A European Public Health Project, DG-SANCO, 2005-2008

B.I.R.
Best Information through Regional Outcomes

The BIRO Adaptor

From Databases to XML files

Pietro Palladino
Dipartimento di Medicina Interna
Università degli Studi di Perugia

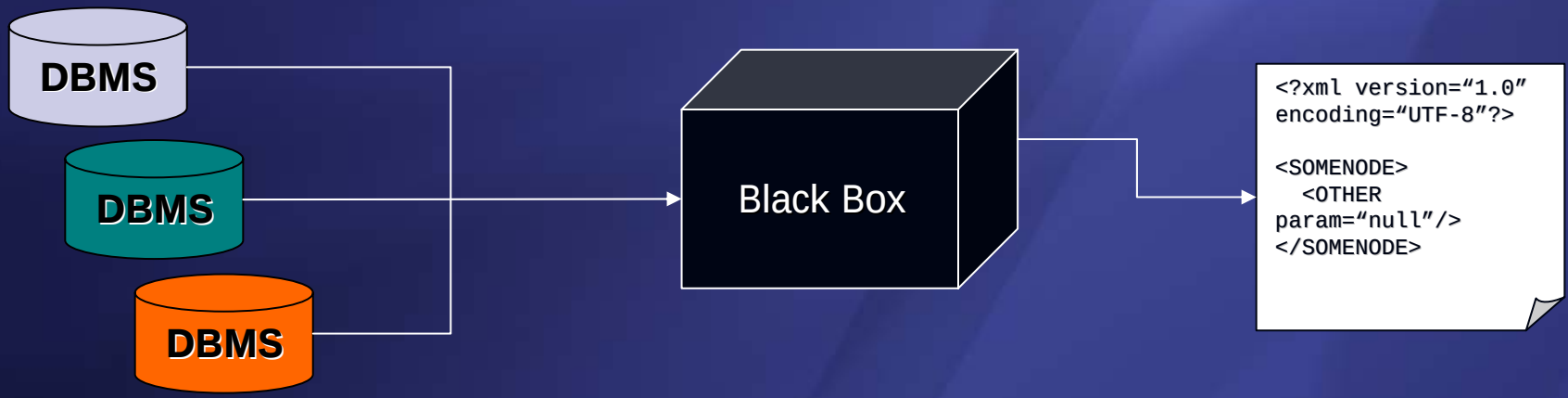
BIRO meeting @Rome
December 3-4, 2007

Agenda

- The ‘how to export’ problem
- The **BIRO** Adaptor solution
- Configuration file
  - Connecting to database
  - Export XML file
- Using the **BIRO** Adaptor (Perugia example)
- Future extensions

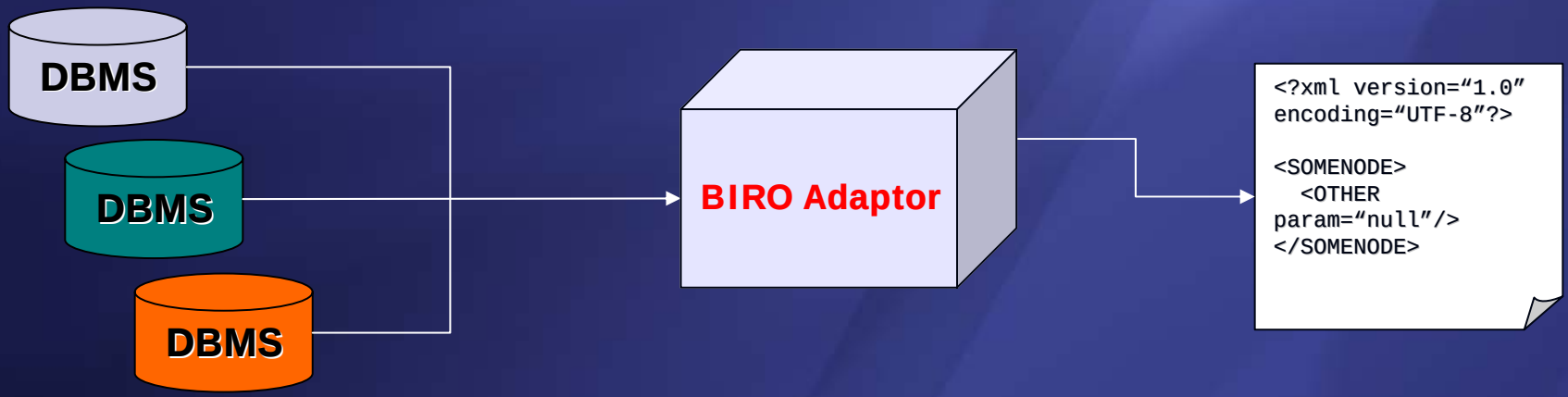
The problem

- Data are stored in different databases in different countries
- Need of some “black box” that makes it possible to obtain XML files following the same XSD schema (**BIRO** Schema)



The BIRO Adaptor

- The “black box” has to be able to connect to different DBMS in different operating systems in a common way
- The **BIRO** Adaptor, a small Java configurable utility, is the “black box”



... and counting ...

The BIRO Adaptor

- How does it work?
  1. It connects to a database
  2. It retrieves data from tables
  3. It organizes collected data into the **BIRO** Data Structure
  4. It writes down data in XML files
     - Note: The way the **BIRO** Adaptor creates XML files has changed a bit since the first release, we'll see it in details later on..
- Data are ready to be sent to the Database Engine
- Steps **1** and **2** depend on the DBMS where data are stored, we need to tell the **BIRO** Adaptor how to operate
- Steps **3** and **4** are **BIRO** standard and are the same among all countries, they depend on how the XSD is defined

How to configure the BIRO Adaptor

- The **BIRO** Adaptor is configured through its configuration file
  - A simple, well commented, standard Java properties file from which the **BIRO** Adaptor retrieves system dependant data
- The configuration is divided into different sections
  - **JDBC** → how to connect to database
  - **MergeTable** → how to retrieve data
  - **BIROFields** → how to match retrieved data with **BIRO** XSD
- Some configuration options and **BIRO** Adaptor behavior have changed since the first version ,we'll see them in these slides

JDBC Section - preliminaries

- **J**ava **D**ata**b**ase **C**onnectivity is used by Java programs to access DBMS in a standard way
- To use this feature one needs to put in the Java classpath the so called JDBC Driver of the used DBMS
  - There exists a JDBC driver for each commonly used DBMS such as PostgreSQL, MySQL, MSSQLServer, Oracle, and so on..

JDBC Section - configuration

- Lines starting with # are comments and are ignored by the parser

```
# JDBC Connection properties, starting section
[JDBC]

# JDBC Driver to use, specify the full name of the Java class
Driver=org.postgresql.Driver
# JDBC url to connect to, specify the full jdbc url
Url=jdbc:postgresql:rrdm
# JDBC User name, specify the database username
Username=postgres
# JDBC Password, specify [optional] the password for the database
# If this field is empty it will be asked on input console
# for more security
Password=
```

MergeTable section – preliminaries

- Now, the **BIRO** Adaptor can connect to the database where data are stored... but it does not know how to retrieve them...
  - Tables' structure and names are different among each DB
- The SQL language is used in DBMS to execute queries to retrieve data
- In every database there exists a query making it possible to retrieve all needed data from different tables
  - Using VIEWS, JOINS, sub-queries, and so on...
- Java, using JDBC, can execute arbitrary SQL queries

MergeTable section - configuration

```
# MergeTable information
[MergeTable]

# Specify the sql query to execute to retrieve all data
# to export in one table
SourceQuery=SELECT * FROM "MergeTable" ORDER BY "ID_CND", "EPIDATE" ;
# Specify the sql query to count rows to export
# (first column MUST be the one with the count)
CountQuery=SELECT COUNT(*) FROM (SELECT * FROM "MergeTable") AS N ;
# Specify date format of the date fields
DateFormat=dd/MM/yyyy
```

- The SourceQuery is the SQL query wich retrieves all data by patient row by row
- The DateFormat field is a new option, it is used to correctly parse different data formats
- The section is called MergeTable since what we want to obtain is a table view in the same format as the already known MargeTable is

MergeTable - example

idPaziente	dataDiNascita	sesso	DiagnosiDiab	tipoDiab	data	Altezza	BMI	Peso	Colesterolo	Colesterolo_HDL	Colesterolo
884	01/03/1918	M	01/01/1971	2	14/10/2002	170.00.00					
885	24/01/1938	M	01/01/1994	2	01/10/1999			89.00.00	229.00.00		
885	24/01/1938	M	01/01/1994	2	01/12/1999	162.00.00		76.00.00	246.00.00	40.00.00	
885	24/01/1938	M	01/01/1994	2	06/06/2001			84.00.00	874.00.00		
885	24/01/1938	M	01/01/1994	2	28/06/2001	162.00.00	32.00.00	84.00.00	154.00.00	32.00.00	
885	24/01/1938	M	01/01/1994	2	19/08/2003	162.00.00			210.00.00	33.00.00	155.00.(
885	24/01/1938	M	01/01/1994	2	10/10/2003	162.00.00			152.00.00	33.00.00	103.00.(
885	24/01/1938	M	01/01/1994	2	02/01/2004	162.00.00			117.00.00	27.00.00	66.00.(
885	24/01/1938	M	01/01/1994	2	11/11/2005	162.00.00			116.00.00	40.00.00	60.00.(
886	08/11/1912	M	01/01/1981	2	01/01/1997	172.00.00	22.10	65.50.00			
887	13/01/1924	F	01/01/1900	2	01/01/1997	153.00.00	35.70	83.50.00			
888	21/10/1946	M	01/01/1996	2	01/01/1997	166.00.00		75.00.00			
888	21/10/1946	M	01/01/1996	2	08/10/2000	166.00.00	27.20.00	75.00.00	210.00.00	47.00.00	
888	21/10/1946	M	01/01/1996	2	11/04/2002	166.00.00	0.00	0.00	212.00.00	40.00.00	118.00.(

- Two possible ways of working:
  - Pre-processing : create a MergeTable in your database (as we did in Perugia) → the query retrieves data from this table
    - Simple query definition, fast execution
  - Post-processing : use a complex query to retrieve a view of different tables → must use JOINS, VIEWS
    - Complex query definition, execution can be slow

BIRO XML - preliminaries

- Once the query has been executed, the **BIRO** Adaptor holds the data to be exported
- To create the XML file the adaptor has to know how to map database columns with XML parameters
- Different databases, with different tables, may have, of course, different column names...
  - There's the need to specify, for each **BIRO** field, the corresponding column name

BIRO XML – configuration (1)

- Some **BIRO** fields value are static (SourceProfile), they do not change from export to export → they are simply stored in the configuration file

```
# Static Fields
[SiteHeader]

DS_ID=2
DS_ADDRESS_1=Via E. Dal Pozzo
DS_ADDRESS_2=06126 Perugia, ITALY
DS_POST_CODE=06126
DS_COUNTRY=IT
DS_C_CONTACT=Massimo Massi Benedetti
DS_C_EMAIL=massi@unipg.it
DS_T_CONTACT=Pietro Palladino
DS_T_EMAIL=pietro.palladino@diei.unipg.it
```

BIRO XML – configuration (2)

- The **BIRO** fields mapping is done writing for each pair BIROField, ColumnName
  - <BIROFieldName>=<ColumnName>

```
# Profile Fields
[Profile]

PAT_ID=id_CND
TYPE_DM=TIPODIAB
SEX=SESSO
DOB=NASCITA
DT_DIAG=DATADIAG

# Episode Fields
[Episode]

EPI_DATE=EPIDATE
WEIGHT=WEIGHT
HEIGHT=HEIGHT
```

BIRO Adaptor at work

- Once configured, the **BIRO** Adaptor can
  - Connect to a database (JDBC)
  - Retrieve data (MergeTable)
  - Map data to BIRO Xml (BIROFields)
- Finally, it writes data to an XML file following the **BIRO** schema
- The Adaptor has been successfully tested in Perugia and in Bucharest
  - The export process took less than 2 minutes for about 172168 patient data rows

Note about the exported XML file

- The first released version of **BIRO** Adaptor created a big XML file for all patient data, optionally GZIPPED
- During our test we encountered some problems with Castor
  - Actually, the problem is not related to Xml parsing that works also for very big files, but for the Castor need to create many Java objects from imported data
- The current version creates many XML files (one for each patient) putting all together in a single ZIP file
  - This version will be released within some days on the **BIRO** forum

Conclusion

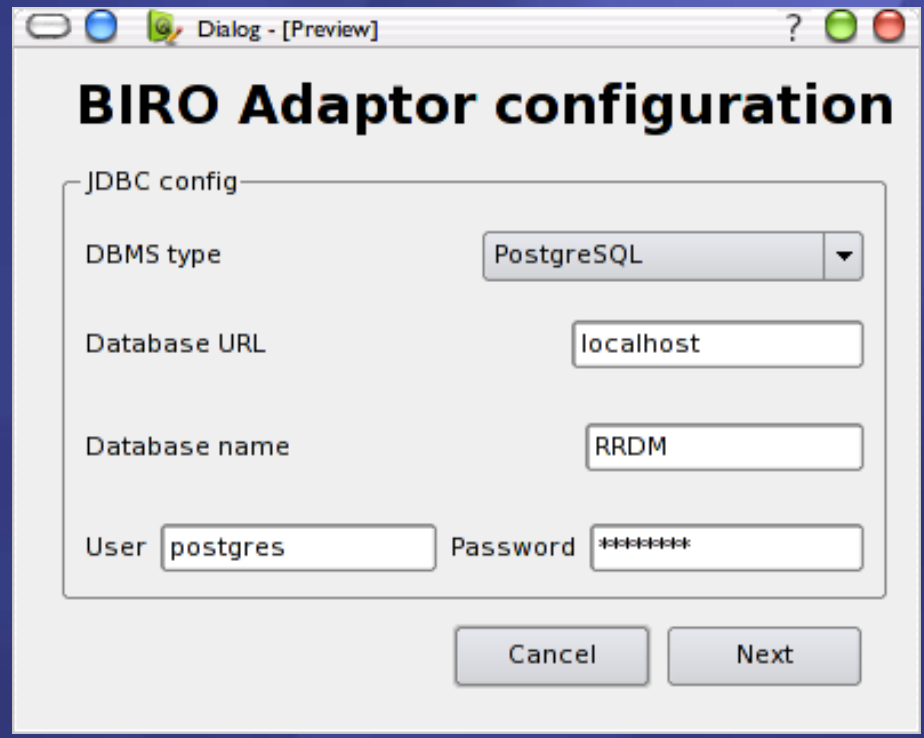
- The so created ZIP file (which contains many little XML files) has to be sent to the Database Engine which will import them into the **BIRO** Database
- The **BIRO** Adaptor should be tested by all partners and improved to meet all requirements

BIRO Adaptor extensions

- There are some extensions we though about to improve and make easy the configuration step
- It is under development a configuration GUI that helps, by the means of wizard components, configuring the Adaptor

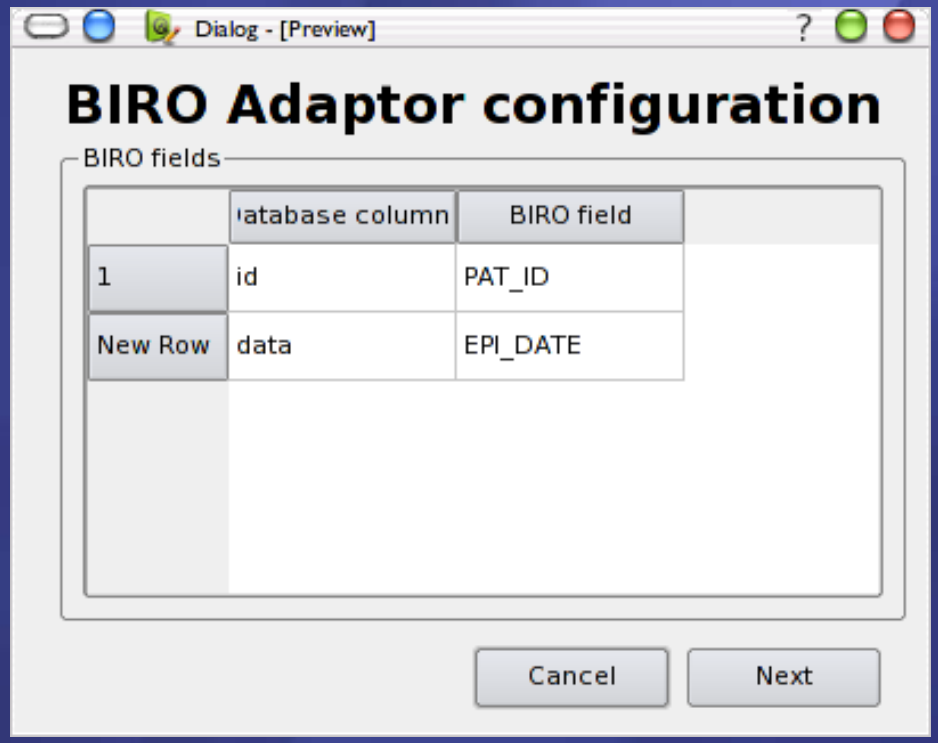
BIRO Adaptor GUI prototypes

- Simple QT-GUI prototype
  - Select JDBC driver from a list and automatically download the needed driver from the web



BIRO Adaptor GUI prototypes

- Simple QT-GUI prototypes
  - When selecting the database to connect to, a widget shows the available columns and helps mapping them to **BIRO** fields



Widgets under development...