研究成果介绍
Web Data Management
Encountering, Keeping and Refinding

- SG-WRAP
- SG-WRAM
- Deep Web Data Integration
- Job Tong
- PIM: A NEW FOCUS
SG-WRAP
A Schema Guided Wrapper Generator
X. Meng, H. Lu, H. Wang, M. Gu

System features:
1. Generating extraction rules with the guidance of user-defined schema;
2. The wrapper generated based on the rules could be more accurate and better reflect the users' requirements;
3. Using different schemas, the wrapper can be easily integrated into the different data integration process.

Application Examples:
- Web robots URL: www.robots-txt.org
- Quotes URL: finance.yahoo.com
- Amazon URL: www.amazon.com

Modeling Document
The fetched HTML page is parsed. Syntax errors, such as missing tags in the original HTML document, are fixed during the parsing process.

Defining Schema
One input is the original HTML page, another input of the system is a user-defined schema for extracted data, which is obtained by the Schema Acquirer and displayed on the screen (just as the following figure shown).

Demonstration Interface
As such, when the user highlights a string in the page shown, its path can be identified, which is displayed as an HTML tree path.

Getting Instances
Establishing the relationship between the HTML tree and schema tree:
- Similarly, when the user clicks an element in the DTD, it is displayed as a semantic tree path in the schema (E).
- The Mapping Acquirer of the system captures those user clicks that associate strings in the HTML page and their corresponding elements in the DTD.
- For example, with the sample page in the left figure, the following mapping can be captured:

  html.body .table[0] .tbody[0] .tr[*] .a[0] .text: Web_robots.robot.name "Alkaline"

Generating Rule
In other words, from user interactions, the system obtains a set of instances of rules that map strings in the HTML page to elements in the schema.

SG-Wrap: Architecture

Web robots.html{
  Robot: body .table[0] .tbody[0] .tr[*] {
    Name: a[0] .text;
    Purpose: table[0] .tbody[0] .tr[0] .td[0] .text;
    Availability: table[0] .tbody[0] .tr[1] .td[0] .text;
  }

  With this set of instances of mapping rules, the Rule Generator generates data extraction rule by an induction algorithm, which takes the list of mapping rules instances L as input and returns a candidate rule by incorporating the similar mapping rule instances into a new extraction rule.

Rule Inducer
Rule Refiner
Wrapper Generator

The data extraction rules are induced from a limited set of instances. In order to guarantee that the data extraction rule is applicable for the entire HTML document, SG-WRAP includes a refining process. The Rule Refiner generates an XML document by applying the induced rule on the input page.

This refining process continues until the user is satisfied with the data extracted. The wrapper is generated based on the final data extraction rule by the Wrapper Generator.

ICDE 2002: San Jose, California, USA
pages 331-332, San Jose, CA, 26 February - 1 March 2002
SG-WRAM
Schema-Guided Wrapper Maintenance for Web-Data Extraction
X. Meng, D. Hu, C. Li

**Schema Guided Wrapper Maintenance**

**Motivation**
- The WWW is extremely dynamic and continually evolving, which results in frequent changes in the structures of web documents.
- Consequently, wrappers may stop working when the structures of the corresponding documents are changed no matter how they are generated.
- It is often necessary to constantly update or even completely rewrite existing wrappers, in order to maintain the desired data extraction capabilities.

- The Web are very dynamic: contents, page structures.
- Original wrappers can stop working: rely on Web page structures.
- Re-generating wrappers is not easy; heavy work load to system developers.

**Step 1: Data-feature discovery**
- Compute features of the data items in the original page
  - Data pattern feature: A syntactic feature
    - Represented as a regular expression, e.g., S (A-Z|0-9){0,9};[A-Z]|0-9))
    - Can be extracted using existing technologies

**Step 2: Data-Item Recovery**
- Traverse the new HTML tree following the depth-first traversal order
  - Use the old features to identify potential data items using 3 matching conditions: Hyperlink & Annotation & Data pattern
    - A mapping list including all the recognized data items
    - Each mapping contains:
      - Value of the data item
      - Path to it in the HTML tree
      - Path of the corresponding DTD element

**Step 3: Block Configuration**
- Observation: Data items are located in semantic blocks.
- Conforms to the user-defined schema.
- Data items are grouped in semantic blocks.

**Step 4: Rule Re-Induction**
- Semantic blocks contain mappings from data items to DTD elements.
- Induce new extraction rule by calling the induction algorithm in wrapper generator.
- Refine the rule by trying to ensure the extraction rule cover all other semantic blocks.
- Generalization is necessary.

**Examples**

**WIDM 2003: New Orleans, Louisiana, USA**
Pages 1-8, New Orleans, Louisiana, USA, November 7-8, 2003
Deep Web Data Integration
An efficient solution to help users access Deep Web
W. Liu, X. Meng

Definition
- The contents stored in Web databases, which can be accessed through query interfaces.

Characteristics
- Scale
  - 307,000 Deep Web sites, 450,000 Web DBs, 1,258,000 query interfaces.
- Structure
  - 348,000 (structured) : 102,000 (text) = 3 : 1
- Content
  - Covering all subject domains in the real world, such as economy, sports, politics, etc.
- Access approach
  - Query interface: the query forms on the Web pages to access Web databases.

Deep Web Data Integration

Description
- Provide users a unified access to search multiple Web databases efficiently, and represent the results under a global schema.

Challenges
- How to find right Web databases?
- How to query multiple Web databases in a uniformed way?
- How to extract structured data from unstructured result pages?
- How to merge all results under a global schema for subsequent analysis?

Interface Integration Module
- WDB discovery: find Web databases (or say, their query interfaces) from deep Web;
- Interface schema extraction: analyze and extract the schema (attribute information) from query interfaces;
- WDB clustering: classify Web databases by domain;
- Interface integration: identify matching attributes among the attributes of the query interfaces in a same domain, and generate a global integrated interface.

Query Process Module
- WDB selection: select the most appropriate Web databases to answer a given query on the integrated interface;
- Query translation: translate the query on the integrated interface into the queries on the local query interfaces;
- Query submission: submit the translated queries to their corresponding Web databases.

Result Process Module
- Results extraction: extract structural data records and data items from result pages returned by Web databases;
- Results annotation: assign semantically meaningful labels for the extracted results;
- Data merging: merge the results from different Web databases under a global schema.

VLDB2006 Ph.D. Workshop
Seoul, Korea Sept 11, 2006
JobTong (工作通)
Professional Job Information Search Engine

Introduction
- Job search engine integrate more than 100 job web sites
- Designed based on web 2.0 principles
- Search, share, collaboration

Deep Web Crawling
- Distributed Crawling System Based On Nutch And Hadoop
- Deep Web Crawling
- Support More Than 100 Computers

Search
- One stop job search engine support more than 100 job resources
- Incremental update
- Very high search speed
- Support job ranking and duplication removal

Share, Collaborate, and more...
- User register and online resume
- Resume write once and export to different format and style
- Job recommendation based on user profile
- Blog, comment and more
- Job trend analysis based on search log and click log

Http://www.jobtong.cn
PIM: A NEW FOCUS
Personal Information Management

Introduction of PIM

PI (Personal Information)
When talking about Personal Information of PIM, we focus especially on the capacity of information to affect change in our lives and in the lives of others. So PI means the personal information which influence him or her in some sense.

PSI (Personal Space of Information)
A personal space of information includes all the information items that are, at least nominally, under that person's control. An information item is a packaging of information. Examples of information items include: paper documents, electronic documents, email messages, web pages, etc.

MEMEX
MEMEX is the first definition of PIM. By Vannevar Bush in 1945, who expressed a hope that technology might be used to extend ability to handle information.

What is PIM?
A branch which focuses on how to keep and store the collected personal information and on how to manage it more efficiently. And PIM can be taken as a collective of various activities which are an effort to establish, use and maintain a mapping between information and need.

Input Technology of PIM

Information Input is the first step of PIM. Which is a process that information is collected from Public Space of Information. Which includes the listed activities:
Encountering: A directed search may return an unexpected result which is potentially useful in another context; we still have many serendipitous encounters and re-encounters with information in our everyday lives.
Auto-collating: Utilize AI tech to collect information automatically.
Manual input: Meta-data and other data need to be inputted manually.
Filtering and auto-classification: For so large amount of information, it is hard to decide what information should be kept for future.
Unification & Integration: Research on how to organize the info coming from various "Information islands".

Output Technology of PIM

Information Output is the third step of PIM. Which is a process that information is taken from a Personal Space of Information and tell the person in a easy way or remind him to do something.

Research topics on PI output:

Finding/Refinding: I remember I have encountered a web-site which has soft jersey of the football team I like best. But where is it?
CHI of PIM: Let a person get info he needs with the most easy way.
Mining: It is so, I have never realize it!
Reminding: Pushing relevant information on the user could serve as an important reminding function.

Key Tech: Push
The goal of research on information output is to:

Short term goal: Make advance work unnecessary for re-finding.
Long term goal: Make it so people hardly need to re-find.
Push and AI tech will play a key role to reach the goal.

A framework showing PIM activities

- Keeping activities: Affect the input of information into a PSI.
- "M-level activities": Affect the storage of information within the PSI.
- Finding/re-finding activities: Affect the output of information from a PSI.

Research Focus of PIM

- Finding, re-finding, reminding and "re-collection".
- Encountering, keeping, organizing & maintaining information.
- From PSI to "GIM".
- Towards a unification & integration of PIM support.
- Measurement and evaluation.
- Search, filtering, auto-classification and Enhancements.
- Digital memories, ubiquitous computing, Beyond email, etc.

Store Technology (Personal Data space)

About PDS
PDS (Personal Data Space) is the reflection of PSI in digital world, which includes all digital info stored in computer with relation to a person.
PDS has similar characters to PSI: pay as you go, open system, revolution, etc.

Research topics on PDS
- Data model of Personal Dataspaces - Security and privacy
- Data indexing, Backup and restore, View
- Data independence
- Meta data: DD, system table, profile
- PDSMS (Personal Dataspaces Management System)

Outlook of PIM Research

challenges & issues of PIM
- Information is fragmented; so, too, is the study of PIM.
- How to protect the privacy and security of personal information?
- Who owns the information in the workplace?
- How do we know what is working and what isn’t?

Outlook
- PIM is a new research area of data Management.
- Great opportunity and great challenge:
  - Encourage multi-disciplinary approaches.
  - Support the development of methodologies, frameworks and benchmarks for the evaluation of PIM tools and techniques.
  - "Pay as you go" is an important rule in PIM research.
  - From PIM to GIM, they will benefit people much.

Innovation data management!
And Innovation personal data management!
XML Data Management

Beauty & Creativity

- OrientX
- OrientStore
- XML Sequencing
- OrientX/Ontology
- Integration Auditing of Outsourced Database
OrientX
A Native XML Database System

System Architecture

Features & History

- Features
  - XML Schema based repository
  - Full support of XQuery1.0 and XPath2.0
  - Flexible native storage strategies
  - Novel Path index and Value index
  - Navigation and Algebra based query processing

- History
  - OrientX 1.0 (2002-2003)
    - OrientStore, Schema Manager, Data Manager
    - XPath Processing Module, XML Labeling Module, Index Manager
  - OrientX 2.0 (2004-2005)
    - Navigation-based Query Engine
  - OrientX 2.5 (2005-now)
    - Algebra-based Query Engine

Native Storage

- OrientX implements four kinds of storage strategies
  - Different storage strategies can be utilized according to application requirement

Storage Strategy | Feature | Application Environment
---|---|---
DEB | Depth-first Element Based | Queries which are consistent of absolute paths
CEB | Clustered Element Based | Queries which contain "//" or "'", and only retrieve the data of target node
DSB | Depth-first Document Based | Queries which need to access subtrees of XML data
CSB | Clustered Subtree Based | Queries which need to access the data based on semantically blocks

Schema-guided Index

- Schema-guided Index can support twig query and predicates efficiently

Navigation Implementation for XQuery

- According to XQuery core syntax, we abstract 13 operators to implement XQuery processing.

Algebra-based Query Processing

- As Relation Algebra, which follows sel-at-a-time style, XML query can be processed efficiently using Tree Algebra

Jing Wang,Xiaofeng Meng,Shan Wang. SUPEX A Schema-Guided Path Index for XML Data(Poster). VLDB 2002
OrientStore
A Schema Based Native XML Storage System

Introduction

- Native XML storage has great impact on I/O during query processing

- There are two issues about native XML storage:
  - The granularities of record:
    - Element Based (EB)
    - Subtree Based (SB)
  - The storage configuration of record:
    - Depth-First Order
    - Breadth-First Order
    - Clustering

Motivation

- How to utilize XML Schema information in XML storage?
- How to cluster records in order to reduce I/O?
- How to combine the two?

DSB & CSB

<table>
<thead>
<tr>
<th>DSB (Depth-first Subtree Based)</th>
<th>CSB (Clustered Subtree Based)</th>
</tr>
</thead>
</table>
| Similarity: Granularity of record is subtree | Difference: 
  - Divide doc into subtrees according to the physical page size
  - Records are stored in depth-first order
  - Partition the schema graph into semantic blocks
  - All instances of the same semantic block are stored together |

Example: one XML doc is stored in DSB and CSB separately

Why Schema?

- For storage:
  - Reduce the storage consumption through replacing tag name with SchemaNode ID.
  - Help to cluster records with the same TYPE.

- For query processing:
  - Help to tell possible ancestor-descendant relationship, which in turn helps avoid navigating unnecessary nodes.
  - Help to eliminate ambiguous path expression, thus avoid visiting unnecessary nodes for paths containing "/" or wildcards "*"

Conclusions

- The finer the granularity is, the more the space consumes, and also the longer the importing, exporting takes.
- Clustering according to schema information dramatically reduces the IO costs than Non-Clustering strategy.
- CEB needs less IO than CSB, because physical page of CEB can hold more element nodes than page of CSB.
- The order of granularity is CEB < DEB < CSB, and the order of space consumptions is CEB > DEB > CSB > DSB

XML Sequencing

Sequencing XML is a novel way of presenting both XML data and XML queries by structure-encoded sequences:
- The sequence data representation preserves query equivalence.
- Structured queries can be answered without expensive join operations.
- Can develop infrastructure that unifies indices on both the content and the structure of XML documents.

Motivation

In most XML indexing solutions, tree pattern is not a first-class citizen, instead, the most commonly supported query interface is the following:

Simple Paths → P (Node Ids)

For queries with "*", "/*" or branch nodes, join operation is needed.

Our sequence-based XML indexing present a major departure from previous XML indexing approaches. It supports a more general query interface:

Tree Pattern → P (Doc Ids)

The tree structure itself is used as the basic query unit and no join is needed.

Query Equivalence

- False Alarm:

  D: <P, PL, PLs, PLl, PLll>
  C: <P, PL, (LS, Pll), Pll>

- Solution:
  - Introduces a class of sequences called constraint sequences, which preserve query equivalence.

Storage Infrastructure

Path Index + Tree → DocID

Performance-oriented Sequencing

Index size comparison

OrientX/Ontology
Large Scale Ontology Data Management

System Overview

- OrientX/Ontology supports management of Ontology encoded as OWL documents.
- Physical storage model is based on file system which utilizes semantic model of OWL data.
- Inference and query are implemented on such physical storage model.
- Currently supports characters of OWL Lite & SPARQL query

Semantic is not a fixed thing

- Semantic represents the understanding of people for some thing. It always changes with people.
- E.g. Concept "Foreigner". American treat Chinese as a foreigner and Chinese treat American as foreigner. So concept "Foreigner" has different semantic for American and Chinese.
- E.g. Different teacher will have different definitions for 'excellence student'.

Original data to describe the right picture
1. pic-uri type singer;
2. pic-uri sing xxx;

Then people can define "FinnSinger" (xxx type singer) and (xxx sing yyy)\Rightarrow (xxx type FinnSinger)

- People will not use same rules to express the different semantics for the same data.

RDF MODEL

\[ M \subseteq R \times U \times (R \cup L) \]

- \( R \) is the set of resources
- \( U \) is the set of URI references \( \text{rdf:Property} \in U \)
- \( L \) is the set of blank nodes
- \( P \) is the set of properties \( U, B, L \) are disjoint \( P \subset R, \text{rdf:type} \in P \)

Storage Design

- Class part:
  - \( C = \{ \text{URI} | \text{Ex} < \text{URI}, \text{rdf:type} \text{owl:Class} > \} \)
  - \( R_c = \{ \text{C}, \text{C} | \text{C}, \text{C} \in C, \text{Ex} < \text{C}, \text{Ex} < \text{C}, \text{rdf:sibClassOf} \text{C} > \} \)
  - \( R_c = \{ \text{URI}, \text{C} | \text{Ex} < \text{URI}, \text{rdf:type} \text{C} > \} \)

Inference Method

- Initial Method
  - Appropriate when Database is initialized
  - RSAB (Rule Static Association Based)
  - RDAB (Rule Dynamic Association Based)
  - RGSB (Rule Grouped Sorted-based)

- Incremental Method
  - Appropriate when new data and old data coexist
  - PGSB (Pattern Group Sharing Based)

User Interface

Yan Chen, Jianbo Ou, Yu Jiang and Xiaofeng Meng. HStar-a Semantic Repository for Large Scale OWL Documents. ASWC 2006
Xiaofeng Wang, Jianbo Ou, Xiaofeng Meng and Yan Chen. Abox Inference for Large Scale OWL-Lite Data. SKG 2006
Integrity Auditing of Outsourced Database

Introduction

- Providing Database As-a-Service
  - Instead of taking care of all the database management task in house, all the user’s data is sent to a service provider

- The BENEFIT
  - In this way you can eliminate in-house hardware, software and expertise needs to run DBMS

- Challenges:
  - Communication overhead
  - Security Concern
  - ...

- Our Focus
  - Validation of query results (make sure that the query results returned by the service provider are both correct & complete)

Querying Encrypted Data

- Traditional Encryption Method
  - Can only handle equivalent query containing (≤, =)

- Query-Friendly Encryption Scheme
  - Special Index Approach (Hacigumus et.al., SIGMOD 2002)
    - Drawback: Query result is super set of the real result
  - Order-Preserving Encryption (Agrawal et.al, SIGMOD 2004)
    - Benefit: Enable flexible query predicates
    - Benefit: Guarantee the indistinguishability of encrypted data distribution from original data distribution

Related Work

Merkle Hash Tree Based Approach
Using aggregated signature to generate a proof of correctness & completeness

Challenge Token Based Approach
An approach based on probability

Database Outsource Scenario

Auditing Correctness

- Correctness
  - All the results must originate in the owner’s data and has not been tampered with

- Correctness Threats:
  - One may modify some fields of our data
  - One may add some malicious tuples into our data
  - One may duplicate some tuples in our data
  - ...

Auditing Completeness

- Completeness
  - Result includes all records that satisfying the query

- Completeness Threats:
  - A malicious application server may only execute the user query on part of the data or may just return part of results
  - One may delete some tuples from the data
  - One may delete some tuples from the query result
  - ...

Mobile Data Management

- Clustering Moving Objects in Road Networks
- Indexing of Moving Objects in Road Networks
- PhoneDB
- Quality Aware Privacy Protection for LBS
- “小金灵”
Clustering Moving Objects in Road Networks


Motivation

- **Application Example**
  - Real-time monitoring traffic congestion condition

- **Existing method**
  - Clustering moving objects in Euclidean spaces
  - Clustering static objects in spatial networks

**Challenge:** Objects moving in spatial networks & Metric is network distance

**Goal:**
1. Minimize cost of clustering and its maintenance
2. Minimize network distance computations
3. Support multiple types of clusters in a single application

Framework: CMON

- **Main Idea**
  - Cluster block (CB) as underlying clustering unit
  - Easy to maintain
  - Serve as a building block of different types of clusters
  - Clustering process
    - Continuous maintenance of CBs
    - Periodical construction of global clusters
    - Incremental network extension for CMON construction

- **Continuous maintenance of CBs**
  - **Splitting event in mid of segment**
    - Problem: the neighborhood of objects changes over time
    - Solution: dynamically maintain the order of objects over time

  - **Splitting event at the end of segment**
    - Use group splitting approach: split the CB by removing old ones, if the event time

- **Periodically Construct CMON**
  - **Main problems**
    - How to use CBs to construct application-level cluster with different criteria
      - Distance-based, Density-based, K-partitioning clustering
    - How to reduce network distance computation among CBs
      - Incremental network extension

- **Construct Clusters with Different Criteria**
  - **Distance-based CMON**
    - Definition: For each cluster, the minimum distance with other objects in the cluster is not longer than a user-specified threshold \( d \) \(( d \gg 1)\)
    - Combination of CBs based on their network distance
  
  - **Density-based CMON**
    - Definition: For each cluster, the average density is higher than a given threshold \( E \)
    - Same as the Distance-based CMON construction, but a dynamic minimum-distance constraint, related to density of candidate CB

  - **K-Partitioning CMON**
    - Definition: Given a set of objects, group them into the K clusters such that the sum of distances between all adjacent objects is minimized
      - Initially select K CBs as the seeds for K clusters and assign the remaining CBs to their nearest clusters to make distance sum minimum

Conclusions and Future work

- An unifying framework for clustering moving objects in network to support different cluster criteria
- Splitting clustering costs into different granularity in conjunction with movement feature in the road network

- **Future work**
  - Predictive Clustering of Moving Objects
  - Movement prediction at intersections

DASFAA 2007: Bangkok, Thailand April 9-12 2007
Indexing of Moving Objects in Road Networks
X. Meng, J. Chen, Y. Guo, Z. Xiao

Motivation
Indexing moving objects in road network setting with efficient indexing update performance

How to exploit the network constraints in the index to support frequent location updates of moving objects?
- The spatial property of objects movement is captured by the network and indexed by an R-tree.
- The R-tree remains fixed since the road network seldom change.
- Index the objects on each road segment by a dynamic structure.

A graph of cellular automata (GCA) Model
GCA - integrate traffic movement features into model of moving objects and the road network
- Structure of GCA
  - Each edge in the GCA consists of a cellular automaton (CA)
- Instance of a GCA
  - A mapping from the cells of the GCA to moving objects.
- Transition of GCA

Index Structure
- Adaptive Unit (AU) - a dynamic data structure
  - Group moving objects with the same direction, similar velocity and location.
  - AU = (uid, objId, object, upperBound, lowerBound, edgeId, ...)
- R-tree for the road segments
  - Adaptive Units
    - Dimension reduction by the road network.
    - One-dimensional AU structure.
    - Predicted Trajectory bounds.

Trajectory Prediction
- Simulation based Prediction - SP method
  - Simulate object's future trajectory.
  - Two simulated trajectories based on different assumptions on the traffic conditions.
  - Linearization of the discrete points to form trajectory bounds.
  - Adaptation of trajectory bounds.
  - Assumptions the same trends (slope) of bounds and adjust only initial locations.

Index Update and Query Algorithms
- Update Algorithms (the update of AUs)
  - Creating an AU
  - Dropping an AU
  - Adding objects to an AU
  - Removing objects from an AU.
- Query Algorithm (Window Queries)
  - Spatial search in the R-Tree.
  - Transform the 2D search to 1D search.
  - (X1, Y1, X2, Y2) -> (S1, S2)
  - Find the intersected AUs.

Conclusions and Future Work
- AU index scheme achieves high update performance.
  - Few update frequency; an accurate prediction method.
  - Low update and query cost; one-dimensional AU structure.
- More works:
  - Predictive query algorithm (processing at the intersections).
  - KNN, CkNN processing in road networks.

PhoneDB

A Small Footprint DBMS for Mobile Phones

J. Chen, S. Yin, C. Lai and X. Meng

PhoneDB

- Scalable
- High availability
- High-performance
- Transaction-protected data management
- Ease development and maintenance of cellphone applications

Platform
- Lenovo Smartphone

Hardware
- ARM7 39M
- 1.5M RAM Memory
- 8M FLASH NAND
- Nucleus OS & TI FPC

Software
- Nucleus Plus/ROSE33 OS
- TI Flash File System
- TIMMI Framework

Implementation
- Develop with ANSI C
- PhoneDB LIB of 100K size
- Use 100K RAM at runtime

Architecture

- Access Methods
- Buffer Pool
- Storage Manager
- Transaction
- Logging Manager

Design Rules

- Compression for data structure and code
- Reduce the usage of RAM
- Minimize write operation
- Make full use of fast-read operation
- Minimize updates to reduce erase operation

Key Techniques

- Key-based data model
  (key, value) representation like BerkeleyDB
  Application-defined scheme

- Write Buffer
  Data are read directly from flash memory
  Records in buffer must be dirty
  Defer and reduce write and erase operation
  Improved LRU Replacement, combined with Update Frequency Based (UFB)

- Log-based recovery
  "Immediate write"
  Write log record to data file
  Delete log record and update log record
  Write data record to data file
  Timestamps of index and data
  Update index in RAM
  Keep the consistency of index and data

Key Techniques

- Log-based record storage

- Append-only data storage
- Add the log record to the data file
  - Data records
  - Log records (delete log and update log)
- Separate the index file from data file
  - B* tree with append write
  - Splitting Prediction for ordered data insertion
  - Space recycle
  - Garbage collection when few valid data
  - Application shared pointer
Quality Aware Privacy Protection for Location-based Services
Z. Xiao, X. Meng, J. Xu (HKBU)

Location-Based Services
- Mobile yellow page
- Buddy trackers
- Electronic tour guides
- Traffic navigation
- Electronic coupons
- Emergency support service

Privacy threat and Requirements
- Unique identifier and Location Information cause identification of special individual
- Location Anonymity
  - k-anonymity model: cloak the user’s location by an extended region large enough such that it contains at least k-1 other users (location anonymity set)
- Identifier Anonymity
  - k-anonymity model: the requests’ locations are cloaked such that any location is covered by at least k-1 other requests so that a request is not distinguishable from the other k-1 requests (identifier anonymity set)

System Model

Quality Aware Location K-Anonymity Model
- Location Privacy
  - the request must be anonymized by the predefined maximum cloaking latency $t + \Delta$
- Temporal QoS
  - the cloaking region must be inside a circle $L \subseteq \Omega(x, \delta)$
- Spatial QoS
  - the cloaking region must be a certain distance from the user's location

Cloaking Algorithm
- Directed Graph $G=(V,E)$
  - $V$ the set of requests
  - $E$ the neighborhood between the corresponding nodes (requests)
- Min Heap
  - to order the requests according to their cloaking deadlines
- Spatial Index
  - over the location points of all requests
  - window query to quickly find the neighbors of a request

Improvement with dummy
- Motivation
  - When cloaking fails, generate dummy requests can guarantee a 100% success rate.
  - Only need to maintain the in-degree $r_k$ and out-degree $r_k$ of each node $r$

- Requirements
  - Dummies should be both in-degree neighbors and out-degree neighbors of $r$, thus the privacy level will be higher.
  - Dummies must be indistinguishable from actual requests.
  - Dummies should satisfy the spatial QoS requirement of $r$

DASFAA 2007: Bangkok, Thailand, April 9-12, 2007
“小金灵”嵌入式移动数据库系统

系统概述

“小金灵”是面向掌上电脑、PDA、手机等移动设备，进行数据存储管理的嵌入式移动数据库产品

应用需求:
- 公共信息发布
- 票务信息、天气和交通信息
- 实时数据采集
- 保险或健康数据采集
- 移动商务
- 移动防盗平台、移动证据管理
- 多媒体应用
- MP3、动画、照片、TV等
- 个人信息管理和位置服务

设计要求:
- 支持多种嵌入式操作系统
- 微小型内核结构
- 完善的数据库管理功能
- 高效的事务管理功能
- 高效的查询处理
- 多种安全保护机制
- 支持多种连接协议，接口简单实用
- 自动管理功能

体系结构

三层体系结构
- “小金灵”嵌入式数据库应用：移动客户端应用
- “小金灵”同步服务器：移动客户端与远程主数据库间的数据交换
- 远程主数据库服务器

技术特性及性能指标

1. 平台适应性
   支持WinCE, PalmOS, Eudora, Linux等
2. 微小内核，有效存储空间小
   占用内存30KB，执行代码308KB
3. 支持基本的SQL功能
   ANSI SQL：创建、数据集、更新、查询
4. 支持多数据类型
   Int, float, decimal, char, date, time等
5. 支持内部滚动条

典型应用

- 面向数据采集：食品卫生监督检测系统
- 面向无线通讯：军用后勤指挥掌上电脑系统
- 面向信息家电类：手机、机顶盒、车载计算等

应用开发与系统管理工具

- 应用开发接口API和构件库
- DB CENTER管理工具
- 同步应用管理器（Sync Manager）