

Model Migration Approach for Database Preservation

Arif Ur Rahman | Gabriel David | Cristina Ribeiro

{badwanpk,gtd,mcr}@fe.up.pt

Approach

Involves three major activities

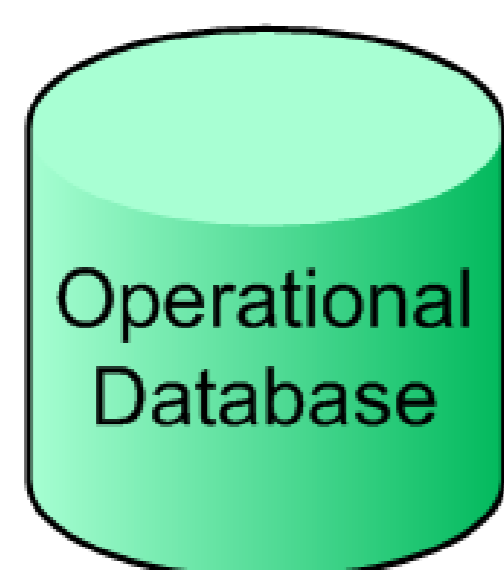
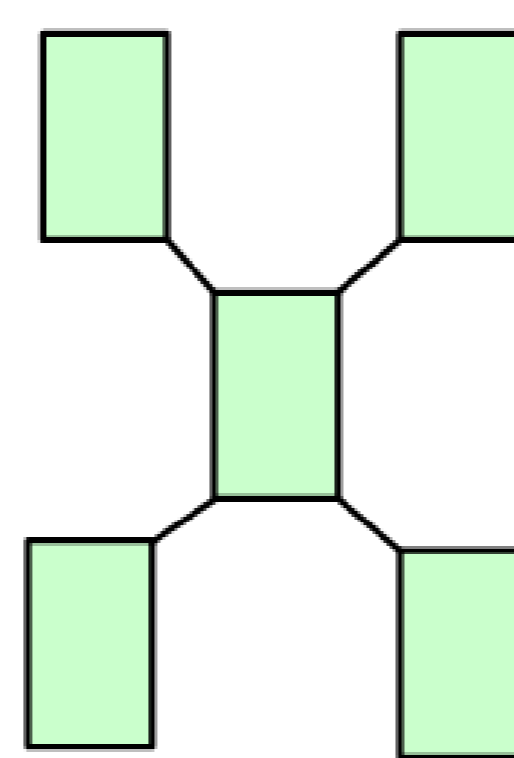
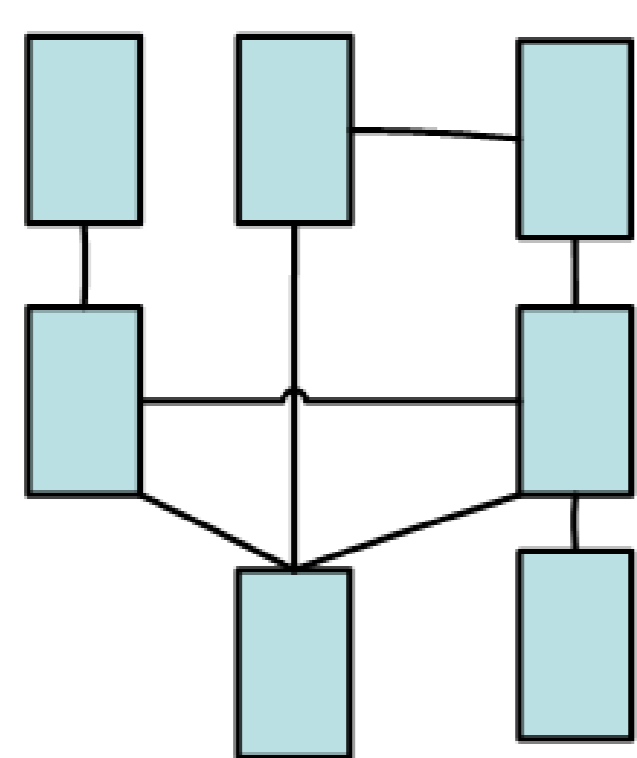
- Calculating the information embedded in code (functions, triggers, procedures)
- Storing it in dimensional model instead of relational model
- Export the information in the dimensional model to XML

Relational Model

- Relational Model of real size information systems - very complex and no quick access
- Important knowledge from the application domain embedded in code
- Better for transaction processing – not required for the preservation version

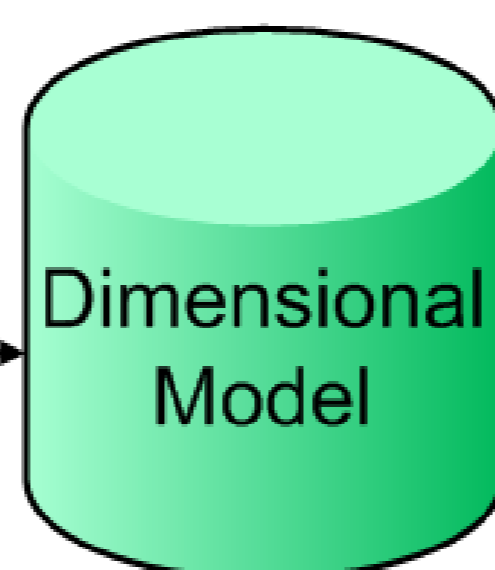
Dimensional Model

- Dimensional Modeling – Simpler and better for query processing and decision making
- Use of fact table with dimensions (star schema), for each significant process
- Dimensions have levels and hierarchies which makes aggregations of measures easier



Operational Database
Specific DBMS
Complex Model
Application Logic

Extraction,
Transformation
and Loading



Dimensional Model
General DBMS
Simple Model
No business logic

deserialize

serialize

```
<?xml version="1.0"
encoding="iso-8859-
1"?>
<DB name="XXX"
date="today">
<STRUCTURE>
...
</STRUCTURE>
<DATA>
...
</DATA>
</DB>
```

Preservation
format

Conclusions

- Alternative to preserving code in database preservation
 - Execute the code
 - Preserve the results in simple and easily accessible model
- Before migration, decide what is to be kept for the future and what can be discarded
 - Similar to the evaluation, elimination and description work of archiving
- Migration Process
 - Steps are documented
 - Results coming from the data in the dimensional model are compared with the operational system
 - The process is done with controlled loss of information