

# From Organization Business Model to Information System: One approach and Lessons Learned

Janis ILJINS<sup>1</sup> and Maris TREIMANIS

University of Latvia

**Abstract** The system ISTechnology and lessons learned of its use in the business modeling and development of information systems are analyzed in the paper. The system consists of the meta-model and applications. The meta-model enables to define a platform independent business model of the organization. The applications provide the definition and interpretation of the business model. Interpretation of the business model provides functionality of the information system in the selected platform. The lessons learned confirm that the development and maintenance cost of information systems can be significantly reduced by use of the ISTechnology.

**Keywords:** meta-model, business modeling, software product line, information system, model driven architecture

## Introduction

Many approaches are used to carry out business modeling and information systems (IS) development. The ISTechnology is one of them. The first version of the ISTechnology was launched more than 14 years ago. Till nowadays the ISTechnology was successfully used in the development of various information systems [1, 2, 3] and is continuously improved. The goal of the present paper is to explain ideas of the ISTechnology and present the results of empirical research to show the practical benefits of the proposed approach.

The first chapter of the paper describes the most widely used ideas, approaches and tools for business modeling and IS development. Some of the abovementioned ideas can in integrated way be found in the ISTechnology. The second chapter gives a concise description of the meta-model and the applications of the ISTechnology. The lessons learned and benefits from the use of the ISTechnology in business modeling and IS development are presented in the third chapter.

---

<sup>1</sup> Corresponding author: Janis Iljins, University of Latvia, Raina bulv. 19, LV-1586, Riga, Latvia E-mail: Janis.Iljins@lu.lv

## **1 Related Work**

The aim of business modeling is to define such a model of an organization, which would be understandable for all stakeholders involved – business and IT experts, end-users, management, etc. [4]. The model is used to analyze, simulate and optimize business of the organization, as well as for more or less automated development of IS.

### ***1.1 Business Modeling Languages and Tools***

A lot of universal and domain specific business modeling languages and support tools for business modeling exist nowadays [5, 6, 7]. Many of them, being based on the business model, are able to perform some automation of a software development process. Rational Rose is one of the most popular tools which help to generate software from the model defined in UML language [8]. At the same time, for example, the business modeling language GRAPES-BM and its support tools allow both - to define the business model and to carry out the model simulation [9].

The Intalio|Cloud BPM can be mentioned as one of innovative tools which support the development of the business model and software [10]. This tool supports one of the newest standards for the business model description – Business Process Modeling Notation (BPMN).

There are also some business model based specialized tools, for example, Microsoft BizTalk Server. If web-services are used, there is a possibility to automate execution of these services. For such purposes Web Services Business Process Execution Language [11] and the appropriate tools were developed.

### ***1.2 Standard Software***

Various universal and widely used systems are also developed. For example, SAP, Microsoft Navision and many others, which are called standard software by the authors [12]. Enterprises use standard software and describe the business processes according to the models offered by these systems. These systems can also be customized. For example, Microsoft Navision has its own programming language for development of specific functionality of the organization. If the organization chooses standard software, often a lasting and expensive installation process begins, during which – either the functionality of the system or/ and the organization business model are changed [13].

### ***1.3 Software Reuse***

Also the role of software reuse is significant, for example, Software Product Lines (SPL) techniques can be mentioned [14]. SPL is successfully used in situations, where software development costs are high and safety, and performance are critical according to the authors [15]. One of SPL suggestions is building a family of software products in such a way that you can customize variability for specific requirement sets [16].

### ***1.4 Model Driven Architecture***

Model Driven Architecture (MDA) is not a business modeling method. It does not define how to build a business model, but it may be used to get the organization's IS appropriate to the business model. Transformations are used to get one model from other. Transformations may occur in several steps in MDA practice – from general Platform Independent Models (PIM) towards more specific and closer to implementation models (PSM's) until we get a code [17, 4]. According to [4], it is a business model, which can serve as the highest and most general description of the system, which may be further transformed into specific models. MetaEdit+ is the most popular tool, which supports MDA [18].

## **2 ISTechnology**

ISTechnology consists of the meta-model and applications for definition of a platform independent organization business model and implementation of this business model in the specific platform. The ISTechnology, being based on the organization platform independent business model, is compatible with the main ideas defined in [17, 13, 15].

### ***2.1 ISTechnology Meta-model***

According to [19], in order to define an organization business model, one should answer six questions – Why?, What?, How?, Who?, When? and Where?. ISTechnology meta-model allows to answer these questions and also supports recommendations of [20].

The latest version of ISTechnology meta-model consists of 59 object classes and 45 relations. The notion of “Module” is used to improve readability of the meta-model. Each module consists of object classes and relations. The ISTech-

nology meta-model consists of the Workplace, Organizational, Object, Process, Report and Monitoring modules. The classes and relations of the ISTechnology meta-model have predefined semantic. This semantic is used for automatic documentation purposes of the organization platform independent business model and to provide functionality of the IST Shell application (see Chapter 2.2).

Let us concisely consider the most relevant object classes and relations of the ISTechnology meta-model.

### 2.1.1 The Workplace Module

**The Workplace Module** is used to define the sets of employees' rights and responsibilities in the organization business model (Fig 1.).

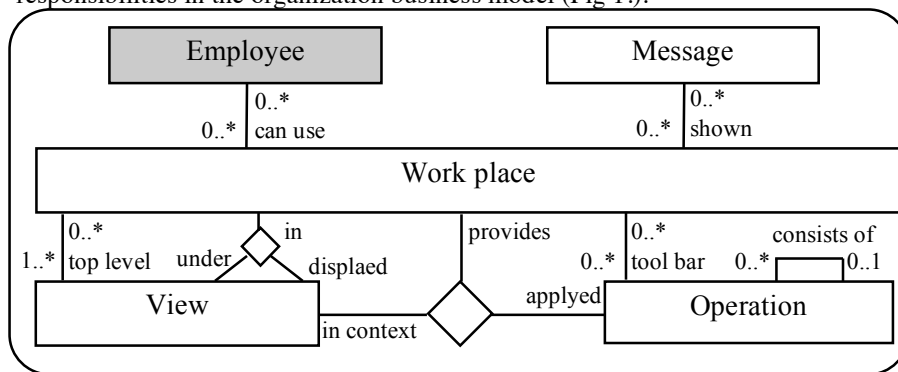


Fig. 1: Workplace Module

The class “*Workplace*” is a placeholder used to define the set of employees’ rights and responsibilities.

The class “*Employee*” (see 2.1.7) and the corresponding relation define employees, which can use the specified workplace. The shaded rectangle means that the object class is defined in the other meta-model module. For example, the object class “*Employee*” is defined in the “*Organizational module*”.

The class “*View*” and its relations define business objects available in a workplace, which in case of necessity may be structured in views and sub-views.

The class “*Operation*” and the corresponding relations define operations to be executed on business objects in the workplace. As a result of the execution of operations, new objects can be created, the existing ones can be deleted, and attribute values of business objects can be changed. In case of necessity the operations may be hierarchically structured.

The class “*Message*” and its relation define reminders, which can be viewable in the workplace and indicate that in the workplace according to the business model some operation must be carried out. For example, the reminder “*Deal* is

waiting approval!” is shown in the dealer’s workplace, if the transaction of buying a certain amount of some currency is carried out by another dealer.

### 2.1.2 The Process Module

The Process Module provides definitions of state–transition types of business objects and the corresponding implementation mechanism for object state–transitions (Fig. 2).

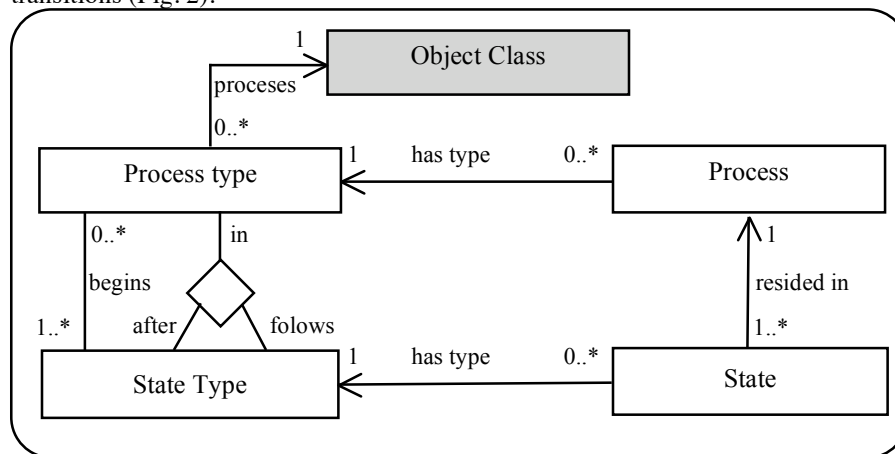


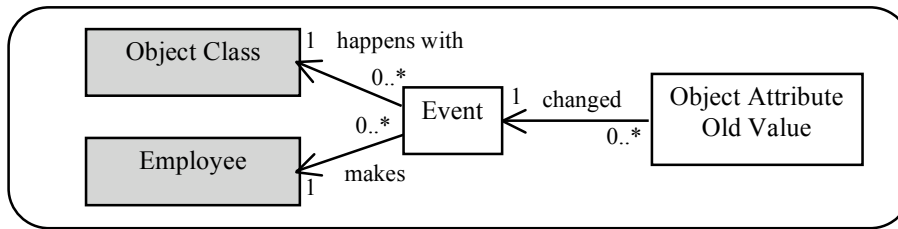
Fig. 2: The Process Module.

The class “*Process Type*” allows to define state transition types of business objects, while the class “*State Type*” defines states of a business object. Relations among the state types identify state transitions of business objects. The binary relation „*Process Type*” – “*State Type*” (cardinality – many-to-many) identify the initial states of the specified process type. The ternary relation „*Process Type*” – „*State Type*” – „*State Type*” identifies the permissible state transitions.

The classes “*Process*”, “*State*” and the corresponding relations allow executing state transitions of the business object according to the defined process types.

### 2.1.3 Monitoring Module

**The Monitoring Module** provides the organization business model monitoring, audit and improvement facilities (Fig. 3).



**Fig. 3:** Monitoring module

The class “*Event*” records all operations performed with each Object Class instance during the implementation of the business model. The main attributes of the event:

- time of the event
- business process type and instance
- object class and instance, by which the operation was performed
- operation – user’s entry / exit from the system or viewing, creating, correcting and deleting of the business object
- user, which has performed the operation

The class “*Object Attribute Old Value*” records the values of the previous business object attribute, which was deleted or modified. It should be noted, that not all operations (for example, viewing of attributes) create changes in the values of the attributes.

#### 2.1.4 Other Modules

**The Objects Module** enables to define the organization business object classes and their relations. This module plays the role of the Object model [7].

**The Reports Module** enables to define and prepare reports on the defined organization business model and business model dynamics, as well as business domain specific reports.

**The Organizational module** enables to define organizations, their locations (countries, regions), departments and their organizational structure, as well as employees and their roles. The module enables to define business model components, traditionally used for such purpose. The business model components defined by the organizational module are used to define the authority and responsibility of the employees and departments (Workplace module), as well as for monitoring and analysis of the business model dynamics (Monitoring module).

## ***2.2 ISTechnology Components***

ISTechnology consists of IST Shell, domain nonspecific (DNS) and Direpo applications. These applications are included in all IS being developed by ISTechnology. In development of IS, in addition to ISTechnology components, also domain specific (DS) applications have to be developed.

**IST Shell application.** IST Shell application provides the definition and interpretation of the business model according to the IST meta-model. Interpretation of the business model enables to perform operations with business objects. IST Shell provides the „central” user interface of the ISTechnology – window with subwindows and menus. The presentation logic of user interface is defined by the concrete implementation of IST Shell. The user interface enables employees to select objects defined by the business model and perform operations with them. The parameters, appropriate to the context of execution, are transferred to operations in standardized way. Operations with business objects are performed by help of IST DNS applications, Direpo and DS applications.

**IST DNS applications.** IST DNS applications are reusable and configurable applications, which can be used in all business domains. For example, application for business objects state transitions, which in collaboration with the user, by offering the user permissible state transition option of the given type object, provides transition of the object state received in the parameters. The current version of ISTechnology contains 15 reusable DNS applications

**IST Direpo application.** IST Direpo application provides preparation of reports according to the definitions established in the business model. The application interprets the report definition, obtains data form the database and formats them in the Rich Text Format document.

**IST DS applications.** DS applications are business domain specific applications, for example, currency buying applications. Within one business domain these applications may be reused directly or customized to specific organization business requirements.

## **3 Lessons Learned from the Use of ISTechnology**

### ***3.1 Objects of Case Study***

Let us consider the experience of use of the ISTechnology in the financial business domain (Table 1).

**Table 1.** IS developed by the ISTechnology

Type of Information Systems	Number of installations in different banks	Year of implementation
VOIS – currency operations IS	2	1995, 1996
PFS – pension fund IS	1	2000
VUS – security accounting IS	3	2002, 2008, 2009
LPS – resources management IS	3	2004, 2007, 2009

VOIS – ensures currency exchange and deposit deals. Conclusion of buying/selling contracts, control of different limits and registration of currency positions, confirmation of deals, corresponding payments in the SWIFT system and control of the payments are ensured. The processing of the deals is determined by the type and the risk of the deal - either the deal is concluded with the bank or the client of the bank.

PFS – ensures conclusion of contracts with the client, registration of the client pension fund, processing of pay-in/pay-out transactions. Input of the above mentioned data from the outer systems has been automated. Deposits of the participants of the pension fund and the employer of the participant, as well as the profit from investments are registered. Calculation of the taxes from the pensions paid has been automated.

VUS – ensures processing of security deals in the portfolios of the bank or the clients of the bank. Security registration in the client and bank security accounts as well as sending and processing of deals in outer systems (SWIFT, stock exchange, depository) are ensured. The system fixes the security prices and reevaluates the portfolios.

LPS – ensures investment fund portfolio management. Emitting and clearance of the fund parts, processing of security deals and cash payments, reevaluating of the fund actives and preparation of the balance, determination of the fund part value are ensured. Automated exchange of the deal data with the holding bank security registration system is ensured.

Business processes supported by the above mentioned IS traditionally are carried out by three departments:

- Front-office negotiating and enter deals (for example, buying or selling of currency, security, etc.).
- Back-office ensuring technical transactions of deals (for example, payments, using the SWIFT, etc.)
- Middle-office provides monitoring, control and analysis of the processes within the context of different bank investment portfolios, etc.

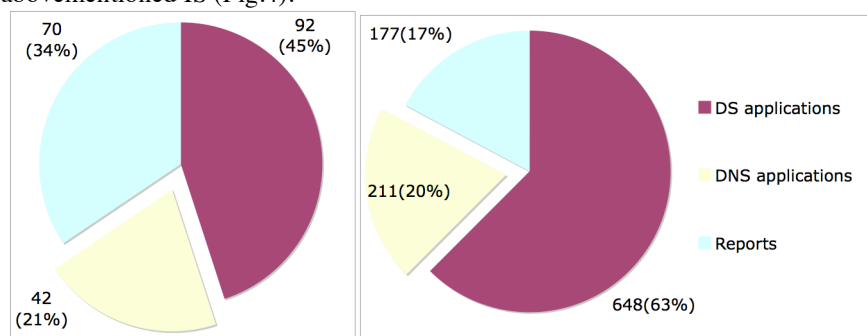
IS functionality appropriate to the bank business model is defined by the IST meta-model based platform independent model and IST applications. It should be mentioned that the business model of each bank is specific and, accordingly, IS



work places functionality and number of users in each bank are different. IST DS applications were reused directly or modified according to specific requirements of banks. It should be noted, that these applications are characterized by complicated calculation algorithms (calculations of portfolios currency positions, etc.) and non-functional requirements (extensive keyboard use in entering deals, etc.) For development of such applications it is necessary to use universal programming languages.

### 3.2 Development Effort Analysis

Let us consider the percentage of components of the ISTechnology in the abovementioned IS (Fig.4).



**Fig. 4:** The percentage of components in PFS and LPSIS on the left and in VOIS, VUS IS on the right.

PFS and LPS IS are functionally simpler, the number of applications is smaller and more than a half (55%) of functionality was provided by use of ISTechnology components. However, also in technically complicated VOIS and VUS IS almost a half (in total 37%) of the business functionality was provided without programming, i.e. by defining and configuring the business model and IST DNS applications.

The experience of use of the ISTechnology testifies the following development effort (Table 2):

**Table 2.** Estimated Effort

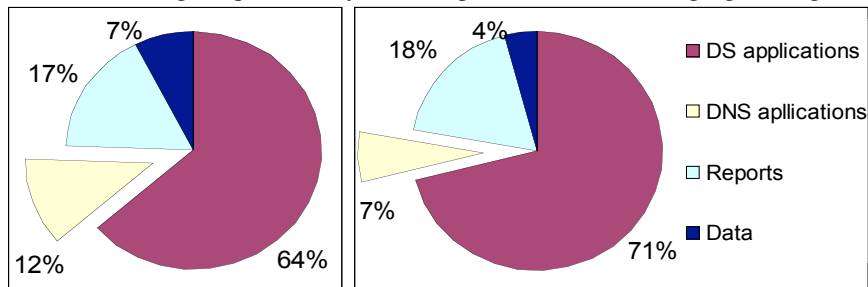
Development activity	Estimated effort (*)
Changes in business model by IST Shell or DNS application configuration for specific situation	2 – 8 wh
Report development (IST Direpo)	8 – 24 wh
Domain Specific Application Development	8 – 24 wh

(\*) measured in numbers of average work hours (wh)

From Table 2 it can be concluded that effort decreases 3-4 times, if the development activity is to be accomplished by changes in the organization business model or by configuring IST DNS applications. Taking into consideration the number of applications of various categories in the abovementioned IS, it can be concluded that development effort of these IS, in comparison with the situation, if all applications should be developed by programming, has decreased approximately 4 times.

### 3.3 Maintenance Effort Analysis

More than 600 change requests (CR) had been registered since 2004. The results of the change requests analysis are depicted in the following figure (Fig.5).



**Fig. 5:** The percentage of the total number of CR on the left and the percentage of consumed time on CR on the right

According to the experience of use of the ISTechnology, if in the IS maintenance process the change request is implemented by changing only the platform independent business model or IST DNS applications, the economy of effort is approximately 2 times (see Fig 5). However, as it might be seen, such change requests make approximately 12% of the total number of change requests. On the one hand it testifies, that the business modeling language is understandable enough for customers and enables to define this model quite precisely already at the moment of development. On the other hand it means that it is essential to widen the range of IST DNS applications and business modeling facilities.

## 4 Conclusions and Future work

This paper presents integrated approach to the development of the organization business model and appropriate IS. The experience obtained in practice testifies, that the ISTechnology meta-model can be successfully used in defining of the organization business model. It is easily perceptible to the users and enables to in-

involve users actively in definition, optimization and maintenance of the platform independent business model. The platform specific models are not described by meta-models, but there were ISTechnology applications for specific platforms developed. In that way, the business model is supported with appropriate IS functionality in specific platform.

The experience of the use of the ISTechnology shows, that approximately 40% of the IS functionality and 12% of the IS change requests are implementable by performing changes in the platform independent business model and in domain nonspecific applications. Taking into account, that effort of implementation of new functions and change requests in the business model and in domain nonspecific components are 2-4 times less effort-consuming, it can be concluded that the ISTechnology enables to economize considerably the effort and cost of IS development and maintenance.

The experience of use of the ISTechnology has outlined also the following most essential directions of development:

- Integration of various organization business models. Already now ISTechnology is used in situations, where, for example, one bank and its branch enterprises have several IS, which are supporting integrated business processes.
- Improvement of the meta-model, by providing more detailed definition of the business goals of organizations (answering the question „Why?“).
- Wider provision of IS functionality with domain nonspecific components. The analysis of the developed till now applications shows, that their number, in comparison with the existing one, can be increased by 20% and accordingly the development and maintenance effort of IS can be decreased significantly.

## **Acknowledgements**

This research was partially funded by the ESF project “Scientific staff for ESF project implementation”. ESF project No. 2009/0216/1DP/1.1.1.2.0/09/APIA/VIAA/044.

## **References**

1. Treimanis M.: ISTechnology – Technology Based Approach to Information system Development, Proceedings of the Third International Baltic Workshop “Databases and Information Systems”, Volume 2, Riga, 1998, pp. 76-90.
2. Iljins J.: Design Interpretation Principles in Development and Usage of Informative Systems, Scientific papers volume 669, University of Latvia, (2004) 99 – 106

3. Iljins J.: Metamodel Based Approach to IS Development and Lessons Learned, Databases and Information Systems Baltic DB&IS, (2008) 167 – 178.
4. Patrascioiu O.: Model Driven Language Engineering, A thesis submitted to The University of Kent for the degree of Doctor of Philosophy (2005)
5. Kalnins A.: Business Modelling. Languages and Tools. Progress in industrial mathematics at ECMI, Springer (2002)
6. OMG Unified Modeling Language™ (OMG UML), Superstructure, Version 2.2, <http://www.omg.org/spec/UML/2.2/Superstructure>, (2009)
7. Rumbaugh J. and Co: Object-Oriented Modeling and Design, Prentice-Hall, (1991)
8. Quatrani T.: Visual Modeling with Rational Rose 2000 and UML, Addison Wesley ISBN: 0-201-69961-3, (1999)
9. Kalnins A., Barzdins J., Kalis A.: GRADE-BM: Modeling and Simulation Facilities, Proc. of NWPER'96, Aalborg University, (1996) 71-86
10. IntalioCloud, <http://www.intalio.com/>
11. Web Services Business Process Execution Language Version 2.0, Public Review Draft, (2006), <http://docs.oasis-open.org/wsbpel/2.0/wsbpel-specification-draft.html>
12. Knöll H.D., Kuhl L.W.H., Kuhl R.W.A., Moreton R.: Optimising Business Performance With Standard Software, Birkhäuser ISBN 3528057653, 783528057657, (2001)
13. Dorner C., Draxler S., Pipek V., Wulf V.: End Users at the Bazaar: Designing Next-Generation Enterprise Resource Planning Systems. - IEEE Software, September/October 2009, pp. 45-51 Volume 26, Issue 5
14. Software Product Lines, <http://www.softwareproductlines.com/>
15. White J., Hill J.H., Tambe S., Gokhale A.S., Schmidt D.C., Improving Domain-Specific Language Reuse with Software Product Line Techniques, IEEE Software, Domain-Specific Modeling, (July, August 2009)
16. Clements P., Northrop L., Software Product Lines: Practices and Patterns, Addison-Wesley (2002).
17. Kleppe A., Warmer J., Bast W.: MDA Explained: The Model Driven Architecture™: Practice and Promise, Addison Wesley, ISBN: 0-321-19442-X, (2003)
18. MetaEdit+, <http://www.metacase.com/>
19. Sowa J.F., Zachman J.A.: Extending and formalizing the framework for information systems architecture - Technical, IBM Systems Journal, (1992)
20. Nalbone J., Ambler S.W., Vizdos M.J.: The Enterprise Business Modeling Discipline, Prentice Hall, (2005)