Evolutionary Student Research Projects in Domain Specific Modelling for an ERP-System with ADOxx

Presentation at ProSE Workshop Vienna, Oct. 31st. 2018

Frank Wolff & Kai Bieler

www.dhbw-mannheim.de
Two perspectives of this research:

- **Model-based ERP-configuration**
  - modelling methodological aspects
  - IT-system integration

- **Challenging innovation project in context of university course**
  - student capabilities
  - group and communication dynamics
Vision of the Project

Student course projects in business informatics at DHBW:

- **Standard projects**
  - application of familiar and established technologies
  - focus on integration of full stack of development issues
    - requirements with reconciliation of conflictive goals
    - implementation
    - testing

- **Innovative and research oriented projects exceed this, with**
  - incorporation of complex relatively new technologies
  - requisite inclusion of ‘new’ knowledge
  - high level of interdependencies
Basic Options for Research:

- Building a complete framework with integrated modelling- and ERP-functionality
  - most interesting from methodological perspective
  - very high effort required
  - more easily transferable to diverse systems

- Combination of existing systems
  - building on rich set of functionality
  - perspective in some aspects limited by chosen systems
  - must accommodate with some peculiarities of systems chosen
Selection of Suitable Systems for Exploratory Research

- Modelling system - ADOxx
  - definition of high quality graphics
  - adaptive incorporation of custom attributes
  - set of common interfaces

- ERP-system - iDempiere
  - adaptable workflows
  - multi-language and multi-currency features
  - web-services for data exchange
  - available with source-code and documentation freely
Structure of ERP-System iDempiere

Client-level

- Web-shop client
- Web-client
- opt. Swing-client

Application-level

- Jetty-http-Server
- ERP-Functions
- Extensions
- Reports
- Webservice & File-interface
- Accounting engine
- Workflow
- Model-based administration

Interfaces

Database-level

- Database

Server

- OSGI component framework
- Java virtual machine

Interfaces
Obstacles in the beginning:

- Anticipated problems
  - no complete definition of project goals
  - heterogeneous quality of documentation
  - high demand on abstraction in metamodelling

- Unexpected obstacles
  - systems provided on student laptops not fit for development environment
  - complexity of required XML-processing
  - group size effects on communication and decisions for redirection
Project management team
- General
  - Guiding project and adaptation
- Administration
  - Communication and control of current tasks

ADOxx team
- Domain-design team
  - Symbols for model elements
  - Attributes of classes
  - Comparison of modelling languages
- IT-development team
  - Implementation of designed classes
  - Supplement of expressions for renaming of classes
  - Analysis of XML-interface
  - Development of AdoScript routine for data-exchange

iDempiere team
- Domain-design team
  - Documentation of data structures
  - Tests with relevant configuration data
- IT-development team
  - Transform data for iDempiere Web-services
  - Automating the creation of web-services
Overview on model driven configuration of iDempiere with ADOxx
Some details on required XML-conversion etc.

Output of XML from ADOxx
- Inclusion of a dedicated button
- First solution three steps: initiating call, calling and then receiving XML-file
- Some additional elements in output fields

Conversion (part A)
- Translation of XML to JSON

Conversion of model-data (B)
- Filtering of object attributes - translation of attribute names
- Automatic configuration of required web-services (to be put aside in 2\textsuperscript{nd} project)
- Translation of relationships - query of technical key - inclusion in object-data
- Translation to XML

Input to iDempiere
- Ordering sequence of web-service executions
- Assembly of web-service-call and execution
- Processing of results (i.e. technical keys)
Model-based configuration of ERP-systems

First Results for Configuring iDempiere with ADOxx models

- Handling of subtle interdependencies with diverse design options
- Standard interfaces adequate for direct transfer
- Collection and documentation of knowledge on
  - ADOxx- XML-model-data
  - XML-transformation
  - iDempiere web-services
  - ADOxx – model-creation
  - iDempiere – import of model-data
- Student project can produce reusable results for iterative innovation projects
Model-based configuration of ERP-systems
Further Observations, Questions and Ideas

- ADOxx focus on single developers – missing out
  - structuring of different types of classes
  - support for concurrent development missing
  - assisted combination of library fragments areas

- Incorporation of standard web-service ESBs

- Influence of bias on choices of projects
  - technologies for development
  - focus of work options e.g.
    - ADOxx vs.
    - JavaScript routine config. tables

- ...

Further Observations, Questions and Ideas

Reflexive Research

Foundation for numerous successive innovation projects

- Cooperation with companies
  - Effects of size of company
  - Focus in different domains

- Projects interesting base due to provision and active usage of relevant technologies

- Improvement for succeeding projects

- Long list of interesting and demanding features
  - extension to other content areas
  - integrating other goals and bi-directional exchanges
Questions and comments?

www.dhbw-mannheim.de