Business process modelling - methods and methodologies

Tomáš Rippl
Katedra informatiky a kvantitativních metod
Fakulta informatiky a managementu, Univerzita Hradec Králové
tomas.rippl@centrum.cz

Abstract
Business process modelling is a concept useful in a wide variety of situations and for many purposes. Above all, its products are invaluable for contemporary software development (both business people and IT developers gain benefits from using it).

Many software development methodologies treat business process modelling as a way of creating formalised view of a business domain, the purpose of which is to ensure a close (optimally perfect) alignment of IT solutions with the needs of a business. The product of BPM is a set of deliverables called "domain model". The model and the concepts used must be understandable by both developers and end-users.

Business process models are not only useful for producing software that is aligned with the requirements of the business but it can also serve as a useful part of the knowledge management, cost analysis, business process analysis, etc.

The purpose of this work is to introduce and highlight the main ideas and principles of two methodologies and IT systems development with the aim at business process modelling concepts: Select Perspective (from Select Business Solutions) and ARIS (from IDS Scheer, AG). The final part concentrates on the description of main differences between these two approaches, evaluates the benefits of each and suggests the area at which usage of these methodologies is suitable.

Keywords: Select Perspective, ARIS, Business process modelling

1. Select Perspective
"Every business faces constant change, and these changes increase the pressure on IT development to deliver successful solutions. Changes to the business make new demands on IT developers to respond quickly to the opportunities or threats to the core business. Software development is measured in weeks rather than months, which means that the development process must be responsive to the different needs of each project. The new approach must be agile".

Select Perspective (SP) is a product of Select Business Solutions (formerly Select Software Tools), one of the three divisions of Aonix, a member of Gores Technology Group. SP is about managing software assets. It is a blend of improved and adapted object-oriented methods. The syntax used is based on Unified Modelling Language (OMG UML). SP is aimed at building adaptable systems (i.e. systems prepared and designed for change), which is reached by the use of components (encapsulated set of services with clearly defined interfaces and with minimal interdependency among components). SP is compact, highly practical, delivery-focused approach to component-based software development.
1.1 Objects and components

SP moved to component approach in 1997, when object technology and object-based development became the preferred approach to delivering modern solutions, but some of the promised benefits were hard to realise (e.g. reuse of objects). Components encapsulate (contain) more functionality than objects and allow higher level of reuse, which can substantially increase the speed and reduce the error rate. They are relatively small units of deployment with a published interface consisting of services definitions (signatures). There are several benefits of using components and performing component-based development (CBD):

1. technical benefits
   - simplicity is reached through abstraction
   - protected boundary is used
   - functionality encapsulation
   - possibility of substitution (versioning)

2. business benefits
   - possibility of parallel development
   - solution is more about assembly and less about development, which implies faster development cycles

3. economic benefits
   - good components can be sold / bought at a good price

1.2 Tools

During a CBD project numeric types of tools are used. The most comprehensive ones are component factories, which encompass business process modelling, requirements management, visual modelling, component management, quality assurance, application assembly, and deployment. Aonix, Web Gain, and Computer Associates are examples of such tools vendors. Moreover, different kinds of component managers, visual modelling tools, requirements management, application assemblers, test managers, application animators, patterns, O-R (object-relation) mappings, and project / process management and rollout tools are used through the CBD.

Similarly to SP, which is “a blend of object-oriented methods” methodology, the Select Component Factory and Select Process Director (the main toolsets produced by Select Business Solution, are blend of several tools from different categories mentioned above assembled for the purpose of providing an integrated environment for organisation and business process modelling, and full-value CBD.

SP categorised CBD work-flows into three distinct groups of activities: supply, manage, and consume.

The purpose of the SUPPLY work-flow is to build and deliver complete components usually according to predefined component specification (produced by the supplier or by the consumer - i.e. “design by contract”). The component can be re-used, rented, bought, or built from scratch. The schema of SUPPLY workflow can be seen at Picture 1:
Main SUPPLY deliverables are:
- component/service specification
- component/service model
- component executable
- component package

The MANAGE work-flow (the component management) is aimed at component or service supply (whether they are internal or external to the organisation), and to monitor components and service re-use. The main purposes (benefits) of re-use are: work save, reduced number of errors, reduced maintenance costs, and reduced re-work. The schema of MANAGE work-flow can be seen at Picture 2:
Main MANAGE deliverables are:
- component / service catalogue
- component / service specification
- candidate component / service
- certified component / service

The CONSUME work-flow

From the business process modelling (BPM) point of view, the CONSUME work-flow is the most important part of SP. At the start of each project there are business users’ requirements. At the end of a project there is (hopefully) a delivery of one or more business solutions. One of the critical success factors is the alignment of the solution delivered with underlying business needs.

As can be seen at Picture 3 (the CONSUME work-flow scheme), there are seven (five major and two administrative) work-flows:

*Figure 3: Select Perspective - The CONSUME work-flow*

Let's concentrate on the Business Alignment work-flow. The purpose of this work-flow is to ensure the full definition of requirements. As the outputs must be understandable for both business people and IT personnel, a formalised view with small and easy-to-understand set of symbols is used. With the help of these concepts, the boundary of IT solution is identified and requirements are defined. The main output of the CONSUME work-flow is the domain model, which consists of business process model, use case model, business rules catalogue, non-functional requirements, and definitions of constraints.
The Business Alignment work-flow is an iterative and incremental process, at which communication between end-users, business analysts, and developers is essential. The aim is to reach and maintain the alignment with business and its requirements and the software solution that is to be produced. If the solution is to support business processes, the BPM is a good and proven instrument that can help. SP defines two kind of models: 1) process hierarchy diagram (PHD), which captures the static structure of processes, and 2) process thread diagram (PTD), which illustrates flows of work through activities and tasks. Moreover, there are two possible types of models: as-is (which says what happens in the business now and which is usually created first), and to-be (which depicts new activities, work-flows etc. necessary for the new process. It is usually created to fix prioritised issues). Creation of an as-is BPM often helps to uncover and clarify the business issues. The to-be BPM shows how business changes will be implemented in the organisation. The result of this activity is a program of change: business change and system change. The requirements regarding information, business rules, constraints, and non-functional requirements can be discovered from a detail BPM. Without capturing and exactly formulating these, the business alignment of the final solution would be endangered.

1.3 Summary

Select Perspective is a proven, pragmatic approach to delivering successful software solutions in a wide variety of context. The benefit of the software solution is measured by two indicators: 1) whether it offers useful services (services that somebody in the organisation needs and uses), and 2) whether it provides those services at a required level. Business process modelling instance employed by the Select Perspective products a domain model of a business reality and business requirements, that can - if created thoroughly and properly - ensure achievement of required level in both indicators.
2. ARIS

Modelling of business processes and all relating factors and domains (holistic view of business and its processes) is seen as a critical and decisive competitive factor by ARIS (Architecture of Integrated Information Systems). "A company needs to recognise, streamline, and support interconnected processes through optimised information systems environment. Business process models are a crucial prerequisite for analysing business processes, bringing projects in line with the overall company objectives, and finally for finding the perfect information structures in the form of a compound of distributed, integrated systems to support these lean organisational structures."

ARIS recognises that increasing number of quite different modelling methods is available, which on the one side supports the idea of importance of BPM but on the other side brings increasing complexity and confusion. As a consequence, efforts have been made to define standardised general concepts (so-called architectures) for IT systems and modelling methods development. ARIS has been developed by professor A. W. Scheer and is a typical example of such an architecture. As it is a general concept, it allows to develop modelling methods, and it also serves as an orientation framework for complex development projects because in its structuring elements it contains an implicit procedural model for the development of integrated information systems.

2.1 Concept of ARIS architecture

The design of ARIS is based on an integration concept which is derived from a holistic analysis of business processes. When modelling business processes comprehensively, a very complex model may be the result. To reduce such complexity, the architecture of ARIS comprises several different views, whose mutual relationships are strictly defined. A second approach of reducing complexity is the analysis of different descriptive levels (from the point of view of models' proximity to information technology). This ensures a consistent description from business management-related problems all the way down to their technical implementation.

2.2 Descriptive views

The components necessary to provide a full description of a business process are thus procedures, events, products/services (statuses), processors, organisational units and information technology resources. All those aspects are separated to groups according to the criteria specified by the methodology. The principle is to minimise the relations between views and permit necessary relations between the components in single views. The general schema of ARIS architecture and the relationships among single views is depicted on Picture 5:
Organisational view represents a combination of the users and the organisational units as well as their relationships and structures. In data view information objects and changes in them are described. Functions (activities) result in performances - products / services; they are described together with their relationships in product / service view. Processes (the functions to be performed) are described in the function view. Control view is the additional view which describes the relationships between other views and thus forms the integrating part of the methodology.

2.3 Descriptive levels

Descriptive levels are based on the different levels of abstraction regarding to the usage of concrete information systems to support business processes realisation. The result is a three-tier division illustrated at Picture 6:
At the level of operational business problem rough facts and technology options are described. At the next level facts are translated into semantic models using a formalised description language. In the design specification level the requirements are translated into IT modules and transactions description. At the implementation level, the design specification is transferred to concrete hardware and software components.

The essence of ARIS architecture is the combination of descriptive views and descriptive levels, as can be seen at Picture 7:
2.4 Modelling

Operational business problem definition

The purpose of this phase is to describe the initial semantic business process, i.e. the business problem before the individual descriptive objects within the ARIS architecture (views and levels) can be modelled. The model expressing the problem description needs to cover as many facts as possible from the data, function and organisational structure views including the interrelationships existing between them.

The necessary interrelationships are best recorded in condensed form as process chain diagrams (PCDs) which also provide an overview of the information system that is to be dealt with. When analysing business processes in a process chain diagram that describes an actual situation, weak points in the current problem solution can be highlighted. These weak points can be either media breaks between IT-related and manual processing or organisational breaks (e.g. the department in charge / organisational unit is changing frequently). The analysis particularly shows data redundancies, multiple entries and time delays within a procedure, making it possible to derive many possible ideas for improving the target procedure that is being defined.
Function view - Requirements definition

A function is a technical task or activity performed on an object in support of one or more company objectives. Usually, the criterion for establishing functions are information objects such as a customer inquiry or a production order.

Function tree

Complex functions defined at a high compression level can be divided into sub-functions to reduce their complexity. "Elementary functions are functions which cannot be reduced any further for the purpose of business process analysis."

Function trees (or hierarchy diagrams) can be used to represent the sub-structuring. Function trees help to reduce complexity but illustrate only a static representation.

Y diagram

The Y diagram is used to represent the major functional areas of a company on a highly aggregated level.

SAP application diagram

The SAP applications diagram permits an approach to the SAP R/3 reference model that is geared to the SAP R/3 application system modules.

Objective diagram

This type of diagram allows to define the objectives of the activity of BPM / BPR. Target and target hierarchies can be defined. A target defines future company objectives, which are to be achieved by supporting critical factors and creating new business processes. Critical factors specify the aspects which need to be considered in order to achieve a particular company target. They are assigned to company targets in the objective diagram.

Function view - Design specification

The design specification of the function view contains the specification for the application system and module types, for the modular structure, for the draft of the individual transaction steps, and the definition of input and output presentations in the form of lists and screen drafts. "An Application system type typifies individual application systems which have exactly the same technological properties."

Application system type diagram

This diagram enables the display of the modular structure of application systems type. The individual parts of an application system type are module types. "A module type represents a component of an application system type which can be executed independently. Module types represent the typification of individual modules which have exactly the same technological basis. IT function types are the smallest units of a module type for the purposes of a transaction. They are produced by individual program modules and must always be carried out completely to process an individual work step."

Function view - Implementation

The purpose of this view is to show the actual modular structure of an application system.
Application system diagram
In this diagram the specific application systems and modules can be assigned to the application system and module types described in the design specification. "An application system (module) is a single copy of an application system type (module type) which can be uniquely identified, by its license number for example."

Data view - Requirements definition
Data model
The requirements definition of the data view includes a description of the semantic data model of the field which is to be examined. According to the ARIS division principle this description contains both the objects which specify the start and end events of a process chain as well as the status descriptions of a process chain's relevant environment. The most widely used designing method for semantic data models is Chen's entity-relationship model (ERM)). ARIS adds some constructs to the original model and creates eERM (enhanced ERM).

"Entities are real or abstract things that are of interest for those tasks in a company that are currently under consideration. If entities of the same type are grouped into sets, they are called entity types. The individual occurrences of these are entities. Attributes are properties describing entity types." A relationship is a logical link between entities. Hence, the existence of relationships directly depends on the existence of entities. If relationships of the same kind are grouped into sets, they are called relationship types.

Relationship types are distinguished according to the number of entity types linked to them, i.e. unary, binary or n-ary relationships. The complexity (or cardinality) indicates how many entities of one entity type can be assigned to an entity of the other entity type. The cardinality operators defined in ARIS eERM are slightly different from the original Chen's ERM. The difference brings the advantage in the form of less confusion especially in n-ary relationships.

The value ranges of attributes are called domains. A 1:1 relationship must exist between one entity type and at least one domain. The values of this domain uniquely identify the individual entities. Therefore, they are called the key attributes of the entity type.

As has been already written, ARIS extends the classical Chen's ERM especially in the field of design operators - classification, generalisation, aggregation, and grouping are the four main design operators. The most interesting extension is in the aggregation part where reinterpreted relationships (entity types that originally were relationship types) is introduced. Moreover, data clusters describing the logical view onto a number of entity and relationship types of a data model that are needed in the description of a complex object are used.

Technical term model
This model helps to capture the variety of terms defining information objects and is especially useful in larger companies.

eERM Attribute Allocation Diagram
This model is used to mitigate the complexity of eERM by means of separating objects' attributes to a special model.
Alternative forms of representation are SAP SERM (structured entity-relationship model), IE (information engineering) data model, SeDaM (semantic data model) etc. Among other tools, models, and diagrams used are Document type definition (DTD) model, Material flow diagram, Data Warehouse Structure Model, Authorisation hierarchy, and Project management Data model.

**Data view - design specification**

In the design specification, the logical data structures designed in the requirements definition are transformed into a form of description on which concrete database systems can be built. ARIS provides the relational model for this. Relation diagram and Attribute allocation diagram are used.

**Data view - implementation**

In this phase a table diagram is created in which a database system's tables and fields can be described.

**Organisation view - Requirements definition**

Companies are complex social structures which are divided into handy units. To deal with the complexity, patterns are defined and rules are established. The result of this process is called organisation.

**Organisation chart**

The organisation chart is used to describe the structure of organisational units (as task performers) and their interrelationships, depending on the selected structuring criteria.

**Shift Calendar**

Shift calendars can be assigned to personnel and material resources, and specify when a resource is mainly available.

**Organisation view - Design specification**

The company's organisational structure as represented in the organisational chart should be supported using communication and information system landscapes.

**Network topology**

The structural requirements for the information systems can generally be defined in the design specification in the form of network topologies.

**Organisation view - Implementation**

**Network diagram**

The network diagram is used to illustrate the actual realisation of the network topology determined in the design specification.

**Material flow modelling - technical resources**

**Process View/Control View - Requirements Definition**

The relationships between the objects of the data organisation and function views are analysed in the control / process view.
Combining functions with organisations
The link between the function view and the organisation view is used to assign the functions defined in the function tree to the task performers (organisational units) in the organisational chart. This assignment defines an organisational unit's responsibility and decision-making power for its allocated functions. Looking at this organisational allocation in the course of a process chain (business processes) the degree of functional integration is defined, i.e. the functional steps within a business process that are to be processed by an organisational unit.

Combining functions with data - Event Control: Event-Driven Process Chain (EPC)
The procedural sequence of functions in the sense of business processes is represented in process chains. This means that the start and end events can be specified for each function. Events not only trigger functions but are also results of functions.

Combining functions, organisations, and data
In this model, several diagrams and tools are used: eEPC / PCD, Value Added Chain Diagram, Rule Diagram, Communications Diagram, Classifications Diagram, and Input / Output diagram.

Object-oriented modelling
The ARIS class diagram allows to assign the role of classes to information objects (clusters, entity types, relationship types).

Other models
Process selection matrix, Material flow modelling - eEPC (material flow), SAP ALE models, Role allocation diagrams, and many other models can be used in this view and at this level.

Process View/Control View - Design Specification
Access diagram
The relationships that have been explained in the design specification descriptions of the other views can be included in the access diagram of the control view.

Program structure diagram
This diagram allows all relationships to application system types, module types and IT function types provided by the other model types of ARIS to be modelled regardless of the ARIS division into views.

Program flow chart
The program flow chart (PF) is used to show the procedural sequences of a program. The processing sequences are shown by the relationships between the objects. This diagram does not represent any data.

Screen Diagram
A screen diagram is used for describing screens during software development. The goal of this model is an automatic derivation of screen diagrams from the diagram.

Process View/Control View - Implementation
Access diagram (physical)
This type of diagram helps address and capture relationships that exist between concrete application systems and organisational units, for example, and not those between application system types and organisational units.

**Product / Service View**
Products can be a consumable product, a material type, an operating resource type, a technical operating supply type or a packaging material type. The trigger for the creation of a product / service is always the need of an organisational unit or a customer. ARIS provides various model types for describing the products and services performed in a company.

**Product / Service Exchange Diagram**
The product / service exchange diagram is used to map the creation of products / services as well as their exchange within the company.

**Product / Service Tree**
Products / Services can be viewed at different levels of abstraction. It is therefore useful to store these relationships in a model showing the partial products / services that make up a complete product / service. This static aspect is represented in the product / service tree.

**Product Allocation Diagram**
In addition to the general product / service diagrams which belong to the graphic models, the product models offer the possibility of creating a more abstract representation. The product allocation diagram is primarily used to analyse product creation in public administration. Like the product / service exchange diagram, this model type can be used to show which organisational units provide or use which products, and which functions are required for the creation of the products, or for which functions consume the products as an input.

**Product Tree**
The purpose of the product tree is to analyse the composition of products in public administration. This model corresponds essentially to the product / service tree, whereby the possibility of modelling replacement products is dispensed with. The product tree is located at the requirements definition level of the product / service view.

**Product Selection Matrix**
In the product selection matrix the focus is on an organisational unit and the products that fall within its responsibility.

**Competition Model**
This model supports the analysis and evaluation of the competitive environment in which the company competes. The industry structure strongly influences the strategies which are potentially available to the company.

### 2.5 Tools
ARIS Toolset is a family of tools developed by IDS Scheer AG. It is based on ARIS architecture and it consists of relatively independent tools, such as ARIS Easy Design for BPM, ARIS ABC, ARIS Simulation, ARIS Web Publisher, or ARIS Lotus Notes Connectivity.
2.6 Summary
To sum up, the ARIS architecture forms the framework for the development and optimisation of integrated information systems as well as a description of their implementation. In this context, with the emphasis on the technical descriptive level, the ARIS concept serves as a model for creating, analysing, and evaluating business management process chains.

3. Confrontation of approaches
Comparing the two approaches, several apparent differences can be found.

3.1 Paradigm
When comparing the basic concepts of both methodologies, the first obvious difference is a different basic paradigm - Perspective is object oriented while ARIS is a typical example of a structured methodology. The object orientation in Perspective is even more apparent with the use of components, where the degree of abstractness, encapsulation, and possibility of re-use is by one level higher when comparing with the classic object-oriented approach.

In ARIS, the structural approach is apparent from the first sight at the HOBE. Products, organisations, and especially functions and data views are separated, modelled, and then interconnected again in the control / process view to represent the business process in a systematic way. It is true that ARIS allows to use concepts and diagrams that are typically used in object-oriented methodologies and languages (UML for example) but the concepts are always created by aggregation of entities defined in separated HOBE descriptive views at different descriptive levels. Moreover, ARIS mostly uses break-down structure diagrams created by top-down analysis.

3.2 Scope
In comparison with ARIS, Perspective is much more specialised. ARIS considers itself not only as a methodology but also as a general framework (architecture) for creating methodologies and methods of business process modelling / managing. Perspective is an agile methodology strictly aimed at IS development, ARIS is a general framework covering all possible areas of business analysis. It contains a large number of tools with rules for their usage and their relationship description. It reminds a UML but it is much broader, which is derived from the HOBE concept. The intention of ARIS is the modelling, association, measuring, and optimising (or eliminating) of every relevant thing / person / entity / process / activity.

3.3 Purpose and scope
Select Perspective is aimed at finding the right way to develop adaptable and manageable information systems that effectively support dynamically changing business processes. To deal with changes and requirements on the speed of delivery, Perspective relies upon the component-based approach / development. Business process modelling (and a domain model as a product of this activity) is iteratively used / produced in increments to ensure the alignment with the needs of
business. The main purpose of BPM in Perspective is to define the scope of software support and business requirements that not only grasp the current state but also continually re-adjust themselves according to the changes in business environment.

ARIS considers the holistic management of business processes as a critical factor of organisation performance and competitiveness. To reduce the apparent complexity of a holistic business process model, ARIS introduces descriptive views and descriptive levels, which are very similar (if not identical) with the concept of three architectures. What is new is the interconnection (integration) of descriptive views and descriptive levels, the product of which is the extended house of business engineering (HOBE). The purpose of business process modelling in ARIS is to capture the current state of organisational processes, the optimisation / streamlining of them, and to help finding the right integrated architecture of information systems that effectively support defined business processes.

References

