Content

- Introduction
- Pitfalls in DW development
- Traceability as a solution
- Expected results & benefits
- Summary
- Ongoing research
Content

- Introduction
- Pitfalls in DW development
- Traceability as a solution
- Expected results & benefits
- Summary
- Ongoing research
Introduction

- Data Warehouse
  - Integrates several heterogeneous data sources in support of management’s decisions
Introduction

- Data Warehouse
  - Integrates several **heterogeneous** data sources in support of **management’s decisions**
Introduction

- Data Warehouse
  - Integrates several *heterogeneous* data sources in support of *management’s decisions*
Introduction

- Development
  - Current development approaches make use of up to 4 layers:
    1. Requirements (CIM)

![Diagram showing student success, improvement of teachers, and identification of successful students.]

Lucentig
Introduction

- Development
  - Current development approaches make use of up to 4 layers:
    2. Conceptual Models (PIM)
Introduction

- Development
  - Current development approaches make use of up to 4 layers:

  3. Logical Level (PSM)
Introduction

Development

- Current development approaches make use of up to 4 layers:
  4. Implementation (Code)
  
  CREATE TABLE STUDENT …

  CREATE TABLE SUBJECT …

  CREATE TABLE STUDENTSUCCESS …
Introduction

- Overview
Introduction

- Overview
Content

- Introduction
- Pitfalls in DW development
- Traceability as a solution
- Expected results & benefits
- Summary
- Ongoing research
Pitfalls in DW development

- The lack of traceability makes us unable to perform operations over multiple models:
  - How do we calculate the “quality” of the DW?
  - Which requirements cannot be fulfilled?
  - How do we introduce changes without losing all the previous work?
  - Why perform the matching between requirements and sources multiple times?
Pitfalls in DW development

- Quality of the DW
  - How *complete* is the **current design**?
Pitfalls in DW development

- Traceability of user requirements
  - Do we have the **necessary data**?

Requirements

Out of 183 tables

...And I am not sure if this is the right table...
Pitfalls in DW development

- Propagation of changes
  - The **impact** of **little modifications** can be **huge**
Pitfalls in DW development

- Propagation of changes
  - The **impact** of **little modifications** can be **huge**

Analyse the hours of study of each student

Students by hours of study

Lucentig
Pitfalls in DW development

- Reconciliation
  - Whenever we **introduce or modify** an element, we have to **match it** against the data sources
    - **No data, no use**

- Once we **finish** building the **DW**, we **still** have to **load the data**

Where did you say I had to put this…..?

<table>
<thead>
<tr>
<th>FI.PER</th>
<th>PK</th>
<th>file_code</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>hs_code</td>
<td>id_code</td>
</tr>
<tr>
<td></td>
<td>nam_name</td>
<td>nam_app</td>
</tr>
<tr>
<td></td>
<td>addres</td>
<td>city</td>
</tr>
<tr>
<td></td>
<td>birth_city</td>
<td>birth_date</td>
</tr>
</tbody>
</table>
Content

- Introduction
- Pitfalls in DW development
- Traceability as a solution
- Expected results & benefits
- Summary
- Ongoing research
Traceability as a solution

- Including traceability in the process:
  - First, define a trace metamodel with the necessary semantics
  - Second, automate trace generation
  - Third, define the necessary trace models and restructure the current process
Traceability as a solution

- Trace metamodel:

Semantic Links
Traceability as a solution

- Automatic trace generation:
  - **Extend** the current **MDD approach considering** the **trace metamodel**
Traceability as a solution

1. Requirements

2. Evolution

3. Datasource information

4. Reconciliation

- CIM
- Initial PIM
- Hybrid PIM
- Final PIM
- CIM2PIMTrace
- PIM2PIMTrace1
- PIM2PIMTrace2
- PIM2PIMTrace3
- DS2PIMTrace
- Datasource

Model to Model
Traces to Model
Content

- Introduction
- Pitfalls in DW development
- Traceability for DW development
- Expected results & benefits
- Summary
- Ongoing research
Expected results & benefits

- Once we have support for traceability, we expect the following benefits:
  - **Quality metrics**: Traceability allows us to calculate quality metrics for the DW
    - E.g. # of supported requirements, # of data sources, quality of the data, etc.
  - **Traceability of user requirements**: We are able to assess the status of each requirement at any point in development
Expected results & benefits

- **Propagation of changes**: We no longer need to be concerned about re-starting the process all over again due to changes.

- **Reconciliation process**: The reconciliation process is no longer behaves as a black box.
  - We have *explicit* record of the relationships between requirements and data sources.
  - ETL processes now have an *initial plan* instead of starting from scratch.
Content

- Introduction
- Pitfalls in DW development
- Traceability for DW development
- Expected results & benefits
- Summary
- Ongoing research
Summary

- **Tasks** in DW development **require information** from **multiple levels** of the architecture.

- Development approaches lack **traceability** and are **unable** to **match elements** in different levels.

- Introducing traceability in the process **reduces time and development costs** at the same time as it helps to **increase the quality** of the DW.
Content

- Introduction
- Pitfalls in DW development
- Traceability for DW development
- Expected results & benefits
- Summary
- Ongoing research
Ongoing research

- Inter-relating models allows to design new approaches:
  - A new methodology for DW development
  - Combine the **business strategy** with the multidimensional **DW schema**
  - Automatically **generate** analysis tools for decision makers
Ongoing research
Ongoing research
Requirements Engineering in Data Warehouses

QUESTIONS?
(and suggestions!)

Alejandro Maté
amate@dlsi.ua.es