase, we propose a new global filtering technique to determine the possible global candidate partitions from the local candidate partitions obtained in the previous phase that are located in each processing node. It has two sub-phases as follows.

a) Construction of Global Tree

Let us have the possible candidates $P_{candidates}$ in each processing node after the first phase mentioned above. So, if we have the x processing nodes in the system then the total candidates would be $P_{all\_cand} = \cup P_{candidates}$. Here, to identify the global candidate partitions from P_{all_cand}, from all the processing nodes, we propose *computeGlobalCandidatePartitions* algorithm given in sec. 4.2.b.

We use an R* -tree [10] which is a hierarchical, height-balanced data structure. The manager contains all the nodes of the global R*-tree which is created from all the information available in each processing node. Each internal node in the global tree contains entries of the form (*I, child-ptr*), where *I* is a MBR that encloses all the MBRs of the descendants, and *child-ptr* is the pointer to the specific child node. Leaf nodes contain entries of the form (*I, tuple-ptr*), where *I* is a MBR of a local candidate partition or its neighboring partitions.

b) Computation of Global Candidate Partitions

After the construction of the Global Tree (GTree) in the manager, we identify the global candidate partitions from partitions in P_{all_cand}. The procedure *computeGlobalCandidatePartitions* presented below in the Algorithm_Manager is responsible for this phase. In these steps, we first determine the bounds (lower and upper) on D^k for points in each partition indexed in the GTree by procedure *computeGlobalLowerUpper*. Then determine the global candidate partitions i.e. the candidate partitions that can potentially contain outliers.

```
Algorithm_Manager
(procedure computeGlobalCandidatePartitions)
Input: GTree, $P_{all\_cand}$
Output: Global Candidate Partitions (PGlobalCandidatePartitions)
Procedure computeGlobalCandidatePartitions($P_{all\_cand}$, k, n)
Begin
  candidateHeap:=∅
  minDkDist:=0
  for each partition P in $P_{all\_cand}$ do{
    computeGlobalLowerUpper (GTree.Root, P, k, minDkDist)
    if(P.lower > minDkDist){
      candidateHeap.insert(P)
      while candidateHeap.numPoints()-candidHeap.top().numPoints() ≥ n do
        candidateHeap.deleteTop()
        if(candidatHeap.numPoints() ≥ n )
          minDkDist := candidateHeap.top().lower
      }
  }
  PGlobalCandidatePartitions := ∅
  for each partition P in $P_{all\_cand}$ do
    if (P.upper ≥ minDkDist){
      PGlobalCandidatePartitions := PGlobalCandidatePartitions ∪{P}
      P.neighbors := {Q :Q∈ $P_{all\_cand}$ and MINDIST (P, Q) ≤ P.upper}
    }
  return PGlobalCandidatePartitions
end
```

For the purpose of identifying the bounds *P*.lower and *P*.upper for a partition *P*, we find out the *l* partitions closest to *P* with respect to MINDIST and MAXDIST such that the number of points in P_1, P_2, …, P_m is at least *k* in Procedure *computeGlobalLowerUpper* which is basically same as procedure *computeLowerUpper*(sec.4.1). We do it for the all partitions in P_{all_cand}. To determine the MINDIST and MAXDIST, partitions are stored in two heaps lowerHeap and upperHeap and are maintained in the decreasing order of distance values. Thus, partitions with the largest values of

MINDIST and MAXDIST appear at the top of the heaps and used to determine the *P*.lower and *P*.upper.

minDkDist which is a lower bound on D^k for an outlier can be computed using the *P*.lower values for the partitions such that if P_1, P_2, …, P_m be the partitions with the maximum values for *P*.lower and containing at least *n* points, then minDkDist = min{P_i.lower:$1 \leq i \leq m$} is a lower bound on D^k for an outlier. Note that a partition *P* cannot be a candidate if *P*.upper ≤ minDkDist.

After computing the lower and upper bounds for each partition by invoking the procedure *computeGlobalLowerUpper* in procedure *GlobalCandidatePartitions* described above, the partitions with the largest *P*.lower values with at least *n* points are stored in candidateHeap. The partitions are stored in increasing order of *P*.lower in candidateHeap, and minDkDist is thus equal to *P*.lower for the partition *P* at the top of the candidateHeap (steps 7-11). If, for a partition *P*, *P*.lower is greater than the current value of minDkDist, then it is inserted into candidateHeap and the value of minDkDist is appropriately adjusted. In steps 15-21, the set of candidate partitions *PGlobalCandidatePartitions* is computed. Partitions *Q* that can potentially contain the k^{th} nearest neighbor for a point in *P* are added to *P*.neighbors. Finally, we can obtain reduced number of candidate partitions from P_{all_cand}.

4.3 Load Balancing Phase (LBP)

After determining *PGlobalCandidatePartitions* and *P.neighbors* from the previous phase, we can have global candidate partitions and non global candidate partitions respectively which belong to the processing nodes (workers). However, to ensure the distribution of the *PGlobalCandidatePartitions* equally among the workers, we propose a new technique for load balancing in this phase. The main idea of the proposed technique described below is to minimize the turnaround time (completion time) of tasks. It is basically composed of execution cost and communication cost of a task or tasks with all of its connected neighbors determined from *P.neighbors* and *PGlobalCandidatepartitions*, required for detecting outliers. In our case, a task represents a global candidate partition and partitions of its non-candidate neighbors.

We assume that we have a distributed system with *z* heterogeneous processing nodes. A processor's 'load' comprises all the execution and communication costs associated with its assigned tasks. We start first defining 'load' for a processing node (worker) *x*. Then, we find the processing node which is heavily loaded. The time required by the heaviest loaded processor will determine the entire task's completion time. So to find the processing node with the heaviest load, we need to compute the load on each of the *z* processing nodes. The final allocation out of all possible allocation will allot the minimum load to the heaviest-loaded processing node. And finally we can achieve balanced load in each processing node (worker).

Thus, a load for a processing node *x* can be defined as

$$Load_{ix} = E_{ix} + C_{ix} \qquad (1)$$

where, E_{ix} = execution cost for a task T_i on worker x.
C_{ix} = sum of the transmission costs among partitions located in a processing node x and partitions of a task T_i to be assigned onto it.

Considering the 'load' as a cost function for task allocation; we propose an algorithm based on greedy heuristics approach.

```
Algorithm_Manager (procedure LoadBalancing)
Input: PGlobalCandidatePartitions
Output: Allocated candidate partitions (PAllocatedCandidatePartitions)
Procedure LoadBalancing (PGlobalCandidatePartitions)
Begin
1. Determine the no. of neighbors $N_i$ from P.neighbors and the no. of
   global partitions considered as neighbors $P_i$ for each candidate parti-
   tion $P_i$ from PGlobalCandidatePartitions and maintain a list S and
   sort in descending order. Consider each P with the neighbors as indi-
   vidual task i.e. $T_1$, $T_2$, … $T_m$.
2. Determine the load using equ. 1 for a task $T_1$, from S for each proc-
   essing node and maintain a list V with the load $V_i$, where, i=1, 2,…,
   z processing node.
3. Sort $V_i$ into monotonically decreasing order.
4. Assign the task $T_i$ with the smallest value of $V_i$ to the corresponding
   processing node.
5. A copy of required partitions is sent to the processing node selected
   by the assignment (step 4).
6. Maintain a list of unallocated tasks in U from S.
7. If the same processing node (where some other tasks are allocated
   already) is selected for allocation of any other task from U then add
   the previous allocated cost (load) $V_i$ with the present load by equ. 1
   and then go to step 3.
8. Repeat the steps 2 to 7 until U is empty.
end
```

The algorithm described above allocates *PGlobalCandidatePartitions* to each processing node in such a way so that a balanced load can be achieved for the total system. For allocation of the candidate partitions, it always selects the processing node with the minimum load (step 4 and 7) as we calculate first its present load and the load for the possible assignment. Thus, it ensures the minimum turnaround time for processing candidate partitions for the *PAllocatedCandiadtePartitions* in each processing node.

4.4 Outlier Detection Phase (ODP)

After load balancing phase, we have *PAllocatedCandidatePartitions* assigned to each worker from *PGlobalCandidatePartitions* by manager. In each worker, for each candidate partition P from *PAlloactedCandidatePartitions*, the points belonging to the neighboring partitions within distance P.upper from P are the only points that need to be examined when computing D^k for each point in P. The following procedure *computeOutliers* is used to compute n outliers by probing the index to compute D^k values only for points belonging to the candidate partitions in each worker. The manager and the worker nodes use procedure *computeOutliers* to do their part as given below.

```
   Algorithm_Manager
   (procedure computeOutliers)
   Input: PGlobalCandidatePartitions
Output:Top n points with maximum
       distance $D^k$ values considered
       as outliers from each worker
   Begin
1. MPI_Bcast(Distribute the last
   value minDkDist during the procedure
   computeGlobalCandidatePartitions(sec.4.2.b) to each worker.
   Let it be global_minDkDist)
2. MPI_Send (Ask each worker to
   execute computeOutliers)
3. MPI_Recv(Collect n outliers from
   outlier.Heap in each worker)
4. Merge them and sort them in decreasing order.
5. Finally take the maximum with
   top n outliers.
   end
```

```
   Algorithm_Worker
   (procedure computeOutliers)
Input: PAllocatedCandidatePartitions, global_minDkDist
Output:Top n points with maximum
       distance $D^k$ values considered
       as outliers.
   Procedure computeOutliers (k, n)
   Begin
1. MPI_comm_rank(worker gets its
   worker ID number)
2. MPI_Isend(worker makes initial
   request for work and check the
   communication request)
3. if (!worker_id){
4. outlier.Heap=∅
5. while PAllocatedCandidatePartitions ≠ ∅ do {
6. for each point p in PAllocatedCandidatePartitions do
7. InsertIntoIndex(Tree, p)
8. if (P.upper ≥ global_minDkDist)
9.    for each point p in the
             partition P do {
10.getKthNeighborDist
      (Tree.Root, p, k,
   global_minDkDist)
11.if(p.DkDist ≥
   global_minDkDist){
12.        outlierHeap.insert(p)
         if(outlierHeap.numpoints()
           > n)
13.        outlierHeap.deleteTop()
14.       if(outlieHeap.numpoints()
            = n)
15.     global_minDkDist:=
   max{global_minDkDist,
oulier.Heap.top().DkDist
16.           }
17.        }
18.    }
19.return outlierHeap
20.}
21.MPI_Send(send top n outliers to
       Algorithm_Manager(step 3))
end
```

Procedure *computeOutliers* described above computes D^k_n outliers from the candidate partitions in *PGlobalCandidatePartitions* that are generated by *computeGlobalCandidatePartitions* (section 4.2.b) in each worker. It uses the index-based algorithm and takes as an input parameter, the minDkDist (in this case global_minDkDist) value that is computed in procedure *computeGlobalCandidatePartitions* and used to identify the *PGlobalCandidatePartitions* and to identify the candidate partitions. Throughout its execution, the procedure keeps track of the top *n* outliers in outlierHeap and global_minDkDist, the lower bound on D^k for an outlier. Since the value of global_minDkDist is continuously refined during the procedure, partitions whose *P*.upper value drops below global_minDkDist cannot contain any outliers and so can be ignored (steps 6). In Steps 7–16, for every point in *p*, the procedure getKthNeighborDist [3] is used to compute $D^k(p)$ for point *p* and *p* is inserted into the outlier heap if it is a stronger outlier than the current outlier in outlierHeap.

5 Conclusion and Future Work

In this paper, we proposed a distributed algorithm for outlier detection based on distance based approach. The algorithm finds top *n* outliers in its rank based on the distance of a point to its k^{th} nearest neighbor. A global filtering technique has been introduced to reduce the number of local candidate partitions from the original candidate sets. A new load balancing technique has been proposed for balancing the candidate partitions among processing nodes. To the best of our knowledge, this is the first proposal of a distributed algorithm for outlier detection in a shared-nothing multiprocessor environment.

The work is in progress. We have been working towards the implementation of the algorithm. In the future, we present the experimental results of our algorithm.

References

1. V. Barnett and T. Lewis. Outliers in Statistical Data. John Wiley and Sons, New York, 1994.
2. E. Knorr and R. Ng. Algorithms for mining distance-based outliers in large datasets. In Proc. of the VLDB Conference, pages 392–403, New York, USA, September 1998.
3. Sridhar Ramaswamy, Rajeev Rastogi, and Kyuseok Shim, Efficient algorithms for mining outliers from large data sets, In Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, pages 427-438 , Texas, USA 2000.
4. M. Breunig, H. Kriegel, R. Ng, and J. Sander. LOF: Identifying density-based local outliers. In Proceedings of the SIGMOD Conference, pages 93-104, 2000.
5. Spiros Papadimitriou, Hiroyuki Kitagawa, Phillip B. Gibbons, and Christos Faloutsos, LOCI: Fast outlier detection using the local correlation integral, In Proceedings of 19th International Conference on Data Engineering, Pages 315-328, Bangalore, India, March 2003.
6. Raymond T. Ng and Jiawei Han. Efficient and effective clustering methods for spatial data mining. In Proc. of the VLDB Conference, Santiago, Chile, September 1994.
7. Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. A database interface for clustering in large spatial databases. In Int'l Conference on Knowledge Discovery in Databases and Data Mining (KDD-95), Montreal, Canada, August 1995.
8. Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: An efficient data clustering method for very large databases. In Proceedings of the ACM SIGMOD Conference on Management of Data, pages 103–114, Montreal,Canada, June 1996.
9. Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. Cure: An efficient clustering algorithm for large databases. In Proc. of the ACM SIGMOD Conference on Management of Data, June 1998.
10. N. Beckmann, H.-P. Kriegel, R. Schneider, and B. Seeger. The R* -tree: an efficient and robust access method for points and rectangles. In Proc. of ACM SIGMOD, pages 322–331, Atlantic City, NJ, May 1990.

An Improved Approach to Extract Document Summaries Based on Popularity

P. Arun Kumar, K. Praveen Kumar, T. Someswara Rao,
and P. Krishna Reddy

International Institute of Information Technology,
Gachibowli, Hyderabad, India
pkreddy@iiit.net

Abstract. With the rapid growth of the Internet, most of the textual data in the form of newspapers, magazines and journals tend to be available on-line. Summarizing these texts can aid the users access the information content at a faster pace. However, doing this task manually is expensive and time-consuming. Automatic text summarization is a solution for dealing with this problem. For a given text, a text summarization algorithm selects a few salient sentences based on certain features. In the literature, weight-based, foci-based, and machine learning approaches have been proposed. In this paper, we propose a popularity-based approach for text summarization. A popularity of the sentence is determined based on the number of other sentences similar to it. Using the notion of popularity, it is possible to extract potential sentences for summarization that could not be extracted by the existing approaches. The experimental results show that by applying both popularity and weight-based criteria it is possible to extract effective summaries.

1 Introduction

Automatic Text Summarization is an increasingly pressing practical problem due to the explosion of amount of on-line texts. With the rapid growth of the Internet, most of the textual data in the form of newspapers, magazines and journals tend to be available on-line. Summarizing these texts can aid the users access the information content at a faster pace. However, doing this task manually is expensive and time-consuming. Automatic text summarization is a solution for dealing with this problem and is a very active research area.

Automatic text summarization is an extremely active research field making connections with many other research areas such as information retrieval, natural language processing and machine learning. Increased pressure for technology advances in summarization is coming from users of the web, on-line information sources, and new mobile devices, as well as from the need for corporate knowledge management. Commercial companies are increasingly starting to offer text summarization capabilities, often bundled with information retrieval tools [1]. The goal of text summarization is to take a textual document, extract content

S. Bhalla (Ed.): DNIS 2005, LNCS 3433, 310–318, 2005.

from it and present the most important content to the user in a condensed form and in a manner sensitive to the user's or application's needs [2].

In the literature, weight-based, foci-based, and machine learning approaches have been proposed. In this paper we have proposed an improved summarization approach using the notion of popularity. The popularity of a sentence is determined based on the number of other sentences similar to it. Through popularity criteria, it is possible to extract potential sentences for summarization that could not be extracted by the existing approaches. Mainly, the potential sentences in the middle of the given document can be extracted by the popularity-based approach. The experimental results show that by applying both popularity and weight-based criteria it is possible to extract effective summaries.

The rest of the paper is organized as follows: In section 2, we review the related research. In section 3, we briefly discuss about the weight-based and clustering approaches. In section 4, we present the proposed approaches. In section 5, we present the experimental results. The last section consists of summary and conclusions.

1.1 Related Research

In [3, 4], the weight-based method is proposed to extract the sentences based on the weight of the sentences. The basic unit of extraction is the sentence and the practical reason for preferring sentence to paragraph or words is that it offers better control for getting the summaries. The weight of each sentence is computed based on certain features such as location, title, cue words, stigma words and keywords. The higher the weight of the sentence, the more important it is.

Kupiec et al [5] proposed a machine learning approach to extract important sentences from the given document. It is essentially a modified Naive Bayes classifier. For each sentence, the probability of the sentence being included in the summary is computed based on the features such as, Sentence Length Cut-off Feature, Fixed-Phrase Feature, Paragraph Feature, Thematic Word Feature, and Uppercase Word Feature. The sentences with high probability are considered salient.

In [6], extraction of sentences using foci analysis is proposed. Foci analysis deals with identifying the foci in the given document and sending it to the Questioner module that generates questions based on the foci. The Answerer module tries to answer the questions prepared by the questioner module by creating a parse tree.

None of the above methods considers the diversity aspect in the given text and hence fails in identifying the salient sentences. The diversity aspect deals with identifying the main themes in the text (most relevant sentences) at the same time keeping the summary non-redundant.

Related to web-search, the HITS (Hyper-link-Induced Topic Search)[9] is proposed to find authoritative resources by using only connectivity information among the web pages. The intuition behind the HITS algorithm is that a document that many documents point to is a good authority and the document that points to many other documents is a good hub. The HITS algorithm repeatedly updates authority and hub scores so that documents with high authority scores are expected to have relevant contents, whereas documents with high hub scores are expected to contain links to relevant contents. In [10], a method to compute a page rank of web page is proposed. The page rank of a given page is computed based on the page ranks of the preceding pages that have a link to it.

In HITS and Page-rank algorithms, the importance of the given page depends on the number of parent pages. Similarly, for text summarization, we introduced a notion of popularity score for a sentence i.e., the importance of a sentence is determined by the number of other sentences similar to it. Using popularity score we show that it is possible to extract effective summaries.

2 Weight-Based and Clustering Methods

In this section, we briefly explain weight-based and clustering approaches that have been proposed in the literature for text summarization.

2.1 Weight-Based Method

Edmundson [3, 4] presents a survey of the then existing methods to automatic summarization and a systematic approach to summarization that forms the core of the extraction methods. In this method, the basic unit of extraction is the sentence. The main reason for preferring sentences as level of granularity to paragraph is a sentence offers better control for getting the summaries. Another reason is extracts below the sentence level tend to be fragmentary in nature. In addition, by considering the linguistic motivation aspect, sentence has historically served as a prominent unit in syntactic and semantic analysis and sentences can be represented in a logical form and taken to denote propositions.

The weight-based method computes the weight of each sentence based on certain features like location, title, cue words, stigma words and keywords. A sentence is given weight based on its location in the document. This feature is dependent on the type of the document. For example, in technical documents, sentences in the conclusion section are ranked high, while in news articles; first few sentences are ranked higher. Sentences containing title words are considered to have a higher score. Title words are those that are present in the title of the document, headings and sub-headings. Statistically significant words are given higher scores. Cue words are those words containing cue words/phrases like conclusion, concisely etc. They add a positive score to the word. Stigma words are those words that add a negative score to the word. Words like hardly etc come under this category. Keywords are the words that tend to be more redundant and talk about the main content in the given text. Score of a sentence is then computed as the sum of the scores of its constituent words.

The weight of each sentence is computed as:

W(S) = a C(S) + b K(S)+ c L(S) + d T(S)

where, W(S) = Weight of the Sentence, C(S) = Cue Phrases Score, K(S) = Thematic Term, L(S) = Location, T(S) = Title and a,b,c,d being constants.

2.2 Clustering Method

Clustering has been used recently for text summarization in [6,7]. Normally, a document is composed of a set of ideas or themes with elaboration at different levels. Clustering is a method to identify and group all the related sentences. It separates the themes present in the given document into different groups. The assumption is that the clustering method allows us to separate the sentences of the given document into different clusters such that each group represents a sub-theme.

For example, consider that we need to select a set of representatives from a community. The community hierarchy can be organized into different levels. Each level represents a different level of aggregation with the lowest level of granularity being the family. So in order to pick the representatives from the community, we first cluster the people into different groups. For instance, the people can be grouped based on their nativity. Then from the sub-groups, we pick the persons who are popular globally as well as locally and who have the innate talent to contest as representatives.

Clustering Algorithm: The algorithm begins by representing the given text in a graph with a sentence as a node. Two nodes are linked by an edge if their similarity coefficient exceeds a certain threshold. Two words are said to be similar if they match or are synonymous to each other. The similarity coefficient is the ratio of twice the number of similar words divided by the total number of words in both the sentences.

The global similarity coefficient denotes the similarity measure between two nodes in the graph while the local similarity coefficient denotes the similarity measure between a sentence and the similar words in a cluster. The algorithm identifies two sentences with high similarity coefficient, clusters them, and then greedily checks for other sentences whether their similarity coefficient with the common words of the first two sentences is above the threshold. The greedy check is done to maximize the probability of grouping all the related sentences into a single cluster. All the sentences whose similarity coefficient is above the threshold are put in the above cluster. Therefore, this method helps us to identify all the highly similar sentences that represent a particular theme in the text.

Now, all the clustered sentences are denoted by a single node and represented by the common words of the sentences. The graph is rebuilt considering the new node and the non-clustered sentences and the same process is repeated. During the clustering process, the number of words in a cluster keeps on decreasing and is less than the number of words in the non-clustered sentences. Therefore, the probability of matched words would be less and hence the global similarity coefficient keeps on decreasing. Since the global similarity coefficient is a decreasing value, the process of clustering stops when it reaches the threshold. In this way, all the sentences that represent a particular theme fall into one cluster. This method helps in separating out the main themes in the text and hence helps in capturing the diverse aspects in the text.

We now describe the steps of the clustering process. Given a text document, the similarity graph is constructed as follows. The initial value of the global similarity coefficient is the highest similarity coefficient among all nodes in the graph.

```
1 Build graph out of the given text with the sentence as node. Insert an edge
  if the similarity coefficient between the two nodes is above the threshold
2 While (Global Similarity Coefficient > Threshold)
  2.1 Select the nodes $< S_i, S_j >$ that have high similarity coefficient and
      cluster them and store the common words of the two nodes
  2.2 For all nodes other than $< S_i, S_j >$
    2.2.1 Compute local similarity coefficient with the stored common words.
    2.2.2 If (Local Similarity Coefficient > Threshold), add the node to the
          cluster.
          End For
  2.3 Represent the clustered nodes as a new node and denote it by the similar
      words of the sentences
  2.4 Rebuild the graph using the new node and the non-clustered sentences
      and go to step 2.1.
      End While
```

3 Popularity-Based Approaches

In this section, we first present the text-summarization approach based on the notion of popularity. Next, we present a hybrid approach that is a combination of popularity and weight-based approach.

3.1 Popularity-Based Summarization Approach

Given the text document and similarity threshold, the popularity of the given sentence is determined by the number of sentences having similarity measure greater than or equal to the given threshold. The popularity metric helps us to select the highly popular and content rich sentences in the document. The sentence that is similar to most of the sentences contains important key words related to diverse aspects. The advantage of this approach is that it helps in selecting some of the sentences omitted by the previous approaches like weight-based method. This approach allows us to comparatively select more number of sentences from the middle portion of the text (excluding the beginning and ending portion of the text) than the weight-based method.

Text summarization using popularity is carried out in four phases: Preprocessing Step, Building Text Graph, Computing Popularity, and clustering and selection.

- **Preprocessing Step:** In the preprocessing phase, all the stop words are removed from the document. Stop words are the words that tend to be highly frequent in the document and have very little relevance.

- **Building Text Graph:** In this step, the text is represented as an undirected graph G (V, E) with sentence as a node. Two sentences are linked by an edge if the similarity coefficient of the two sentences is above the threshold.
- **Computing Popularity:** The popularity of each node (sentence) in the graph is computed based on the popularities of all the nodes that point to it. The nodes are then sorted in the decreasing order of their popularity.
- **Clustering and Selection:** The sentences in the given text are clustered into themes and from each thematic group, the most popular sentence is selected based on its popularity score.

3.2 Hybrid (Popularity and Weight) Summarization Approach

We propose an improved text summarization approach by combining popularity and weight measures. Note that the popularity of a sentence is determined by the number of similar sentences that a sentence has with respect to other sentences in the text. Certain features like position of the sentence, presence of cue words etc determine the weight of a sentence. The above methods when applied independently fail to select all the salient sentences. By combining the above two methods, it is possible to improve the performance.

For example, consider that the first ten sentences in the given text have the same popularity. Therefore, if the popularity measure alone were applied, it would fail to identify the salient sentence. By taking into account the weight factor, the issue can be resolved. Note that the weight and the popularity measures should be merged in the right proportion. So in a given document, first sentence would be preferred over tenth sentence, as it possesses more weight. On the other hand, consider a situation where first sentence and tenth sentence have the same weight. Therefore, if the weight measure alone were applied, it would fail to identify the salient sentence. By taking into account the popularity factor, the issue can be resolved.

Another aspect is that weight-based approach determines the score for each sentence based on the sentence properties such as position and so on. Whereas popularity-based approach determines the score based on number of other sentences similar to it and extracts additional sentences that could not be extracted by the weight-based approach. Therefore, hybrid approach combines advantages of both approaches and generates effective summaries.

The hybrid approach contains the following steps: Preprocessing Step, Building Text Graph, Computing Popularity, Computing weights, Combining popularity and weights, and Clustering and Selection.

In these steps, the first three steps and the last one are similar to popularity-based approach. The weight of the sentences is calculated based on the location, cue words, title and keywords. It actually gives us the relative strength of the sentence in the document. Thus, the weight when combined with the popularity in a certain proportion will help us to identify the salient sentences in the given document. The proportion ratio depends on the type of text collections. The method proposed above derives its strength by exploiting the features of clustering, popularity and weight of the sentences.

4 Experimental Results

Normally, two kinds of approaches are followed to evaluate text summarization approaches. One approach is to experiment with a set of documents with manual summarizations and the other approach is to evaluate the summaries based on their performance for information retrieval.

We adopt the former way for evaluating our summaries. We adopted this approach, as it is the commonly followed one for evaluating the results while the other approach deals with performance issues. A selected number of users were chosen and were asked to select salient sentences from the texts taken from the test data set. The test data set is taken from a variety of sources like news-wires etc. The users include students of different ages and software engineers. The summaries generated by them were compared to the summaries produced by the system.

Relevancy-score was computed as the ratio of number of matched sentences between the system summary and human summary to the total number of sentences retrieved (by both the humans and the system).

Let H be a set of sentences retrieved by users, S be a set of sentences retrieved by system and M be a set of sentences in common to both H and S. Then,

Relevancy Score = $\frac{2*n(M)}{(n(H)+n(S))}$, where n (M) = number of sentences in M, n (S) = number of sentences in S, and n (H) = number of sentences in H. The higher the relevancy score, the more effective the system is.

In our experiments, the similarity threshold was fixed to be 0.3 that was determined iteratively. It was the same for both the clustering and the popularity approach. For hybrid approach, the score of the sentence is determined by the combining 40 % of weight score and 60 % of popularity score. The proportion was determined experimentally by manually looking at the effective summaries generated by the system. For the weight-based method, the weights of the sentences were computed based on certain features like cue words, title words, thematic terms and location.

Table 1 shows a comparative view of the number of sentences selected using hybrid, weight, and popularity-based approaches that matched with the sentences selected by the users. In all the cases, the number of retrieved sentences by the user/system was 10. The average relevancy score was found to be 0.743 for the hybrid approach, 0.56 for the popularity-based approach and 0.47 for the weight-based approach. The results clearly depict the superiority of the hybrid approach as against other approaches like weight-based method and popularity-based method.

The hybrid approach thus has performed better on account of considering both popularity and weight scores. The popularity approach when applied alone used to omit sentences that are at the beginning and ending of the text. Most of the sentences it used to capture belong to the middle portion of the text. On the other hand, weight-based approach when applied alone used to omit sentences that are in the middle portion of the text. As a result, there was no uniformity in selecting the sentences when these approaches were applied alone and they

Table 1. Comparison of Hybrid approach with other approaches

Input Text	Weight-based	Popularity	Hybrid
Text1	5	4	7
Text2	6	5	7
Text3	7	5	8
Text4	4	2	7
Text5	6	6	7
Text6	5	6	8
Text7	6	5	8
Text8	5	7	9

were biased to-wards a particular portion of the text. Since the hybrid approach takes into account both these features, it shows uniformity in selecting the salient sentences and this approach can be applied to different kinds of text.

5 Summary and Conclusions

In this paper, we have proposed a text summarization approach by using the notion of sentence popularity. The popularity of a sentence is the number of sentences similar to it. The popularity-based method extracts relevant sentences based on the popularity score of given sentence. It is possible to extract sentences that could not be extracted by weight-based approach. The experiment results show that the proposed hybrid method for summarization based on the notion of popularity and weight is giving improved results as compared to the weight-only and popularity-only approaches. As a part of future work, we are planning to conduct extensive experiments on diverse data sets.

References

1. Inderjeet Mani. Recent developments in text summarization. In Proceedings of the 10th International Conference on Information and Knowledge Management, pages 529531, Atlanta, Georgia, USA, 2001.
2. Inderjeet Mani. Automatic Summarization. John Benjamins Publishing Company, Amsterdam/Philadelphia, 2001.
3. H. P. Edmundson. New Methods in Automatic Extracting. Journal of the Association for Computing Machinery, 16(2):264-285, April 1969.
4. Edmundson, H.P. and R.E. Wyllys, Automatic Abstracting and Indexing-Survey and Recommendations), Communications of the ACM, 1961. 4(5): p. 226-234.
5. Julian Kupiec, Jan O. Pedersen, and Francine Chen. A Trainable Document Summarizer. In Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 68-73, 1995.
6. Min-Yen Kan, Single document summarization using focus analysis. http://www1.cs.columbia.edu/ hjing/sumDemo/ FociSum/intro.html, March 2003.

7. T. Nomoto and Y. Matsumoto. A new approach to unsupervised text summarization. Proceedings of the 24th International Conference on Research in Information Retrieval (SIGIR 01), pp. 26-34, 2001.
8. G. Salton, A. Singhal, M. Mitra and C. Buckley. Automatic text structuring and summarization.341-355, Advances in Automatic Text Summarization, edited by I. Mani and M. Maybury, 1999.
9. J.Kleinberg, Authoritative sources in a hyperlinked environment, in proc. of ACM-SIAM Symposium on Discrete Algorithms, 1998.
10. S.Brin and L.Page, The anatomy of a large-scale hyper textual web search engine, in proc. of 7th WWW Conference, April 1998, pp. 107-117.

Author Index