Archiving Relational Databases with SIARD Suite

Amir Bernstein,
Swiss Federal Archives
Presentation, Demonstration & Hands-on

- Relational Databases: a brief introduction
- Archiving Relational Databases with SIARD
- Demonstration: SIARD Suite and command-line
- SIARD Suite hands-on: group exercise
Relational Databases: a Brief Introduction

- Databases, the basics
- Database history, the way to the relational model
- The relational model
Database: The Basics

- A repository for a collection of computerized data files
- A database system consists of:
  - data
  - hardware
  - software
  - users
The Hierarchical Model (1960s)

- 1:1 or 1:n relations
- Redundancies

Football DB

- European Football
  - Leagues
    - Bulgaria
      - Lokomotiv Sofia
        - Hristo Bonev &c.
    - England
      - M. United
        - Dimitar Berbatov &c.
  - National Team
    - Bulgaria
      - Hristo Bonev
      - Dimitar Berbatov &c.
The Network Model (1960s)

- No redundencies
- Complex relations (n:m)

Diagram:
- Football DB
- European Football
  - Leagues
    - Bulgaria
      - Lokomotiv Sofia
        - Hristo Bonev &c.
    - England
      - M. United
        - Dimitar Berbatov &c.
  - National Team
Object-oriented Databases (1980s-1990s)

- Complex objects
- Code and data stored together

Football DB

**Bulgaria** - National Team
Hristo Bonev, Lokomotiv Sofia
Dimitar Petrov, Manchester United

**England** - National Team
John Terry, Chelsea
Sir Robert (Bobby) Charlton, Manchester United

Sponsoring – Bulgarian National Team
Sportfive Bulgaria
FA Marketing
The Relational Model (1970s)

- Introduced by Edgar F. Codd around 1970

- Basic assumptions:
  - Data have a longer life than software, hardware or systems
  - Data must be independent of software, hardware or systems
  - A query language must be standardized
  - All queries must be treated equally
The Relational Model - Advantages

- The model disconnects the schema (logical organization) of a database from the physical storage methods.
- It allows the separation of content and media.

**External Level**
- User defined views

**Conceptual Level**
- Logical view, "community user view"

**Internal Level**
- Physical description (blocks & pages), storage view
The Relation Model

- A simple table structure
- All information stored in tables

<table>
<thead>
<tr>
<th>N#</th>
<th>NAME</th>
<th>NATIONAL TEAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1</td>
<td>Dimitar Berbatov</td>
<td>Bulgaria</td>
</tr>
<tr>
<td>N2</td>
<td>Hristo Bonev</td>
<td>Bulgaria</td>
</tr>
<tr>
<td>N3</td>
<td>Michael Ballack</td>
<td>Germany</td>
</tr>
<tr>
<td>N4</td>
<td>Hannu Tihinen</td>
<td>Finland</td>
</tr>
<tr>
<td>N5</td>
<td>Marco Amelia</td>
<td>Italy</td>
</tr>
<tr>
<td>N6</td>
<td>Philipp Degen</td>
<td>Switzerland</td>
</tr>
<tr>
<td>N7</td>
<td>Tranquillo Barnetta</td>
<td>Switzerland</td>
</tr>
<tr>
<td>N7</td>
<td>Christoph Spycher</td>
<td>Switzerland</td>
</tr>
</tbody>
</table>
The Base Tables (Entities)

- Relations instead of redundancies

<table>
<thead>
<tr>
<th>League</th>
<th>Team</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1</td>
<td>BVB</td>
</tr>
<tr>
<td>L2</td>
<td>Byer Leverkusen</td>
</tr>
<tr>
<td>L3</td>
<td>FCZ</td>
</tr>
<tr>
<td>L4</td>
<td>Chelsea</td>
</tr>
<tr>
<td>L5</td>
<td>Munchester United</td>
</tr>
<tr>
<td>L6</td>
<td>Livorno</td>
</tr>
<tr>
<td>L7</td>
<td>Lokomotiv Sofia</td>
</tr>
<tr>
<td>L8</td>
<td>Eintrach Frankfurt</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>National Team</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1</td>
<td>Bulgaria</td>
</tr>
<tr>
<td>N2</td>
<td>Germany</td>
</tr>
<tr>
<td>N3</td>
<td>Finland</td>
</tr>
<tr>
<td>N4</td>
<td>Italy</td>
</tr>
<tr>
<td>N5</td>
<td>Switzerland</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Player</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>Philipp Degen</td>
</tr>
<tr>
<td>P2</td>
<td>Primin Schwegler</td>
</tr>
<tr>
<td>P3</td>
<td>Hannu Tihinen</td>
</tr>
<tr>
<td>P4</td>
<td>Michael Ballack</td>
</tr>
<tr>
<td>P5</td>
<td>Dimitar Berbetov</td>
</tr>
<tr>
<td>P6</td>
<td>Marco Amelia</td>
</tr>
<tr>
<td>P7</td>
<td>Hristo Bonev</td>
</tr>
<tr>
<td>P8</td>
<td>Christoph Spycher</td>
</tr>
<tr>
<td>P9</td>
<td>Kresimir Stanic</td>
</tr>
</tbody>
</table>
The Relation Tables (Relations)

<table>
<thead>
<tr>
<th>Player</th>
<th>National Team</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>Philipp Degen</td>
</tr>
<tr>
<td>P2</td>
<td>Primin Schwegler</td>
</tr>
<tr>
<td>P3</td>
<td>Hannu Tihinen</td>
</tr>
<tr>
<td>P4</td>
<td>Michael Ballack</td>
</tr>
<tr>
<td>P5</td>
<td>Dimitar Berbetov</td>
</tr>
<tr>
<td>P6</td>
<td>Marco Amelia</td>
</tr>
<tr>
<td>P7</td>
<td>Hristo Bonev</td>
</tr>
<tr>
<td>P8</td>
<td>Christoph Spycher</td>
</tr>
<tr>
<td>P9</td>
<td>Kresimir Stanic</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>National Team</th>
<th>Player</th>
</tr>
</thead>
<tbody>
<tr>
<td>N1</td>
<td>Bulgaria</td>
</tr>
<tr>
<td>N2</td>
<td>Germany</td>
</tr>
<tr>
<td>N3</td>
<td>Finland</td>
</tr>
<tr>
<td>N4</td>
<td>Italy</td>
</tr>
<tr>
<td>N5</td>
<td>Switzerland</td>
</tr>
</tbody>
</table>
Easy Queries

- All queries are possible
- Efficient search method

```sql
SELECT NATIONAL.PLAYER,
    NATIONAL.TEAM AS "NATIONAL TEAM",
    LEAGUE.TEAM as "LEAGUE TEAM"
FROM NATIONAL, LEAGUE
WHERE LEAGUE.PLAYER = NATIONAL.PLAYER;
```

<table>
<thead>
<tr>
<th>PNL</th>
<th>Player</th>
<th>National Team</th>
<th>League</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNL1</td>
<td>Hristo Bonev</td>
<td>Bulgaria</td>
<td>Lokomotiv Sofia</td>
</tr>
<tr>
<td>PNL2</td>
<td>Dimitar Berbatov</td>
<td>Bulgaria</td>
<td>Manchester United</td>
</tr>
<tr>
<td>PNL3</td>
<td>Michael Ballack</td>
<td>Germany</td>
<td>Chelsea</td>
</tr>
<tr>
<td>&amp;c...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Archiving the Relational Model

- What do we have to archive?
  - At least all tables

- Attention!
  - Datatypes must be suitable for archiving
  - Database table must be archived in a format suitable for long-term preservation
  - Values in the filed must also be suitable for long-term preservation
  - No codes
  - No encryption
The Goal: Preserving the Essence

- Data (primary & meta) and relations preserved
- „Look and feel“ is lost
Choosing the right Format

- Why format matters…

Try to read these disks with a modern machine

...10010100100...

Know the alphabet and translate

...23,010273,9300,00005…

See that it’s a data base. Know the language of that data base. Perform some statements in this language

„Shadrach gave 1 bushel of barley to the temple…”

„At the cbot February 1989, the trade limit for barley $0.09 per bushel …"
The SIARD Format

- **Software Independent Archiving of Relational Databases**

- SIARD is a universal file format, facilitating
  - SIARD converts database content into a single SIARD file
  - A SIARD file is a ZIP file (ZIP64) containing XML files
  - The SIARD file format is based on open standards: SQL:1999, XML, XML Schema, UNICODE, ...
The SIARD Archive

- Primary data
  - “content” folder with:
    - Folder for each table
    - All tables in xml format
    - LOB folders

- Metadata
  - “metadata” folder with:
    - One XML file (metadata.xml)
    - Includes all metadata from all levels
The SIARD Archive in a Glance:
SIARD Archive – an Open Format

- Official Planets format for archiving databases
- Can be used free of charge
- Downloadable for the SFA website
The SIARD Suite

SIARD Suite 1.0

Upload
Download

Examine and edit metadata

Databases

SIARD file
Prerequisites

- SIARD is platform independent
  - It operates in a JAVA environment (Java SE 1.5 or higher)

- SIARD can run on a single computer with a common GUI

Installation

- Click & install
- or direct use from a USB stick
The SIARD Suite Components

- **SiardEdit**
  - Edit your metadata
  - Create a SIARD-Archive with a new set of metadata
  - Match your metadata against those of a different archive
  - Update and complete your existing set of metadata
  - View and sort your primary data

- **SiardFromDb**
  - Convert your database into a SIARD-Archive
  - Create a full SIARD-Archive (with both metadata and primary data in the SIARD format), or:
    - Generate an empty SIARD-Archive (i.e. containing no primary data)

- **SiardToDb**
  - Facilitate your research within a given database
  - Load your SIARD-Archive into a database instance (with tables, views etc.)
  - Comfortably navigate and search within your database
SIARD Demonstration

- A stroll through a SIARD Archive (LADIS)
  - Using SIARD Edit
  - BLOBs in SIARD

- Archiving an Oracle DB with SIARD

- What’s inside? A look at a SIARD file

- ODBC connection and archiving a local MDB
SIARD – Hands-on!

- Four work groups
  - Archiving a database with SIARD (local / server-based)
  - Upload a SIARD archive into a database instance

- Rapporteurs
  - Your opinion on SIARD Suite
Exercise I – Create a SIARD Archive

- Launch SIARD Suite
- Download an Oracle database (cf. the following page)
- Navigate through the Data base using the SIARD Suite Editor
- Try to:
  - Add metadata
  - Edit the primary the data
  - Find the added meta data
  - Retrieve data to an Excel Sheet
- Please report to the plenary session
Exercise I – Create a SIARD Archive

- Database password: crm
Exercise II – Create a SIARD Archive

- Download an Access database
  - Use the database „crm“ provided on the USB stick (folder: databases)
  - Create a ODBC connection (remember the connection name)
  - Create a SIARD archive using the ODBC connection you have defined

- Navigate through the Data base using the SIARD Suite Editor

- Try to:
  - Add metadata
  - Edit the primary the data
  - Find the added meta data
  - Retrieve data to an Excel Sheet

- Please report to the plenary session
Exercise III – SIARD Archive to DB

- Download an Access database
  - Locate the “accounting.siard“ archive provided on the USB stick (folder: databases)
  - Create a new empty Access Database
  - Ensure you have read and write rights in this database
  - Create a ODBC connection for the database (remember the connection name)
  - Launch SIARD Suite.
  - Open the **accounting.siard**
  - Upload the SIARD archive into your empty access databases using the ODBC connection you have created

- Navigate through the Data base using MS Access
Exercise III – SIARD Archive to DB

- Try to:
  - Add metadata
  - Edit the primary data
  - Find the added metadata
  - Retrieve data to an Excel Sheet

- Please report to the plenary session
Any Questions?

- For further information please contact the Swiss Federal Archives:

  For SIARD: Amir.Bernstein@bar.admin.ch
Thank you! / Благодаря!