Information Integration for Improved City Construction Supervision

A Data Level Information Integration Approach

Information Center
Beijing Municipal Construction Committee

Dr. Xie Dongxiao
Director
Oct-2008
Agenda

- The Inevitability - Demands for Integration
- Ultimate Goals - Data Sharing & Exchange
- The Right Path - Application Level vs. Data Level
- Roadmap – Data Level Integration
- Step by Step – Data Level Integration
- Enhanced Manageability – Related Software Designs
The Inevitability – Demands for Integration

- **Current Issues**
  - Inefficient to share information with different structures and perspectives
  - Difficult to have different applications to work accordingly
  - Impossible to search or analysis information across the board
  - Loosely-defined data exchange has been the bottleneck to serve higher demands

- **Future Challenges**
  - Establishing a highly efficient data sharing mechanism to build a cooperative working environment across different departments
  - Defining an intelligent data exchange infrastructure to serve broader demands
  - Maintaining a centralized data management to support search and analysis for complex demands
  - Building data warehouse combined with BI tools to support object orientated data analysis in various dimensions to support decision making

- **Our solution**
  - A fully pledged data level information integration is the key to hold the final success for solving our current problems as well as tackling future challenges
Agenda

- The Inevitability - Demands for Integration
- Ultimate Goals - Data Sharing & Exchange
- The Right Path - Application Level vs. Data Level
- Roadmap - Data Level Integration
- Step by Step - Data Level Integration
- Enhanced Manageability - Related Software Designs
Ultimate Goals - Data Sharing

- Our views for data Sharing
  - Seamless Sharing: within the same agency, different systems have access to the same information at any given time, greater data usage and better collaboration.
  - Single Owner: a piece of data is maintained by its single assigned owner while used by others, assuring high data quality.
  - General Analysis Support: information can be further used to support general analysis.

- Mechanisms for data sharing
  - Butterfly Schema
    - A basic peer to peer data sharing, a data pipe is built between single reader and the owner, once they agreed.
  - Star Schema
    - An orchestrator controls all the data sharing and maintainness. It sends data to different systems while receiving data from its corresponding owner for maintainness.
Data Sharing - Butterfly Schema

- **Pros:**
  - Easy to use when little data sharing is required

- **Cons:**
  - No centralized access control nor orchestration
  - Building direct data pipe between two peers results a spaghetti environment
  - Cost of maintainness increases dramatically as sharing structure gets complex
  - Data sharing chaos, everyone get confused
Data Sharing – Star Schema

- **Pros:**
  - The center orchestrates all the data sharing and maintaining
  - Owner and its readers do not need to build peer to peer data pipes
  - Cost of maintainness has linear increases regardless of sharing complexity
  - Good for complex sharing demands

- **Cons:**
  - The center needs to maintain high availability
  - The orchestrator needs to understand the nature for every data sharing demand while trying to maintain data at high quality at all times
Ultimate Goals - Data Exchange

Our views for data exchange:

- Cross Agency Collaboration: Information can be exchanged among agencies on demand, based on pre-defined rules.

- Final vs. Intermediate: Information exchanged among agencies usually presents result of the work, unlike data sharing needs to reflect the current state of the information regardless of its completeness.

- A must-have for e-Government: With data exchange mechanism, we can start to work on inter-government collaboration, the e-Government.

- Promotes Standard Protocols: the key to build a successful exchange mechanism is to evolve a set of protocols that can be widely used by others.
Data Exchange – Features

- **Unique Identity:** Everyone involved in data exchange has its own ID to identify itself in the system.

- **Enforcing Security:** Exchanged information will be encrypted and everyone will have its own private key for decryption upon information arrival.

- **Service ID:** Everyone can adapt or create a data exchange service in the system. Each service will be assigned with a unique ID, which leads to instructions on how to interpret the received data.

- **Client:** Everyone involved in data exchange will use a client software to get connected with the exchange service center.

- **Server:** Each data exchange server will need a managing software to handle various of work: client registration/logon, moderate data exchange, data forwarding, etc.
Agenda

- The Inevitability - Demands for Integration
- Ultimate Goals - Data Sharing & Exchange
- The Right Path - Application Level vs. Data Level
- Roadmap - Data Level Integration
- Step by Step - Data Level Integration
- Enhanced Manageability - Related Software Designs
The Right Path – Application Level

- Integration at Application Level
  - A top-down approach:
    - Analyze all workflows across the agency
    - Correctly partition the resource into different departments
    - Derive a single application solution for the entire agency
    - Applying a unified software structure to present each workflow while maintaining their inter-relationships
    - Defining all the rights and constrains for the user
  - Pros and Cons:
    - A clean design: unified solution, instant data sharing
    - Unfriendly to changes: rapid changes of workflows within the agency will require constant changes made in the system, sometimes can be catastrophic
    - Feasible for agencies that are unlikely to introduce changes in their operations – banking, insurance, telecommunication, etc.
Integration at Application Level

Before

<table>
<thead>
<tr>
<th>Application 1</th>
<th>Application 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB1</td>
<td>DB2</td>
</tr>
<tr>
<td>Application 3</td>
<td>Application 4</td>
</tr>
<tr>
<td>DB3</td>
<td>DB4</td>
</tr>
</tbody>
</table>

After

Integrated Application System

Integrated Data Base

Application Level Integration
The Right Path – Data Level

Integration at Data Level

- A workflow independent approach:
  - Build a subject-oriented database for centralized data sharing and maintaining
  - Build control mechanism to give each application the right to access information based on their work demand
  - Publish data services to start sharing
  - Each application update itself to connect to different data services for information sharing
  - Each data is properly maintained by its owner application at the code level
  - Periodically check data quality and notify the owner application whenever improper maintainness occurs

Pros and Cons

- Subject-orientated data integration process has little to do with the current workflows, thus relatively stable
- Extra management work is introduced to maintain a high efficient data sharing
- Feasible for agencies that are likely to change its operations and re-allocate its resource based on the fast developing market – construction, real estate, etc.
Integration at Data Level

Before

- Application 1
  - DB1
- Application 2
  - DB2
- Application 3
  - DB3
- Application 4
  - DB4

After

- Application 1
  - DB1
- Application 2
  - DB2
- Application 3
  - DB3
- Application 4
  - DB4

Data Level Integration

Central Database

Access Control
Agenda

- The Inevitability - Demands for Integration
- Ultimate Goals - Data Sharing & Exchange
- The Right Path - Application Level vs. Data Level
- **Roadmap - Data Level Integration**
- Step by Step - Data Level Integration
- Enhanced Manageability - Related Software Designs
Roadmap – Data Level Integration Overview

Phase 1
- Information Resource Planning
  1. Outline Available Information Resources (I. R.)
  2. Make Data Standards and Access Rules
  3. Publish Data Sharing Technical Standards White Book

Phase 2
- Building Fundamental Database
  1. Build Fundamental DB
  2. Develop and Deploy Data Services and AC Mechanism
  3. Develop & Deploy Real Time Mismatch System
  4. Develop & Deploy Shared Query System

Phase 3
- Building Data Center & Exchange Mechanism
  1. Build & Deploy Data Exchange Mechanism
  2. Develop ETL and Build Center Database
  3. Develop & Deploy SGB for Center Database

Phase 4
- Building Data Warehouse with HI
  1. Build Data Warehouse
  2. Deploy HI Tool(s)
# Roadmap – Data Level Integration Overview

## Data Level Information Integration Work Sheet

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2. Make Data Standards and Access Rules</td>
<td>Data Standards &amp; Access Rules</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Phase 2: Building Fundamental Database  | 4. Build Fundamental DB                                               |                                          |                                         |                                        |
|                                        | 5. Develop and Deploy Data Services and AC Mechanism                  | Data Service and Access Control System   | Application to Fundamental DB Access Standards |
|                                        | 6. Develop & Deploy Real Time Maintainness System                    | Real Time Maintainness System            | Fundmental DB Maintaining Standards     |
|                                        | 7. Develop & Deploy Shared Query System                              | Shared Query System                      | Fundmental DB Data Query Inquiry Standards |

<table>
<thead>
<tr>
<th>Phase 3: Building Data Center and Exchange Mechanism</th>
<th>8. Build &amp; Deploy Data Exchange Mechanism</th>
<th>Data Exchange Standards</th>
<th>Data Exchange Management System</th>
<th>Data Exchange Standards</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>9. Develop ETL and Build Center Database</td>
<td>ETL System</td>
<td>Subject Oriented Data Update Standard</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10. Develop &amp; Deploy SQS for Center Database</td>
<td>Data Center Query Inquiry Standards</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Phase 4: Building Data Warehouse with BI</th>
<th>11. Build Data Warehouse</th>
<th>Data WH Models</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12. Deploy BI Tool(s)</td>
<td>Business Intelligence Tool(s)</td>
<td>BI Usage Inquiry Standards</td>
<td></td>
</tr>
</tbody>
</table>
Agenda

- The Inevitability - Demands for Integration
- Ultimate Goals - Data Sharing & Exchange
- The Right Path - Application Level vs. Data Level
- Roadmap - Data Level Integration
- Step by Step - Data Level Integration
- Enhanced Manageability - Related Software Designs
Phase 1: Information Resource Planning

1. Information Resource Planning
   - Analyzing Workflows
   - Collect all available Information
   - Produce I.R. Index

2. "Data Standards & Access Rules"

3. "Data Sharing Technical Standards White Book"
Step 1: Outline Available Information Resources

- Adapt theories and practices introduced by IRP (Information Resource Planning)
- Analyze all the current workflows to understand the input and output of each workflow as well as their inter-relationships
- Using available tools, consistently applying the same methods/standards to identify the semantic of each piece of data and thus find out the duplicated data
Step 2: Make Data Standard and Access Rules

- **Data Standard**
  - Understand the meaning of workflow-independent design, that is to identify commonly shared subjects among the data, by analyzing each workflow.
  - Partition all the data (without multiplicity) based on the subjects they present, and define relationships among different subjects.
  - Subjects later become tables; data becomes attributes of the table; relationships between each subject form the reference constrains of the tables - the data standard.

- **Access Rules**
  - Access rules contain the rights for different applications to maintain and retrieve data.
  - Since there are duplicates for certain data, it is critical to assign the ownership to one of the duplicates that ensures the highest authenticity.
  - Applications will commit to maintain its own data as soon as it registers for data sharing.
  - Other applications who are not the owner will be given access rights to read the data if needed.
Example of Data Standard
Practice of Making Access Rules

- Access rules are defined in a “5Levels-4Rights-3Tables” scheme
  - 5 Levels:
    - Department
    - Work
    - Corresponding application
    - Data Table
    - Attributes of the data within data table
  - 4 Rights:
    - Read
    - Add/Insert
    - Modify
    - Delete
  - 3 Tables:
    - Department and Workflow Relation
    - Access Rights Matrix for Application and Subject
    - Access Rights Matrix for Application and Subject Attributes
“5level-4rights-3sheets” Scheme Diagram

Department Level

Work Level

Application Level

Subject Table Level

Attribute Level

Department & Workflow Relations

Access Rights Matrix for Application and Subject

Access Rights Matrix for Application and Subject Attributes
## Access Rights Matrix for Application and Subject Attributes

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Get</td>
<td>Mod</td>
<td>Add</td>
<td>Del</td>
</tr>
<tr>
<td>Construction Project Monitoring</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Building Contract Bidding</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Building Quality Control</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Construction Security Monitoring</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Building Permit Control</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Company Certification</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Career Certification</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Certification Real Time Management</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Statutory Enforcement Management</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Project Final Inspection</td>
<td>✓</td>
<td>✓</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Step 3: Publish Technical Requirement for Data Sharing

- The standards explain the statutory requirement for any application to update itself in order to enable data sharing.

<table>
<thead>
<tr>
<th>Data Sharing Technical Standards White Book</th>
<th>Chapter Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. Purpose</td>
<td>a. Requirements for Data Sharing</td>
</tr>
<tr>
<td>b. Audience</td>
<td>b. Make Update Plan</td>
</tr>
<tr>
<td>c. Key words Definitions</td>
<td>c. System Update</td>
</tr>
<tr>
<td>d. General References</td>
<td>d. System Testing</td>
</tr>
<tr>
<td>2. IT Infrastructure</td>
<td>e. Deployment and Training</td>
</tr>
<tr>
<td>a. Hardware Requirement</td>
<td>f. System Beta Release</td>
</tr>
<tr>
<td>b. Required System Software</td>
<td>g. System RTM Release</td>
</tr>
<tr>
<td>c. Database Requirement</td>
<td>h. Operation Management</td>
</tr>
<tr>
<td>d. Related Applications</td>
<td>i. Collecting New Requirements</td>
</tr>
<tr>
<td>a. System Management Standards</td>
<td>a. Register to Data Sharing</td>
</tr>
<tr>
<td>b. Fundamental DB Standards</td>
<td>b. Data Retrieve and Upload</td>
</tr>
<tr>
<td>c. Web Access Standards</td>
<td>c. Procedure for changes</td>
</tr>
<tr>
<td>d. OA System Access Standards</td>
<td>6. Tech. Documents Required from the Vendor</td>
</tr>
<tr>
<td>e. Basic Tech. Standards</td>
<td></td>
</tr>
</tbody>
</table>
Phase 2: Building Fundamental Database

Phase 2: Building Fundamental Database

“Data Standards”

Fundamental DB
Shared Query System

Fundamental Database

“Access Rules”

Data Services
& AC System

Fundamental DB
Real Time Maintainness System
Step 4: Building Fundamental Database

- Analyze the data standard obtained from phase 1 for possible errors.
- Finalize the naming convention for subjects (tables) and data (attributes).
- Introduce necessary index, views, and constraints to enforce service performance as well as to maintain the correct relationships among different subjects.
- Write scripts to build the database.
- Check the hardware environment to meet the performance requirements.
- Run scripts to generate database, import history data (Possibly need data migration tool support) and required CDD (Common Data Dictionary).
Process Diagram for Building Fundamental Database
Step 5: Develop and Deploy Data Services and Access Control Mechanism

- **Data Service Development**
  - Can be automatically generated by managing tool since all the services are atomic and subject orientated, independent to workflow
  - Applications can call services in different orders to meet their work requirements

- **Data Service Deployment**
  - All services are deployed in the form of Webservices, thus compliant with all XML standards
  - The keys to deployment are load balancing and dynamic hardware resource allocation

- **Monitoring Data Service**
  - Monitoring the performance of each data service
  - Monitoring the data access pattern of each application and across the board to tune up data serving for better sharing experience
  - Monitoring is done by watchdog software developed to reflect real-time situation
Data Service Deployment Choices – Centralized Deployment

- **Definition:**
  - Maintaining single set of data services for each subject (table)
  - All data sharing requirements go through a single moderator
  - Moderator calls to the correct data services for different sharing requests

- **Pros:**
  - Unified service controlling and monitoring, easy access control

- **Cons:**
  - Lack of personalization
  - Single point failure threatens the whole system
  - Service update error will affect entire system
  - Error on access control will cause illegal data sharing
Schematic for Centralized Deployment
Data Service Deployment Choices – Distributed Deployment

- **Definition:**
  - Deploy set of data services separately for each application based on its specific sharing demands
  - Access control is built-in with the service set to enforce proper data sharing
  - The same data service to different application can have different versions (attributes right control)

- **Pros:**
  - Better personalization
  - Single point failure has no effect on the system
  - Data services deployment does not interfere with other services
  - Better access control and monitoring
  - Easy to fine tune data services to utilize available hardware resources

- **Cons:**
  - Managing different versions of data services to the same subject is a challenge
  - Multiple deployments for different applications are inefficient
  - Hard to monitor the overall service performance
Schematic for Distributed Deployment
Step 6: Develop Database Real Time Maintainness System

- Maintaining data stored in fundamental database
- Build backup and restore mechanism
- Build lock down mechanism for different subjects
- Build service logs to trace back possible errors
- Improve overall service availability
Step 7: Build Shared Query System

- Provide fast access to all the data stored in the database.
- Easy to configure queries across subjects for higher management needs.
- Present data in different formats (charts, graph, dashboard) for intuitive views of the current data storage.
- All end users can leverage the established queries.
- A query can be copied/shared to other SQS systems for cross-agency collaboration.
Phase 3: Build Data Center and Exchange Mechanism

Phase 3: Building Data Center and Exchange Mechanism

“Data Exchange Standards”

Data Exchange

Data Center Exchange Control System

DC Data Shared Query System

Data Center DC

10

8

9
Step 8: Build Data Exchange Mechanism

Structure of Data Exchange Server System

Sender
- Organize Data
- Data Receiver
- Data Sender
- Address Book
- Logging Management

Receiver
- Organize Data
- Data Receiver
- Data Sender
- Address Book
- Logging Management

Data Exchange Service Control System
- Exchange DB
- Logging Management
- Address Decoding
- Account Authentication

Package Receive Service
- Package Send

Package Download Service
Step 9: Build Data Center

- A collective data storage for its subsidiary fundamental databases
- Has its own data standard, subject oriented, which can cover all the subjects defined among its subsidiaries
- Exchange mechanism is responsible for receiving, extracting, and storing data into data center for every subsidiary fundamental database, along with its defined subjects – a temporary storage
- A ETL tool is used at data center to periodically analyze the subjects from different subsidiaries, convert the data into data center's own standard, and store the converted data under proper subject – a final storage
Data Center and Exchange Mechanism Diagram
Step10: Build Shared Query Tools for Data Center
Phase 4: Data Warehouse and BI Development
Step 11.12: Building Data Warehouse with BI

- Using Data Center as source
- Compliant to all Data Warehouse features and requirements – Object Orientated, Multiple dimensions with Data Mart support, etc.
- Select proper BI tools for data mining
- Establish data warehouse management support
- Contribute in decision making process
Agenda

- The Inevitability - Demands for Integration
- Ultimate Goals - Data Sharing & Exchange
- The Right Path - Application Level vs. Data Level
- Roadmap - Data Level Integration
- Step by Step - Data Level Integration
- Enhanced Manageability - Related Software Designs
Enhanced Manageability – Related Software Designs

- All developed software will serve the managing requests for better data sharing and exchange

In summary:
- IRDS: Information Resource Directory System
- AC: Access Control System
- RTM: Real Time Maintainness System
- SQS: Shared Query Support System
- DX: Data Exchange System
RTM Structure Diagram

Data Initialization Management
- DB Initialization
- Static Data Maintainness
- Data Convert Tool
- Data Collect Tool

Data Operation Management
- Data Search
- Data Insert
- Data Modification
- Data Delete

Data Monitoring Management
- Volume Monitor
- Traffic Monitor
- Data Health Monitor

Data Backup/Restore Management
- Total Data Backup
- Incremental Data Backup
- Data Restore
- Data Import/Export

Data Structure Mapping
- Dynamic SQL Generation

Database Operation Management
- Database Backup/Restore

Database Manage Engine

Fundamental DB
SQS Structure Diagram
Thank You

Thank you !