Data Masking
Secure Sensitive Data
Improve Application Quality

Becky Albin
Chief IT Architect
Becky.Albin@softwareag.com
Data Masking for Adabas

The information provided in this PPT is entirely subject to change

Data Masking for Adabas is scheduled for release late 2011 or early 2012

SAG is looking for customers willing to participate in a beta-test in October time-frame

Using ‘Partner’ technology - GridTools
Data Masking - What the Analysts say

**Forrester Group**
(Noel Yuhanna) states that, “All enterprises dealing with private data in test environments should mask or generate test data to comply with regulations such as PCI, HIPAA, SOX and European Union (EU)”
80% of all threats come from inside and 65% are undetected

**Accenture and Information Week**
Security breaches are increasingly coming from inside

**Gartner**
70% of all security incidents come from insiders

**Ernst & Young**
An insider attack against a large company causes an average of $2.7 million US in damages, where the average outside attack costs only $57,000
Why Mask Data?

- Improve application quality
  artificially generated test data is not sufficient enough

- Secure your sensitive data in
  - Development environments
  - Test centers
  - Offshore activities

- Provide a real business data training environment without publishing
  sensitive data

- Compliance with legal regulations
Why Mask Data?

- Ability to consistently create reduced and secured test data
- Rapid masking of production data from across the enterprise to, deliver “de-identified” data for testing
- An essential and safe training environment for end-users, when using live production data for testing or training
- A repeatable and automated solution to reduce the resources needed to create test data
- Typical test data and volumes used in system testing is not always sufficient
- Creating easily high quality training data with low investment
Close a Gap
Provide what’s required - Hide what’s not necessary
Basic Concepts of Adabas Data Masking Solution

1. Unload & extract production data into a common format
2. Analyse inventory & classify data
3. Define test data creation rules
4. Extract masked data subset
5. Load into test environment

- Relational Databases
- Hierarchical Databases
- Sequential Files
- Data Subset
- Extraction Rules
- Referential Model
- Privacy Data
- Masking Rules

Knowledge Base
Architecture and Data Flow
Available Mainframe (z/OS) Solution (Version 1)

1. Create meta data structure in the knowledge base from Adabas SQL catalog (CDD)
2. Create rules via the UI
3. Establish the run-time knowledge base (same type as the business database. Usage: referential integrity (sub-setting) and intermediate results. Meta data and run-time knowledge base is a DB2 database.
4. Create process parameter file for run-time engine. Parameter per masking task
5. Read business data to be masked according to the parameters
6. Write masked data to the target database
Architecture and Data Flow
Distributed Scenario

1. Create meta data structure in the knowledge base from Adabas SQL catalog (CDD)
2. Create rules via the UI
3. Establish the run-time knowledge base (same type as the business database. Usage: referential integrity (sub-setting) and intermediate results
4. Create process parameter file for run-time engine. Parameter per masking task
5. Read business data to be masked according to the parameters
6. Write masked data to the target database
Data Masking - Secure Data and improve Quality

- Data Sources
- Meta data
- Samples

- • Masking
- • Sub-Setting
- • Analyze
- • Configure
- • Assign
- • Synthetic Data Creation

Solution to
- • Accelerate development
- • Reduce test cycles
- • Date shifted data
- • Functional testing
- • Regression testing

Ensure legal compliance
Create environments for
- • Development
- • External testing
- • Training

Increase data quality
Architecture and Data Flow
Adabas-based Access - Meta Data

1. Manage meta data
2. Repository for meta data and rules
3. Read meta data from Predict if available
4. Messaging system Accessing remote Databases
5. Accessing Adabas databases
Potential Features of Data Masking for Adabas v1

- Generate test data is a 1:1 relationship between source and target database
- A knowledge base which contains Meta data of source and target environments
- Masking on a field level
- Build a subset
- Masking a unique number with another one but keep the semantic
- Create own masking logic as “user exits”
Screen shot - test data generation
Known Requirements

- Input data sources
  - Adabas
  - Adabas database back-ups (future ?)
  - DB2 and other RDBMS’
  - Sequential files

- Filter during extraction sub-setting

- Define Relationship between tables (Adabas files)

- Manipulate a part of a field (column) possible as “user exits” or wizard-based

- Customer specific masking tables (look-up tables) as input possible

- Any source to any target masking
Features Details

- The properties of the Data Subset Extraction process are:
  - Multiple extraction rule definitions
  - Repeatable extraction schemes
  - Extraction rules wizard to generate definitions for complex extraction patterns
  - Automated extraction of related records using the same key values.

- The supported input formats are:
  - Sequential unload (for all file types except ORACLE and SQL Server)
  - DB2 unload is supported in DSNTIAUL, REORG and UNLOAD format (IBM utilities), and in FIXED, EXTERNAL, VARIABLE and CSV (CA utilities)
  - Direct access (for DB2, ORACLE, SQL Server, VSAM, GDG, sequential files)
  - Image copies (for DB2 files. In this case, a temporary sequential file is created and used).
Features Details

- The process output is always a sequential file
  - Adabas after version 1, ORACLE / SQL Server where the target is a database
  - DB2 is supported in DSNTIAUL and UNLOAD format. A reload facility is provided.
- The Data Subset Extraction allows users to launch ‘Data Entry’ (inserting, modifying and deleting operations) functions such as:
  - Create new Method: Activates a wizard which guide the user in creating new methods
  - Work with Selected Method: Conducts advice and ‘Data Entry’ operations of a selected method. In the method, it is possible to change filters (selection criteria), add or remove files, specify input and output data set for files involved in the method
  - Delete Method: Deletes a selected method.
### List of Data Elements

<table>
<thead>
<tr>
<th>Machine ID</th>
<th>Company</th>
<th>Data Store Type</th>
<th>Data Store Name</th>
<th>Name</th>
<th>Data Element Offset</th>
<th>Assigned Class</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CONSTANT</td>
<td>0_NO</td>
<td>1</td>
<td></td>
<td>SMALLINT</td>
</tr>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CONSTANT</td>
<td>C_NO</td>
<td>3</td>
<td></td>
<td>SMALLINT</td>
</tr>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CONSTANT</td>
<td>E_NO</td>
<td>5</td>
<td></td>
<td>SMALLINT</td>
</tr>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CUSTOMER</td>
<td>C_NO</td>
<td>1</td>
<td></td>
<td>SMALLINT</td>
</tr>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CUSTOMER</td>
<td>COMPANY</td>
<td>3</td>
<td></td>
<td>CHAR</td>
</tr>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CUSTOMER</td>
<td>ADDRESS</td>
<td>28</td>
<td></td>
<td>CHAR</td>
</tr>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CUSTOMER</td>
<td>CITY</td>
<td>48</td>
<td></td>
<td>CHAR</td>
</tr>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CUSTOMER</td>
<td>STATE</td>
<td>63</td>
<td></td>
<td>CHAR</td>
</tr>
<tr>
<td>ALLIE</td>
<td>DXPROJ</td>
<td>ODBC - XDB</td>
<td>CUSTOMER</td>
<td>ZIP</td>
<td>65</td>
<td></td>
<td>CHAR</td>
</tr>
</tbody>
</table>
Data Masking Solution

- Possible Roadmap - to be discussed
  - Introduce the product as is
    - Including Adabas data type support
  - Develop an interface to Adabas
    - Accessing a database directly
    - Read from database backup media
  - Interface to be called from other Adabas related products
  - Meta data handling
    - Predict export
    - Direct access
Adabas Data Archiving

Becky.Albin@softwareag.com
Chief IT Architect
Requirements for Adabas Archiving solution are real

- Customer database volume is growing exponentially
  - Forrester Group estimates that data volume doubles every 2 years

- A lot of data is historical
  - It’s needed - but not in the production database

- Legal requirements demand certain data be retained longer

IT analysts identify ‘DBMS archiving’ as a clear, present, major challenge across all industry segments
Traditional approaches to handling “too much data”

- Add More Hardware Capacity
- Purge old Data
- Focus on rigorous database tuning
- In-House solution
Basic Archiving needs

1. Relieve pressure on the production database
2. Place archive data on cheaper hardware storage device
3. Easy search of archived data
4. Discard archived data based on expiration rules
Subtleties of archive automation

Extraction of data from the production DB is usually on a large-scale.

- Process needs to be restart-able and have pacing capabilities.

Archive may be on a different computer/architecture than the current DB.

- Extraction (from Adabas) and insertion (to the archive) must operate separately from each other.

Data evolves over time.

- Product must allow for changing data layout, etc. AND changing metadata AND changing software levels.
Architecture
Archiving Benefits

1. Smaller Adabas Databases & Files
   Consume less disk & tape space

2. Faster Utilities and Recovery

3. Reduced CPU usage (Applications & Adabas)

4. Compliance w/Data Retention Regulations
   Archived Data Recallable
   Archived Data Searchable
New Release - late 2011

Placeholders for:
- Multiple file plan
  - Multiple files
  - Across Adabas’ Record Types
- Natural Syntax?
- Natural Objects?

Automated archive by defined threshold

Ability to archive unused files in their entirety
### Data Source Settings

**Data Source Short Name:** ADA-5/25

#### EXTRACTOR SETTINGS:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>DAEP</td>
</tr>
<tr>
<td>Type</td>
<td>ADABAS</td>
</tr>
<tr>
<td>Location</td>
<td>Database: 5 File: 25</td>
</tr>
<tr>
<td>Mode</td>
<td>Adabas Commands</td>
</tr>
<tr>
<td>Criteria</td>
<td>AA, 1A, 5A, AA, 1A</td>
</tr>
<tr>
<td>Attachments</td>
<td>FDT</td>
</tr>
<tr>
<td>Scope</td>
<td>Full Archive</td>
</tr>
</tbody>
</table>

#### DATA TRANSFER (FROM EXTRACTOR TO INJECTOR) DETAILS:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schedule Type</td>
<td>Automatic</td>
</tr>
<tr>
<td>Transfer Type</td>
<td>Peer to Peer Synchronous</td>
</tr>
<tr>
<td>Restart</td>
<td>Automatic</td>
</tr>
<tr>
<td>Pricing</td>
<td>On, 50 Reads</td>
</tr>
</tbody>
</table>

#### INJECTOR SETTINGS:

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>FOUAHCM</td>
</tr>
<tr>
<td>Destination</td>
<td>ADABAS</td>
</tr>
<tr>
<td>Filtering</td>
<td>None</td>
</tr>
<tr>
<td>Vault Query Fields</td>
<td>N/A for dest adabas, used for dest vault</td>
</tr>
<tr>
<td>Retention Type</td>
<td>Automatic</td>
</tr>
<tr>
<td>Retention Period</td>
<td>7 Years</td>
</tr>
</tbody>
</table>
Summary

Archiving tools must provide
- Automation, flexibility and ease of use combined with data management over extremely long periods
- Real-time, enterprise-wide administration
- Ability to archive and recall across architecture boundaries
- Exploit cheaper systems for storing historical data
Adabas Data Archiving update

Commonwealth of Massachusetts

- **Before Archiving**
  - SVES-REQUEST 225 million records
  - SVES-RESPONSE 101 million records
  - Nightly batch job ran 4 hours 20 minutes
    - CPU time 13 minutes 10 seconds

- **After Archiving**
  - SVES-REQUEST 113 million records
  - SVES-RESPONSE 49 million records
  - Nightly batch job runs 2 hours
    - CPU time 5 minutes
    - *Savings of $2300 per year for one batch job*
Adabas Data Archiving update

Commonwealth of Massachusetts

- SVES-REQUEST file
  - Occupies 68k DATA cylinders and 25 ASSO cylinders
  - Total: 93k (over 28 volumes)

- DASD chargeback from ITD is .0761 per 1000 tracks/day
- There are 50,085 tracks per volume
- $3.81 per volume per day
- Times 28 volumes equals $106.68/day
- $38,938.20 a year
  - After Archiving savings of approximately $19k per year
Thank you!