Studies in Computational Intelligence 518

Zdzisław Kowalczuk
Cezary Orłowski

Advanced Modeling of Management Processes in Information Technology

Springer

Studies in Computational Intelligence

Volume 518

Series Editor

Janusz Kacprzyk, Warsaw, Poland

For further volumes:
http://www.springer.com/series/7092

Zdzisław Kowalczuk · Cezary Orłowski

Advanced Modeling of Management Processes in Information Technology

Springer

Zdzisław Kowalczuk
Department of Decision Systems
Faculty of Electronics, Telecommunications
and Informatics
Gdansk University of Technology
Gdansk
Poland

Cezary Orłowski
Department of Information
Technology Management
Faculty of Management and
Economics
Gdansk University of Technology
Gdansk
Poland

ISSN 1860-949X ISSN 1860-9503 (electronic)
ISBN 978-3-642-40876-2 ISBN 978-3-642-40877-9 (eBook)
DOI 10.1007/978-3-642-40877-9
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013948657

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

This book deals with the issues of modelling the management processes of information technology and IT projects and its principal subject is the model of *information technology management* (MITM) and its component models (contextual, local) describing initial processing (IPP) and the Client – Supplier/Provider – Project (C-S-P) *maturity capsule* as well as a decision-making system represented by a *multi-level sequential model* (MSM) of IT technology selection, which eventually acquires a fuzzy rule-based implementation.

In this work, we assume that the process of making or manufacture of the product, which is the software, is implemented under the project of software preparation (called also *software development*) covering software *manufacturing* and *management* processes.

In this work, it is also assumed that the *provider organisation* (*project team*, *provider team*, *manufacturing team*) is responsible for the development and implementation processes of enterprise architecture, while the *client organisation* is primarily responsible for supplying the requirements for the process of the development and implementation of enterprise architecture.

The client organisation usually delegates a so-called *client team* consisting of a group of members that realises the IT project together with the provider team/organisation. The selection of the client team members is often supervised by the provider organisation. The situation is similar with the *provider team,* whose members are selected from the provider organisation and whose aim is to realise the project in cooperation with the client team/organisation.

The MITM management model is verified and modified during the course of the project. To verify its applicability, real data will be applied corresponding to the project's variables, among which are: quality of the provider and client organisations as well as quality of the project organisation, referred to in this work as the (C-S-P) maturity capsule. The capsule will in fact constitute the basis for the evaluation of the level of project management and of the MITM model.

The applicability of the model will be verified in a number of environments, mainly in IT projects and in the organisations carrying out such projects. In particular, there will also be materials[1] obtained from three IT standards: COBIT (version 4.1), ITIL (version 3.0) and TOGAF (version 8.1).

The knowledge resulting from the COBIT and ITIL standards is important for the initial processing of the project data, which aims at evaluating the maturity of the provider and client organisations in question. The TOGAF standard will be used mainly to evaluate a specific indicator, measuring the degree of global maturity of a project, called *negentropy*.

In describing the processes of enterprise architecture development and management, and of the maturity of the client and provider organisations, basic mathematical tools (matrices and vectors) will be used as well as a linguistic approach, which increases the utility of developed models. In this way, not only are the essential concepts organised for the use of project team leaders, but also the implementation of IT standards is supported and areas of their application in the selection of information technology are indicated.

The modelled MITM management processes, including initial processes (IPP), will evaluate the applicability of well-known measures and rules of conduct for team leaders, which have been described in these standards.

In general, it can be stated that any attempts to model IT processes inevitably bring to mind Lotfi A. Zadeh's famous statement about the mutual exclusion of the precision and significance of a description in a given field. In the analysed case, the development of such a model which would be useful for team leaders and at the same time in sync with the requirements of standards in IT management also proved to be very difficult. However, a compromise solution was achieved in terms of both complexity and utilitarianism, which is in line (harmonised) with IT standards for evaluating organisation maturity and project (scalar) negentropy.

When analysing types of information technology management (*ITM*) and the use of data obtained from IT standards in the modelling of data processing, the role of fairly general descriptions of IT standards, which were of limited use, prepared by international consortia, became questionable. Those responsible for preparing such descriptions can in their defence say that general standards can be applied to any manufacturing organisation. The question remains, however, whether this over-generalisation leads to a loss of applicability and practicality. As the conducted studies show, each standard – prior to its application – must be adapted to the needs of a particular organisation. The time spent learning and adapting these standards is significant, while continuous changes in standards undermine their utility.

It is worth noting that when the selection of a technology is based on knowledge acquired from experts and/or standards, there is a need to evaluate its degree of certainty.

In IT projects the COBIT and ITIL standards can be used interchangeably, both to evaluate the organisation of the provider and that of the client. However, nowadays when the IT world is focused around services, the ITIL concept is much better for the provider organisation, although there are still organisations that continue to run under the old assumptions, using COBIT to evaluate provider organisations.

---

[1] These are included in the appendices in accordance with their applicability in the specification of model variables.

Therefore, in this book, the ITIL standard will be used to assess the provider organisation and the COBIT standard to assess the client organisation. This solution is closely related to the experience and the results obtained in studies carried out at the Gdansk University of Technology for the benefit of third parties.

After applying the TOGAF standard, its new (9.1) version was released. During the evaluation of its suitability, certain aspects of the standard already proved outdated on the IT market. However, the model which has been developed based on a linguistic description has a universal value and is useful for application and modification. Thus, this proves the validity and utility of formal descriptions of IT standards which allow for the adaptation of the evaluation of an IT organisation or project to changing requirements.

This study, which aims to identify the methods and directions of research on IT evaluation, relies on project experience in information technology management. In terms of applicability, this work is also significant for diagnosing the applicability of IT standards in the evaluation of IT organisations. Unfortunately, this diagnosis is not optimistic, and suggests a need for better preparation of these standards. In view of the knowledge acquired, which is also documented in this book, the evaluation of an organisation based on the use of IT standards should not only be examined according to the marketing result, but also according to the suitability of standards for the evaluation or self-evaluation of the client and provider organisations.

The results of this analysis may prove to be valid for those who will prepare new standards so that - apart from their own visions – they could, to an even greater extent, take into account the capabilities and needs of the leaders of project and manufacturing teams.

Keynote: *This book is primarily concerned with the problem of modelling the management of technology. However, at the present stage of information technology development, IT itself is used in the management of projects. With the maturity of projects dynamically changing, their management is more and more associated with the management of technology. Thus there is a need for the effective modelling of both project maturity, as well as technology management. This thought is the central theme of the book.*

The material presented in this book is largely the result of the implementation of two research projects: "The study of information technology using smart systems" (Ministry of Science and Higher Education) and "The evolution of the IT support organisation of a bank" (GE Money Bank[2]), carried out between 2007 and 2010.

We would like to thank our colleagues from the Department of Information Technology Management at the Technical University of Gdansk, especially Adam Czarnecki, MSc, Tomasz Sitek, PhD, and Artur Ziółkowski, PhD, for their help in gathering material for this work. The support of our families and friends was also invaluable.

[2] Currently Bank Przemysłowo-Handlowy.

The book is intended for IT professionals using the ITIL, COBIT and TOGAF standards in their work. Students of computer science and management who are interested in the issue of IT project and technology management will also benefit from this study. For young students of IT, it can serve as a valuable source of knowledge in the field of information technology evaluation. This book is also dedicated to specialists in modelling socio-technical systems. Since this work is strongly based on the knowledge of IT standards, it is advised that the reader who is less familiar with this field should begin by studying the three appendices introducing this topic.

Zdzisław Kowalczuk
Cezary Orłowski

Contents

List of the Most Important Symbols and Abbreviations

a_t	–	primary negentropy variable, the relative width of the repository of architectures
ar_t	–	(scalar) variable of the size of artifacts (interpretation of the expertise level he_t)
bt_t	–	variable of the extent of realisation of the project budget
$\mathbf{c}_t$	–	vector variable of client maturity
c_t	–	scalar evaluation of client maturity
d_t	–	primary negentropy variable, the relative length of the documentation catalogue
hcc_t	–	variable of the level of client competence suitability
hcm_t	–	variable of the level of client matching
he_t	–	variable of the level of knowledge and understanding of the project area (height/level of expertise)
hmd_t	–	variable of the level/height of other processes of management and architecture development (TOGAF)
hmm_t	–	variable of the height of the applied IT project management methods
hmt_t	–	variable of the height of advanced IT tools in IT venture/project management
$\mathbf{hT}_t$	–	two-dimensional evaluation of the level/height of advanced IT technology in IT venture management
hT_t	–	scalar evaluation of the level/height of advanced technologies (tools hmt_t and methods hmm_t) in IT venture management ($hT_t = pr_t$). In a starting phase (before using negentropy), $hT_t = pr_t = rd_t$ can be applied by Project Managers.
$\mathbf{IT\_proj}_t$	–	IT project (venture) described using vectors
$\overline{\mathbf{p}}_t$	–	three-dimensional (primary) negentropy of an IT project
$\mathbf{p}_t$	–	two-dimensional (secondary) negentropy of an IT project
p_t	–	scalar (tertiary) negentropy of an IT project

pd_ADM_t – variable of the level of ADM development processes
pd_Cont_t – variable of the level of (support) processes of developing structures of Continuum
pm_ADM_t – variable of the level of management of the (proper) ADM development process
pm_Cont_t – variable of the level of managing the support process of Continuum development
pr_t – variable of primary negentropy, the height/level of the ITM section ($pr_t = hT_t$)
$psch_t$ – variable of the extent of realisation of the project schedule
rd_t – variable describing the degree of project completion in terms of management
$\boldsymbol{s}_t$ – vector variable of provider (supplier) organisation maturity
s_t – scalar variable of provider/supplier organisation maturity
t – independent variable – the current duration of the project realization
t_{end} – project end/completion time
$\mathbf{z}_t$ – multi-dimensional evaluation of the level of IT project management
z_t – scalar evaluation of the level of IT venture/project management
z_t^c – sub-level of management resulting from client maturity ($z_t^c = c_t$)
z_t^s – sub-level of management described by supplier/provider maturity ($z_t^s = s_t$)
z_t^p – sub-level of management related to project negentropy ($z_t^p = p_t$)
P_ψ – function of project management functionality growth (according to $\mathbf{z}_t$)
P_ω – function of the transfer (level-change) of the quality of project management
P_ϕ – function of an increase in the level of project management
$\mathcal{R}_{[0,1]}$ – set of real numbers from a closed interval [0, 1]
$\mathbf{\Pi}$ – methodological transformation matrix of the effect/impact of growth in the maturity capsule C-S-P (of the Client, the Supplier/Provider, and the Project organizations) on the increase in management level
α_i – factor $\alpha_i \in \mathcal{R}_{[0,1]}$ of the impact of client maturity; a variable of the matrix $\mathbf{\Pi}$
β_i – factor $\beta_i \in \mathcal{R}_{[0,1]}$ of the impact of provider maturity used in the transformation $\mathbf{\Pi}$
γ_i – factor of the impact of (secondary) project negentropy $\gamma_i \in \mathcal{R}_{[0,1]}$ applied in $\mathbf{\Pi}$

$\Delta\ a_t$ – variable of the increase in the width of the architecture repository
$\Delta\ ai_t$ – variable of the growth of organisation maturity in the domain of acquisition and implementation
$\Delta\ am_t$ – variable of the growth of organisation maturity in the domain of application management
Δar_t – (scalar) variable of the growth of project artifacts
$\Delta \mathbf{ar}_t$ – vector of growth in project artifacts
$\Delta\ bp_t$ – variable of the growth of organisation maturity in (the domain of) business perspectives
$\Delta\ d_t$ – variable of the increase in the length of the documentation catalogue
$\Delta\ ds_t$ – variable of the growth of organisation maturity in (the domain of) (delivering and supporting)
Δgpd_t – variable of the growth of good practices in (the field of) development
Δgpm_t – variable of the growth of good practices in (the field of) management
ΔhdT_t – variable of the increase in the height of development technology
ΔhmT_t – variable of the increase in the height of management technology
$\Delta\ ict_t$ – variable of the growth of organisation maturity in (the domain of) infrastructure management
Δka_t – variable of the growth in processes located in key areas
$\Delta\ m_t$ – variable of the growth of organisation maturity in the domain of monitoring and evaluation
Δpd_t – variable of growth in the number of development/manufacturing processes
$\Delta\ pism_t$ – variable of the growth of organisation maturity in the domain of service management planning
Δpm_t – variable of growth in the number of management processes
Δpr_t – variable of growth in the height of the ITM section
$\Delta \mathbf{pr}_t$ – vector of growth in the height of the ITM section
$\Delta\ sm_t$ – variable of the growth of organisation maturity in the domain of safety management
$\Delta\ sp_t$ – variable of the growth of organisation maturity in the domain of providing services
$\Delta\ ss_t$ – variable of the growth of organisation maturity in the domain of service support
$\Delta \mathbf{y}_t$ – vector of the growth of the functionality of IT technologies

ADM – *Architecture Development Method*
Apache Tomcat – Web application server type
API – *Application Protocol Interface*, functionality type of IT systems
ASP – *Active Server Pages*, name of an application of the .NET framework library
BaselII – name of the law passed by the Brazilian Committee on Banking Supervision
BI – *Business Intelligence*, type of IT systems
Blue Cloud – IBM model of service support through data centres
BPEL – *Business Processes Execution Language*
BPML – *Business Processes Modelling Language*
CAO – *Chief Architecture Officer*, the chief/project architect
CBMBoIT – *Component Business Model for the Business of IT*, model of business processes
CFS – *Customer Fact Sheet*, spreadsheet tool to assess customer requirements
CIO – *Chief Information Officer*
CMMI – *Capability Maturity Model Integration*
COCOMO II – *Cost Construction Model*
COBIT – *Control Objectives for Information and Related Technology*, IT standard
COM – *Component Object Model*, architecture type
Continuum – name of the project repository according to TOGAF
CORBA – *Common Object Request Broker Architecture*
COTS – *Commercial off-the-shelf*, type of software components
CRM – *Customer Relationship Management*, type of IT system for planning customer relationships
CRUD – *Create, Read, Update, Delete*, type of matrix
C-S-P – *Client–Supplier(Provider)–Project*
DBMS – *Database Management Systems*
EA – *Enterprise Architecture*
Eclipse – application development framework for client-server architecture
EJB – *Enterprise Java Beans*, IT systems architecture type
ERD – *Entity Relationship Diagrams*
ERP – *Enterprise Resource Planning*
FHD – *Function Hierarchical Diagrams*
Governance – management process of specific architecture development (*e.g.* SOA, EA)
HR – *Human Resources*
ICA – *Independent Computing Architecture*, type of protocol
ICAM – *Integrated Computer Aided Manufacturing*, type of models focused on manufacturing/development processes

III-RM	–	*Integrated Information Infrastructure Reference Model*
IPA	–	*Information Provider Application*
IPP	–	*Initial Processing Procedures* of project data (& client/provider companies)
ITIL	–	*Information Technology Infrastructure Library*, IT standard
ITM	–	*Information Technology Management*
J2EE	–	development platform based on Java programming language
KGI	–	*Key Goal Indicators*
KPI	–	*Key Performance Indicators*
LAIP	–	*Liberate Application of Information Provider*
MDA	–	*Model Driven Architecture*
MITM		*Model of Information Technology Management*
MSF	–	*Microsoft Solutions Framework*, project management environment
MSM	–	*Multi-level Sequential Model* of selecting IT technologies
OLAs	–	*Operational Level Agreements*, type of service contracts
PRINCE	–	*Project in Controlled Environments*, project management method
PSL/PSA	–	*Problem Statement Language / Problem Statement Analyser*, structural description language
SCAMPI	–	*The Standard CMMI Appraisal Method for Process Improvement*, method of supporting the organisation maturity evaluation process
SCM	–	*Supply Chain Management*, system type
SCRUM	–	project management method
SIB	–	*Standards Information Base*, Continuum information standard
SLA	–	*Service Level Agreement*, type of contract for the provision of services supplied to an organisation
SOA	–	*Service Oriented Architecture*
SOX	–	*Sarbanes-Oxley Act*, name of a legal act regulating the obligation of audit services quality control
SRM	–	*Supplier Relationship Management*
TCO	–	*Total Cost Ownership*, cost evaluation method
TOGAF	–	*The Open Group Architecture Framework*, IT standard
TRM	–	*Technical Reference Model*, technical Continuum standard
VSTS	–	*Microsoft Visual Studio Team System*, venture management environment
Websphere	–	software framework developed by IBM

1

Introduction

The development of *information technology* (IT) provides organisations and companies with a chance to adjust the solutions applied to the business processes being analysed and the level of information technology. It is expected that the applied solutions will improve the flow of information resources (data, information and knowledge), which will result in the simplification and improvement of business management processes [7, 18, 61]. This must be preceded by an analysis of the IT technologies already existing in the enterprise, an evaluation of the usability of IT solutions present on the market, and an evaluation of the possibility of the replacement or integration of technology. A preliminary analysis of the height of information technology is especially applied in large organisations in which capturing the relationship between their current state and their objectives and technologies is essential for their proper development [9, 11, 41] and allows for the implementation of IT to support the management of IT resources. To achieve this, a comprehensive view of the organisation is required, as well as the implementation of a specific *Enterprise Architecture* [61] and its realisation in the currently recommended form of SOA (*Service Oriented Architecture*).

The current state of knowledge about the implementation of enterprise architectures allows the possibility of applying relevant (to the scale of the project) models and methods of project management, manufacturing IT systems and evaluating the organisation and its processes. Methods and models should comprehensively support the management of both manufacturing processes (organisation) and processes for obtaining client requirements [53, 54]. The existing methods are usually only adequate for a small group of ventures.

The creation of comprehensive models is possible due to the use of frameworks for developing enterprise architecture and manufacturing IT systems described in the following standards: TOGAF (*The Open Group Architecture Framework*), ITIL (*Information Technology Infrastructure Library*) and COBIT (*Control Objectives for Information and Related Technology*) [1, 2, 40, 63]. *Architecture frameworks* constitute a set of templates of selected system architectures, and

Z. Kowalczuk and C. Orłowski, *Advanced Modeling of Management Processes in Information Technology*, Studies in Computational Intelligence 518,
DOI: 10.1007/978-3-642-40877-9_1, © Springer-Verlag Berlin Heidelberg 2014

methods and tools for their development and implementation in the form of blocks and their mutual relationships. They also include a list of recommended standards and common products applied by *The Open Group* organisation in the development processes of these architectures. The above-mentioned standards are thus the foundation of this book.

In this work, the term 'enterprise architecture' is shorthand for the concept of ‘the architecture of IT systems’ developed in accordance with the ADM.

The preparation of the above-mentioned desired description of a *company*, using the frameworks recommended by TOGAF, is performed in two distinctly different ways:

- the construction of IT infrastructure at the central level of the company – assuming that its resources are of little importance for the analysed requirements, and
- the expansion of IT infrastructure (usually distributed/dispersed) at a local level - by defining general principles for the future shape of EA.

The experience gained from the implementation of several IT projects [19, 31, 51] shows that it is safer (though not always possible) to use the first approach to a project, in which a new project infrastructure is created taking into account the vision and strategic objectives of the company and applying an appropriate business architecture. Then, the resources (data and applications) of business processes are defined and recorded. The data architecture is also defined by specifying its type, structure and the data collection method. Another important aspect of this approach are suggestions on how to make the transition from the current level of business management to the intended target level.

The second approach depends on the maturity of the client and provider organisations (it is important that the decision-makers are convinced of the effectiveness of this approach). According to the ITIL standard, or the CMMI model (*Capability Maturity Model Integration*), the procedure may involve an initial assessment of the organisation’s maturity, and only later (if at least an average "repeatable" level of maturity is established) the use of an approach suitable to the maturity of the organisation.

Thus, the application of an appropriate approach depends on both the maturity of the organisation and the degree of involvement of the partners who want to realise such a project.

In both cases, a team is appointed for the realisation of the project led by a chief architect – CAO (*Chief Architecture Officer*). It could be the CEO or one of the board members. Beneficial effects for the company are achieved only if the CAO is able to represent the interests of all the parties involved in the venture and, at the same time, is able to delegate powers to the local (domain) level of company management. For larger organisations, the chief architect (CAO) may also represent a number of architects (both at a central level and a domain level). This should be a person with a vision of the project and its processes, not just of technology - because usually a new perspective on the implementation of enterprise architecture is needed (not another failed IT project). Apart from defining the people, a common framework for the project (its schedule and budget), as well as the stages and levels of its detailed implementation must be specified. Under these conditions, overall execution is possible, while minor specific objectives do not blur the vision of the project. It should be noted that the development of enterprise architecture can not be treated in terms of *outsourcing*, as an external provider is not able to describe the business processes consistently (according to EA).

The TOGAF standard describes the enterprise architecture development processes, its management processes, and the compliance (*Governance*) management processes of the architecture development processes with the ADM. The latter are reflected in the specific subprocesses of *EA Governance* and *SOA Governance*. It is worth mentioning here that the commonly used concept of *IT Governance* means *IT management processes* in a broad sense as they include both processes of *EA Governance* and *SOA Governance*.

It is assumed, therefore, that the manner of developing EA can be planned and controlled with the use of so-called EA Governance (*Enterprise Architecture Governance*), *i.e.* procedures to provide corporate governance to an enterprise. These procedures include the development and implementation of architectures according to ADM processes (*Architecture Development Method*) [65] by a team led by a chief architect. The wide range of available methods and management functions requires the chief architect to possess relevant management knowledge and skills in order to use an appropriately organised development repository.

The lack of such knowledge or its limited scope, or the chief architect's focus only on (EA) implementation aspects can contribute to the failure of the project. Although implementation poses no particular difficulties, the application of Governance procedures can be complex (and would require detailed discussion).

An analysis of the company based on the TOGAF standard can undermine the purpose (optimality in the TOGAF sense [62]) of implementing enterprise architecture (EA/SOA), *i.e.* as a result of an analysis carried out by the client or provider, the decision on the development of IT systems is not recommended (pessimistic conclusion).

Specialists of the Gartner consulting firm [61] define *SOA Governance* as a process whose aim is "to ensure that the assets and artifacts achieved as a result of implementing SOA (within the adopted architecture) function as expected and meet the accepted quality criteria", *i.e.* the verification of the compliance of enterprise architecture development with SOA. ISO 38500 can be a base when using SOA Governance as it describes (on the basis of six principles of enterprise governance) the planning and control of the use of information technology.

In the case of companies with a single provider of IT infrastructure, SOA is the problem of the provider, who must then (as with enterprise architecture) comply with the principles of SOA Governance, for example through the use of appropriate management principles and procedures, and rules of determining the success factors for the project.

The analysis of *IT Governance* standards, including EA and SOA, in terms of systems and management, raises the question of the direct or indirect influence of Governance processes on the successful implementation of EA and SOA. In a situation where a company decides to implement enterprise architecture according to Enterprise Architecture Governance, should the implementation of SOA and the application of SOA Governance not be considered as an important complement of enterprise architecture (EA) and its management processes (EA Governance)? Must such a complex project as the implementation of enterprise architecture be carried out with the use of heavy management methods? Should the type of project management solutions adequate to the maturity level of the organisation not be chosen? It seems that specialists implementing IT systems have already become aware of the limitations of heavy methods and are now looking for light methods which adapt to the conditions of implementation. Perhaps it is time to attempt a description of Enterprise Architecture Governance procedures based on simple rules which allow the selection of appropriate solutions.

The implementation of EA does not pose any particular problems, especially with the use of SOA. Unfortunately, the use of SOA Governance procedures can be complex and goes beyond the planned scope of this book.

Thus the scope of this work is narrowed down to the management relationship between the enterprise architecture development processes in a company and the processes described in the TOGAF standard (especially the ADM), and the area of *IT Governance* is limited to *EA Governance* (excluding other elements, such as the development of SOA and SOA Governance processes). Particular attention is given to *Governance,* and the compliance of processes (and ultimately that of architecture) to ensure the future enterprise governance of a company.

The adopted narrow subject matter allows for easier and more precise treatment of the initial processing procedures of data in the modelling used in this study.

At this point it is worth quoting the apt statement on inconsistency by LA Zadeh, who said: "If the complexity of the system increases, our ability to formulate accurate, and at the same time significant statements about its behaviour, decreases, until it reaches a threshold value beyond which precision and significance become almost mutually exclusive qualities" [59]. How precise should the description of the development framework of IT systems architectures be to be adequate for the given complexity of the processes analysed? Is the precision of the framework description proposed in the TOGAF, COBIT and ILTIL standards useful for those who develop architectures and IT systems for companies? Should the needs of project managers not be taken into account to a greater extent?

The purpose of this book is to propose the construction of, and to present, such a rule-based model which, on the one hand, attempts to answer the above questions and, on the other, provides support for managers of IT projects in the choice of IT technologies. The book also mentions research conducted at the Gdansk University of Technology on the choice of methods and tools for the management of information technology projects.

The contents of this book include primarily the *Model of Information Technology Management* (*MITM*), based on contextual submodels of project management: *Initial Processing* (*IPP*) *processes*, the maturity capsule (*C-S-P*) and the selection of technologies (*MSM*), as well as their specifications, which contain a description of the *key elements*:

– project organisation maturity (evaluated according to negentropy; Appendices 1 and 4)
– client organisation maturity (Appendix 2)
– provider organisation maturity (Appendix 3).

Since the above-mentioned client and provider organisations, as well as the organisation of the project will be further analysed jointly in terms of *maturity*, for simplicity, we are going to use the concept of the *maturity capsule* of the Client and Supplier (provider) organisations, and the organisation of the Project, in short C-S-P, which will provide a detailed specification of the evaluation of the quality (level) of project management (as a whole) and will constitute the basis for the models of IT project management.

In order to simplify the discussion we will mainly deal with linear discrete models, specifically speaking, discrete-time and dynamic models with the variable of state ('memory'). Nonlinearity evident from a practical point of view will be represented primarily with the use of the characteristics of saturation and quantisation (to certain levels). The main mechanism for non-linear processing will consist of a linguistic model and a fuzzy rule-based model, with the fundamental participation of linguistic evaluations obtained through surveys (test questions).

The model of project and information technology management, MITM, proposed in Chapters 2 and 3, will undergo a practical verification in Chapter 4.

According to the applied *top-down* approach, first the model is presented, then the specification processes of its variables are presented, followed by the verification of the model.

The **second chapter** presents the basic concept of this work, which includes the construction of the MITM model (*Model of Information Technology Management*), the specification of its variables and the verification environment. It also discusses the methodology connecting the multi-level sequential model (*MSM*) of IT selection with a description of the IPP processes. The MSM model includes the (practically verified) sequential/ordered selection of IT technologies – first the IT project management methods and then the *ITM* management tools. The initial data processing procedures (IPP) are also discussed as they specify (set appropriate parameters for) the results of modelling management processes. The IPP processes described formally (linguistically) create favourable conditions for the flow of data (obtained from currently running IT projects) to the MITM model and – through the maturity capsule – to the component management (decision-making) MSM submodel for IT selection. The presented results form the basis for further specification (the contents of Chapter 3).

The **third chapter** presents the specification of the variables of the MITM model and its IPP submodels (of initial processing), as well as of the maturity capsule and the MSM model (for selection of information technology), using vector-matrix and linguistic descriptions.

Particular attention is given to defining the concept of *project negentropy* and discussing the example of the project *Continuum*, which constitutes a *repository* containing project artifacts, which are in the form of concepts of architectures and tested components (*i.e.* implementation solutions for enterprise architectures), blocks (structural elements) of architectures and components, practices and development principles, guidelines and other artifacts.

In the context of the construction of the Continuum, we will discuss the proper project cycle and supporting development processes (*e.g.* architecture concepts), whereas in terms of using the Continuum resources for development, we will mention the proper development processes. Such an overall relationship – including the positive and negative financial consequences – is shown in Figure 1.1, where the development of the IT systems architecture for the client (within the proper project) is carried out simultaneously with mapping the developed architectures onto the Continuum resources (as part of supporting processes) for later use.

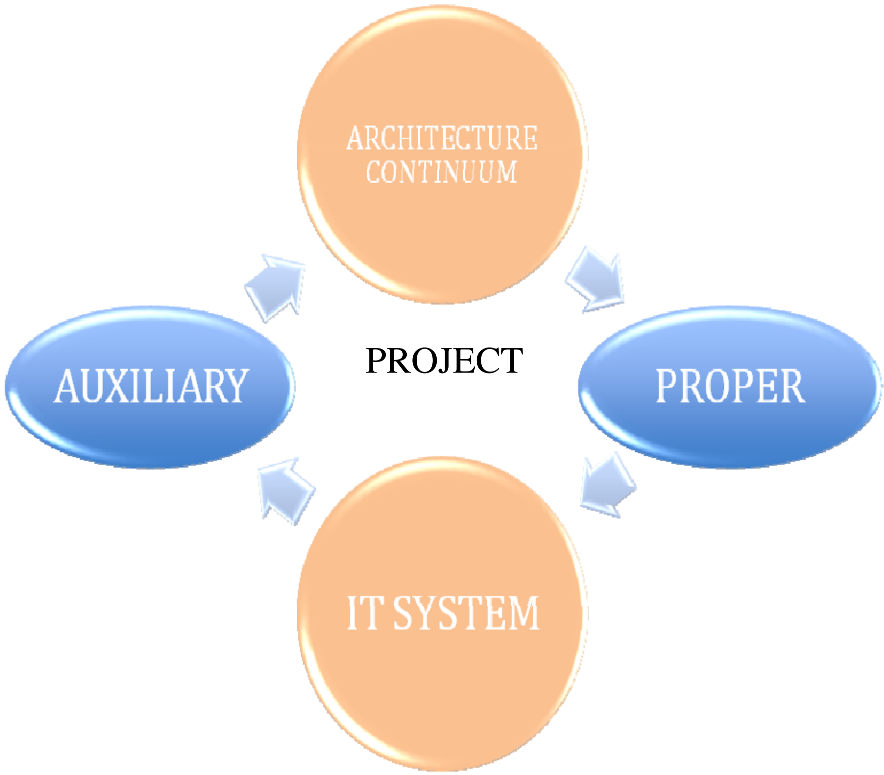


Fig. 1.1 Development being the relationship between the project Continuum and IT systems

Thus, continuous recording and archiving (in the supporting process) of positively verified knowledge of already completed projects (proper projects) in the Continuum resources, ensures a good environment for developing new systems in which provider teams can benefit from this knowledge and use it effectively as a substantive and management support while implementing new architecture development processes.

The integrated approach of the C-S-P *maturity capsule* to the evaluation of maturity, applied in this book, changes the current philosophy of treating independently the concepts of the maturity of the client and provider organisations, as well as the not easily definable notion of project complexity – for the integrated concept of the C-S-P *maturity capsule*: referring to the Client, the provider (Supplier) and the Project. This integration expresses the need for the simultaneous study of the maturity of these three entities while evaluating the level of project management, and it plays a fundamental role in the MITM model.

When assessing the maturity of the client and the provider, it is worth using already recognised IT standards (ITIL, COBIT) and models (CMMI). We suggest that the issue of project evaluation can be solved by introducing the concept of the *structural negentropy* of a project together with appropriate measures. *Vector secondary negentropy,* which is a basic concept in this work, is based on two components: the height of the applied information technologies and the level of expertise, such as the knowledge and understanding of the project.

To properly interpret the two elements, in Section 3.2 of Chapter 3 we will undertake the mapping of *primary negentropy,* described in 3D, onto the above-mentioned vector negentropy, presented in 2D – through the appropriate treatment of the IPP. Such an approach (and the transition from 3D to 2D) facilitates the measurement (determination) of negentropy in both the 2D vector (secondary) version and in the convenient *scalar* (tertiary) version.

This brings us to the possibility of formulating a complete project description contained in the integrated C-S-P *maturity capsule* which allows for the effective and comprehensive quantification of project management. While endeavouring to introduce a common measure of maturity, we are obviously aware of the fact that our initiative requires the approval of the IT environment. What, in particular, relates to the ideas of project negentropy and the maturity capsule proposed in this work. Whenever we mention negentropy, without clarifying the term, we mean project negentropy treated as a tool concept for evaluating the project.

The introduction of project negentropy is particularly important in terms of Governance processes. Until now it was assumed that these processes relate to tracking the dynamics of changes in the enterprise architecture development process and its compliance with the ADM. Following the introduction of negentropy, the concept of Governance can also be identified with the compliance of the architecture development process with the current project negentropy and the state of the maturity capsule. In this way, Governance processes gain a more practical meaning and significance in the realisation of successful projects.

Specified measures are connected to the introduced concept of negentropy. When examining the compliance of the project with the Continuum, there is the possibility to refer to the C-S-P maturity capsule which it contains, as well as to the appropriate negentropy measures, updated in the repository in the course of the project.

The concept of measures involves the concept of *sequencing* (the need to preserve the order of) management and architecture development processes. The proper evaluation of these processes is impossible if their order is not maintained. Similarly, the proper evaluation of architectures cannot occur unless the proper sequence of their development is preserved. Therefore, this book speaks of the *development sequence*, referring to architecture development, including the development of, in turn, architecture vision, enterprise architecture, then IT systems and technologies, and of the *management sequence*, including manufacturing and management processes as well as manufacturing technologies, good management practices supporting the management processes (the use of Continuum resources and own resources) and management technologies supporting management processes and good practices (based on Continuum resources and other external applications).

The introduction of the negentropy and capsule concepts also allows for a change in the philosophy of technology and project management from the separate to integrated treatment of clients, providers and the project, under the common notion of the C-S-P maturity capsule.

Chapter 3 also presents the specifications of client and provider maturity. With regard to the maturity of the client organisation, the organisation's domains are given which correspond to the linguistic variable of 'organisation maturity'. The value of this variable is assumed in accordance with the knowledge of the degree of execution of the manufacturing and management processes. This level is assessed on the basis of questionnaires enquiring about the realisation of control objectives. The same applies to the evaluation of the provider organisation, by isolating the variables corresponding to the level of services offerred by the provider. There are seven variables which describe this level. In the initial IPP processes, a linguistic evaluation of control processes is implemented in the form of competence questions.

Chapter four is devoted to the verification of the developed system of selecting methods and tools for IT project management. Four levels (layers) of verification are suggested.

The first level refers to the verification of the MSM model. It presents a sample implementation environment – the RTC tool (IBM Rational Team Concert) for IT project management. A selection and implementation of plug-ins is carried out for the RTC environment. An IT tool is selected (with the use of a developed plug-in) for project management, and then this solution is evaluated.

The second level covers the topic of supporting Governance processes. It shows how the MITM model is used in a sample company, where enterprise architecture was developed. The need for a comprehensive look at the Governance processes is emphasised (not only from the point of view of TOGAF and SOA). TOGAF/SOA should thus be treated from a common perspective, including a unified linguistic vision and a common approach to managing the implementation of enterprise architecture and SOA in an enterprise.

The third level of verification refers to the use of the MITM model in the maturity evolution process of the provider organisation. For the purpose of the linguistic evaluation of this evolution, the IPP model and the C-S-P maturity capsule are used.

At the fourth level, a verification in the client organisation is performed, assessing its maturity with the use of the IPP model and the C-S-P maturity capsule. The verification elements in this work come from the Department of Information Technology Management at the Gdansk University of Technology [1, 2, 4, 5, 8, 19, 34, 46].

Three main appendices contain the material used in this study to support the modelling, specification and verification of the modelled variables.

Appendix 1 presents the extensive characterisation of standards, which is connected with the need to complete the project negentropy specification presented in the main chapters of this work, the initial processing procedures IPP, as well as the verification processes. Appendix 1 presents mainly the description of the TOGAF standard version 8.1.1. Since the new version is already available [62, 64], the use of the MITM model (and the MSM) requires the mapping of TOGAF 8.1.1 structures onto TOGAF 9 structures. It is assumed that the mapping processes should refer to the following parts of Version 9:

1. Part I: *Introduction to TOGAF 9*:
   This introduction reflects the contents of TOGAF 9 and changes the contents of TOGAF 8.1.1.

2. Part II: *Architecture development method*:
   In the presented description the essence of the TOGAF ADM 8.1.1 was maintained.

3. Part III: *Project Continuum*:
   When compared to TOGAF version 8.1.1, in the TOGAF 9 version substantial changes were made in the description of the project Continuum. In TOGAF 9, a new approach to architecture development was incorporated, as well as a new division of the project repository. The Standards Information Base was removed from the TOGAF 9 specifications. It remains, however, and is still available on the Web [76]. It was also assumed that due to the size and rate of the change of standards in architecture development, it is not possible to maintain and collect such descriptions in the specifications of TOGAF 9 (and subsequent ones), thus the Standards Information Base was left out of this standard.

4. Part IV: *Resource Base*:
   These resources are not included in the TOGAF 9 version. Some elements of the Resource Base will continue to be available. Other

elements of the Resource Base have been moved to other areas of the TOGAF standard specifications.

Appendix 2 contains supplementary material which supports the specification of the 'client organisation maturity' variable of the MITM model presented in this book. There is also data relating to the execution of the control objectives of the studied organisation, which were obtained on the basis of linguistic evaluation questionnaires in the course of examining client organisation maturity.

Appendix 3 consists of material which can be used in the specification of the 'provider organisation maturity' variable. The present description, as was the case in the two previous appendices, uses an approach to briefly characterise data collected on the basis of the ITIL standard to evaluate the provider organisation, and then presents a way to use this compendium of knowledge in evaluating the provider organisation (which may be, for example, the organisation of IT support for a bank).

This book often mentions the notion of *requirements*, both in the context of the discussed IT standards, and in terms of the presented information technology management model.

Therefore, it is worth mentioning the concept of 'requirements' commonly used in software engineering, which refers to the specific characteristics of functions of the developed IT system defined by the client. These characteristics affect both the function itself (what is the system to do?) and its structure (i.e. how these features are to be implemented). This implies a distinction between:

1. functional requirements
2. non-functional requirements.

The first type of requirements are essential in determining the way the system functions. The provider, together with the client, determine how the system is supposed to work (what its structure will be), and how the system is to implement the functions of the company. Sometimes the process of obtaining good requirements [3, 56] is supported by information technology to streamline, organise and document what the client expects.

The determination of non-functional requirements by the provider and the client – those relating to the performance of the system and the conditions of its operation and its reliability - is of great importance in the process of implementing an IT system. The performance of a system, the conditions of its operation and its reliability, as well as ensuring its security, depend on how the two partners, the client and the provider, will determine these requirements.

Development quality is a fundamental concept in the discussion within this book. In general, it is understood here in the ordinary *procedural* sense, requiring the preservation of specific procedures, and in the *structural* sense, which means that the qualitative acceptance of a given artifact (product, architecture, document) is based on the fulfilment of certain structural requirements. An important example, in terms of effective architecture and IT systems development, is the quality of individual *documents* and their *catalogues,* verified in view of the consistency/preservation of their structure. A unique generic concept for this book is *project negentropy*, which constitutes a project's evaluation in terms of structure and parameters. This work contains also various *quantitative quality measures* of manufactured artifacts (products, documents) in the form of their parameters or quantity, or the degree or linguistic level of their realisation or development.

It should be noted that in this book the term *architecture* will be identified with *enterprise architecture*, and the term *project* will be treated and understood as a set of processes arranged according to a schedule and budget existing for its realisation. And the term *company* will be seen as a set of non-material and material assets for specific economic tasks [55].

The information technology management model, MITM, presented in this work, can be implemented and can support the planning of processes to achieve the proper maturity of the client and provider organisations, also with the use of an agent-based system, which is being developed [57-59], and whose task will be to evaluate the environments of IT project realisation and to select the appropriate management method (suitable to project negentropy) and the IT tools to support that method.

2

Model of Information Technology Management – MITM

The starting point in developing a new approach for the selection of information technology is the principle of incompatibility by L.A. Zadeh [59], mentioned in Chapter 1, which refers to the possibility of modelling complex systems. Following this principle, the models presented in this work have been developed maintaining the necessary precision, while retaining their intended functionality. The applicability of the proposed models is associated with a uniform convention for describing models and standards taken from systems theory [13-16]. Ensuring the adequate precision of modelling means the ability to relate reliably to the quality of the manufacturing and management processes which occur in the implementation of IT projects. The functionality of the model was successfully verified [10, 30-32] in terms of the quality of the IT project's description and the effectiveness of its management, as well as the appropriateness of IT standards for the specification of variables.

This Chapter is divided into four parts. In the first part (this introduction) the essential statements and terms of this book are presented. To organise the discussion, these elements are deliberately isolated and presented in the introduction to this Chapter in order to distinguish the terms used in the construction of the model from the formal terms of the model itself. The second part discusses the MITM model as a description of the IT selection process with an ordered/sequential multi-level structure. The importance of IPP processes is revealed here, which rely on the three maturity variables of the MITM model, describing the client, the provider and the project in the maturity capsule. The third part of this Chapter is devoted to these processes.

In the context of the IPP we present the manufacturing/development and project management processes, as well as the management and manufacturing processes in the organisation of the client (*client processes, business processes*) and the provider (*provider processes, manufacturing processes*). The fourth part presents general and linguistic aspects of such models.

Z. Kowalczuk and C. Orłowski, *Advanced Modeling of Management Processes in Information Technology*, Studies in Computational Intelligence 518,
DOI: 10.1007/978-3-642-40877-9_2, © Springer-Verlag Berlin Heidelberg 2014

The applied approach allows a clear view of the detailed specification of descriptive variables of the developed models.

2.1 MITM Contextual Models: Negentropy and the Maturity Capsule

Here are the essential terms and statements and the reasoning behind them referring to the management of information technology and information technology projects. These concepts constitute the basis for modelling.

IT Standards *are sets of good practices and frameworks for development and implementation relating to architecture* (*TOGAF*), *IT systems* (*COBIT*) *or IT services* (*ITIL*).

Examples of the IT standards used above (TOGAF, COBIT, ITIL) are the primary object of consideration in this work.

The term: Enterprise, *according to* [55], *refers to a company as a whole or to one or more of its divisions. It can also mean a parent-corporate structure or a network of cooperating companies.*

It is evident that this term is a different structural version of the functional definition [55] quoted in the introduction. In the above-mentioned, narrower concept of an enterprise (narrowed down to branch offices, several information silos, or key business functions) there is a risk of having no overall vision, which can destroy the concept of enterprise architecture development. Such a partial approach can, however, serve to make the operation of the company more flexible at both the operational and strategic management levels [55].

The proper understanding of the term enterprise also depends on the enterprise architecture development concept assumed by the organisation of the provider.

The term: Enterprise Architecture – *a formal description of the company* [59] *consisting of three parts:*

(i) *framework part: the principles and framework of architecture development*

(ii) *architecture-component part: the vision of architecture, enterprise architecture and architectures of information systems and technology, as well as*

(iii) *methodological part: i.e. methods of implementing these architectures.*

The graphical representation of enterprise architecture is shown in Figure 8.1 in Appendix 3 (Chapter 8), which lists part (i), as well as the individual elements (A, B, C, D) of part (ii) and the elements (E, F, G, H) of part (iii).

The term: Business Image of Enterprise Architecture – *a company's enterprise architecture components, which directly characterise the business properties of the company as seen through IT, organised according to two foundations*:

(i) *business foundation: the vision of enterprise architecture and business architecture*

(ii) *IT foundation*: *the framework part, methodology, and architectures of IT systems and technologies.*

Due to the above dichotomous division, we can refer to the business or IT foundations (*or images*).

The lack of the business image leads to an incorrect evaluation of enterprise resources (*e.g.* it is not known why the application data was created) and of the relationship between business and information technology (the influence of IT support on the functioning of the company is undetermined, which may cause data or application redundancy). Thus the role of image is to identify all the resources of enterprise architecture and to control the relationships between them. In such a case, the provider cannot create enterprise architecture alone, instead a mixed group should be appointed within the company (consisting of employees from different departments of the company) that will see EA in terms of business (image). A well-organised team [31] is essential for the success of such an undertaking. Practices carried out in mature provider organisations acknowledge the role and place of C-S-P teams led by a manager of the architecture group (*Chief Architect*), as the person responsible for the implementation of enterprise architecture in the company.

The foundations of IT and business, their relationships and the impact of IT Governance on the formation of these relationships build up a new coherent picture of the company. This fragmentary approach to IT and a company changes the picture of IT and brings it to the category of Enterprise Architecture, not just technology.

The term: Architecture Principles – *a set of procedures to be followed in the development of enterprise architecture, which ensure its compliance with the ADM, in order to realise the company vision.*

Apart from the framework defined in the introduction, the remaining notions mentioned in the term 'enterprise architecture' will be specified below. It must be stressed that enterprise architecture constitutes the basis for modelling the architecture of a company.

The term: Software/IT Project (**IT_proj**$_i$) *is an ordered set of manufacturing and management processes* (*tasks*), *hereinafter referred to as the schedule, and the running costs of their implementation, hereinafter referred to as the budget. Such a project can be expressed as a* ***two-dimensional vector*** *(structure composed of two parameters*):

$$\mathbf{IT\_proj}_t = \begin{bmatrix} psch_t \\ bt_t \end{bmatrix} \tag{2.1}$$

$psch_t$ – variable of the extent of realisation of the project schedule, $psch_t \in <0, 5>$

bt_t – variable describing the degree of realisation of the project budget, $bt_t \in <0, 5>$

t – current duration of the project, $t \in <0, t_{end}>$

t_{end} – project end (completion) time.

In the above formula <0, 5> means the real-number scope of realisation. In this book, the description and evaluation of projects and their models will be based on linguistic (lexical) measurements in the sense of Mamdani rules [57]. In many cases, the variables (such as the above stages of implementation) will be evaluated on a rating scale of 0 to 5. At the same time, it is accepted that the lowest grade equals 1.

The term: Scalar evaluation of a project, *representing the overall estimation of the degree of project realisation, is expressed by a variable whose value is determined by a function of project parameters* $rd_t = f(\mathbf{IT\_proj}_t) = f(psch_t, bt_t)$:

rd_t – variable describing the degree of project realisation, $rd_t \in <0,5>$.

When realising an IT project, the project manager needs evaluating measurements of the state of the project. A priori evaluation measurements of an IT project [57] are often used, such as COCOMO II, function points, and the degree of complexity. There is, however, a lack of dynamic measurements describing the current state of the project. Such measurements are necessary in complex projects requiring highly dynamic changes, which are difficult to control. Therefore, in the management of such projects, for the estimation of the global maturity (arrangement) of a project, a measurement called 'project negentropy' is recommended.

The term: Vector (secondary) project negentropy *is a measurement of the global maturity (arrangement, perfection) of an IT project, which will be determined on the basis of the height of the information technology applied and the knowledge and understanding of the project by the provider and the client:*

$$\mathbf{p}_t = \begin{bmatrix} hT_t \\ he_t \end{bmatrix} \tag{2.2}$$

$\mathbf{p}_t$ – two-dimensional evaluation of IT project negentropy

p_t – scalar evaluation of IT project negentropy, $p_t \in <0, 5>$

hT_t – variable for evaluating the height of the applied management technology of the IT project, $hT_t \in$ <0, 5> (described below: $hT_t = pr_t$, or even $hT_t = rd_t$)

he_t – variable of the level of expertise, knowledge and understanding of the project area ($he_t = ar_t$), $he_t \in$ <0, 5>.

The cybernetic concept of negentropy (structure) is the contradiction of its physical prototype (the measure of disorder, or dispersion); moreover – it is a vector quantity.

The use of the concept of negentropy simplifies the management of projects – with a current evaluation of the project, the appropriate management decisions can be taken quickly, and thus the chances of completing the project increase in terms of equation (2.1).

The multi-dimensional definition of negentropy allows for a spatial (richer than scalar) representation of the analysed concept (negentropy and variable trajectory), which, from the point of view of project management, essentially extends the possibility of analysis.

The exclusive use of the scalar description p_t would limit the ability to study project negentropy – as a closed, scalar representation of two variables: the height of applied IT and the level of expertise.

In many situations, as is clear from experience, the scalar evaluation $p_t = Q(\mathbf{p}_t) \in$ <0, 5> can also be extremely useful. In addition to an entirely expert evaluation, one of the mathematical methods described in Appendix 4 can be used to scalarise vector concepts. In a practical approach, the synthesis of negentropy can be carried out on the basis of a linguistic evaluation of its components. In a similar way, other vector values defined in this study can be estimated in a scalar way, vector coordinates (MITM variables) can be aggregated, and thus a clear evaluation of the analysed values can be obtained. A hierarchical approach to the multi-dimensional scalarisation of negentropy, based on eight *source* components, is presented in Section 3.3.

The term: The height of the applied IT project management technology *is evaluated on the basis of the quality of the applied methods of IT management and the level of the applied IT tools:*

$$\mathbf{hT}_t = \begin{bmatrix} hmt_t \\ hmm_t \end{bmatrix} \tag{2.3}$$

$\mathbf{hT}_t$ – two-dimensional estimation/evaluation of the height of advanced IT project management technology

hT_t – scalar evaluation of the height of advanced IT project management technology, ($hT_t = pr_t$), $hT_t \in$ <0, 5>

hmt_t – variable of the level of applied IT tools of IT project and system development, $hmt_t \in$ <0, 5>

hmm_t – variable of the quality of the applied IT project management methods (evaluated linguistically, by experts), $hmm_t \in <0, 5>$.

The term: Maturity level of an organisation *is evaluated on the basis of the degree/level of the maturity of the organisation's key processes.*

The term: Supplier/Provider Organisation Maturity:

$$\mathbf{s}_t = \begin{bmatrix} ka1_t \\ ka2_t \end{bmatrix} \tag{2.4}$$

$\mathbf{s}_t$ – vector variable of provider organisation maturity

s_t – scalar variable of provider organisation maturity (degree/level), $s_t \in <0, 5>$

$ka1_t$ – scalar variable of the maturity of key IT system <u>manufacturing</u> processes, $ka1_t \in <0, 5>$

$ka2_t$ – scalar variable of the maturity of key IT system <u>manufacturing management</u> processes, $ka2_t \in <0, 5>$.

The term: Client Organisation Maturity:

$$\mathbf{c}_t = \begin{bmatrix} hcm_t \\ hcc_t \end{bmatrix} \tag{2.5}$$

$\mathbf{c}_t$ – vector variable of client maturity

c_t – scalar evaluation of client maturity (degree/level), $c_t \in <0, 5>$

hcm_t – variable of the level of personality <u>matching</u> of the client to the team, $hcm_t \in <0, 5>$

hcc_t – variable of client <u>competence</u> or suitability to the project, $hcc_t \in <0, 5>$.

The term: Personality matching of a client *means* [23] *a set of individual qualities useful for the joint realisation of an IT project; the qualities affect the choice of the form of cooperation with the provider and its effectiveness; they are also significant in direct contact between the project partners.*

The term: Client suitability (competence) [23] *applies to a set of competence qualities which describe the client's qualifications, experience and references for the project. The evaluation of the competence/suitability arises from direct contact with the client.*

The term: Maturity Capsule – *a set of maturity evaluations of the organisations of the supplier /provider and the client, as well as the project (estimated by scalar project negentropy) which forms the basis for the MITM model.*

The term: IT Project Management Level *identified on the basis of the maturity capsule (with symbols unified with the target management evaluation* z_t*)*:

$$\mathbf{z}_t = \begin{bmatrix} z_t^c \\ z_t^s \\ z_t^p \end{bmatrix} \tag{2.6}$$

$\mathbf{z}_t$ – multi-dimensional evaluation of the level of IT project management

z_t – scalar evaluation of the level of IT project management, $z_t \in <0, 5>$

z_t^c – sub-level of management resulting from client maturity

($z_t^c = c_t$, $z_t^c \in <0,5>$)

z_t^s – sub-level of management described by provider maturity

($z_t^s = s_t$, $z_t^s \in <0,5>$)

z_t^p – sub-level of management related to negentropy ($z_t^p = p_t$),

$z_t^p \in <0,5>$.

Supposition. It is assumed that the measurement of global maturity or the degree of the arrangement of IT project management processes is negentropy. According to the assumed convention, at the initial stage of the IT project its (manufacturing and management) processes have low negentropy, which grows during its implementation (note that in fact the opposite is true with physical entropy).

Justification. Since, according to (2.2), secondary negentropy $\boldsymbol{p}_t$ contains two components (hT_t – a scalar evaluation of the level of applied information technology, and he_t – a scalar evaluation of the level of knowledge of the area, see Appendix 4), at the initial stage $t = 1$:

$$\textbf{\textit{IF}}\ hT_t\ \textbf{\textit{IS}}\ low\ \textbf{\textit{AND}}\ he_t\ \textbf{\textit{IS}}\ low\ \textbf{\textit{THEN}}\ p_t = 1 \tag{2.7}$$

and for $t = t_{end}$ it is assumed that

$$\textbf{\textit{IF}}\ hT_t\ \textbf{\textit{IS}}\ high\ \textbf{\textit{AND}}\ he_t\ \textbf{\textit{IS}}\ high\ \textbf{\textit{THEN}}\ p_t = 5 \tag{2.8}$$

Following the adoption of the secondary vector negentropy as a measurement of the arrangement/maturity of the project, identifying its components requires an evaluation of the development frameworks for architectures and IT systems, and their applicability in the IT management (ITM) process.

Supposition. The frameworks (procedures) of development of architecture and IT systems constitute a potential source of describing manufacturing processes (IT manufacturing and management processes) during the scalar evaluation of project negentropy.

2.2 Information Technology Standards for Evaluating the Maturity Capsule

Large IT consortia which deal with issues of the manufacturing and implementation of IT systems store knowledge on IT projects. On such a basis, they produce and promote IT standards which incorporate comprehensive knowledge about IT project manufacturing and management processes. Since the scope of development is considerable, and each implemented project is specific, the standards promoted by the consortia are based on schemes chosen from the framework which they use. Therefore, it is necessary to evaluate the applicability of such a framework to the needs of ongoing projects.

Supposition. The ability to apply IT standards (TOGAF, COBIT, ITIL) forms a suitable basis for increasing the scalar value of project negentropy.

Justification. The standards of project manufacturing and management include descriptions of IT projects. For project managers, building a credible schedule $psch_t$ and budget bt_t is an extremely difficult task. According to equation (2.1), the scalar evaluation of the scope of a project rd_t is derived from $psch_t$ and bt_t. Standards are an important aid in the reliable creation of these parameters. Thus, by applying the standards we increase the chances of the successful implementation of a project. They allow close relationships of values:

$$rd_t \approx psch_t \approx bt_t \tag{2.9}$$

On the basis of (2.2) and (2.7) it can be assumed that for t=1 :

$$p_t = 1 \tag{2.10}$$

and for t=t_{end} :

$$p_t = 5 \tag{2.11}$$

Supposition. The clarity and faithfulness of standards in IT project management is not perfect.

Justification. On the basis of the analysis of standards, it can be said that they are very general and have many limitations in application. The literature on the use of TOGAF, COBIT and ITIL standards records just a small number of cases of their application in their entirety. Typically, they are only applied in their fragments, while the standard framework provides support only for company decision-making processes. When using COBIT for the entire organisation, some domains are not evaluated, which usually results from the specific nature of a company (*e.g.* specified domains do not exist). The same is true of the fragmentarily-used ITIL version 3 [17], which by nature is more strategic than version 2 of the standard.

TOGAF:

$$\mathbf{z\_TOGAF}_t = \begin{bmatrix} pd\_ADM_t \\ pm\_ADM_t \\ pd\_Cont_t \\ pm\_Cont_t \\ hmd_t \end{bmatrix} \tag{2.12}$$

z_TOGAF$_t$ – multi-dimensional evaluation of compliance with the TOGAF standard

z_TOGAF_t – scalar level of compliance with the TOGAF standard, $z\_TOGAF_t \in <0, 5>$

pd_ADM_t – variable of the level of ADM development processes, $pd\_ADM_t \in <0, 5>$

pm_ADM_t – variable of the level of the ADM management process, $pm\_ADM_t \in <0, 5>$

pd_Cont_t – variable of the level of Continuum development processes, $pd\_Cont_t \in <0, 5>$

pm_Cont_t – variable of the level of Continuum management processes, $pm\_Cont_t \in <0, 5>$

hmd_t – variable of the level of other (support and proper) processes of management and development, $\mathrm{hmd}_t \in <0, 5>$.

COBIT:

$$\mathbf{z\_COBIT}_t = \begin{bmatrix} ppd1_t \\ ppd2_t \\ ppd3_t \\ ppd4_t \end{bmatrix} \tag{2.13}$$

z_COBIT$_t$ – multi-dimensional evaluation of compliance with the COBIT standard
z_COBIT$_t$ – scalar level of compliance with the COBIT standard, *z_COBIT*$_t$ $\in$ <0, 5>
$ppd1_t$ – variable of the process level in domain 1, $ppd1_t \in$ <0, 5>
$ppd2_t$ – variable of the process level in domain 2, $ppd2_t \in$ <0, 5>
$ppd3_t$ – variable of the process level in domain 3, $ppd3_t \in$ <0, 5>
$ppd4_t$ – variable of the process level in domain 4, $ppd4_t \in$ <0, 5>.

ITIL:

$$\mathbf{z\_ITIL}_t = \begin{bmatrix} pITIL1_t \\ pITIL2_t \\ pITIL3_t \\ pITIL4_t \\ pITIL5_t \end{bmatrix} \tag{2.14}$$

z_ITIL$_t$ – multi-dimensional evaluation of compliance with the ITIL standard
z_ITIL$_t$ – scalar degree of compliance with the ITIL standard, *z_ITIL*$_t$ $\in$ <0, 5>
$pITIL1_t$ – level of the service strategy development process, $pITIL1_t \in$ <0, 5>
$pITIL2_t$ – level of the service development process, $pITIL2_t \in$ <0, 5>
$pITIL3_t$ – level of the service manufacturing process, $pITIL3_t \in$ <0, 5>
$pITIL4_t$ – level of the service maintenance process, $pITIL4_t \in$ <0, 5>
$pITIL5_t$ – level of the constant service improvement process, $pITIL5_t \in$ <0, 5>.

In view of the above, on the basis of (2.5) and (2.12) it is assumed that:

IF *TOGAF _standard_scope* ***IS*** *considerable* ***AND*** *hmt*$_t$ ***IS*** *high* ***THEN*** $p_t = 5$ (2.15)

For the development of IT systems it is necessary to know the standards. They support the managers of provider organisations. They constitute a 'framework', supporting the manufacturing and management processes and models which can be customised for the needs of any provider organisation. This issue will be discussed later in this work.

Supposition. It is possible to apply a linguistic description to the standards of enterprise architecture and IT systems development.

Justification. This can be based on an analysis of standards and on the development of their linguistic description. We will do so in relation to the TOGAF, COBIT and ITIL standards.

Supposition. It is possible to build an IT management model for use in IT project management.

Justification. To prove this, the appropriate relationships must be proposed as well as a model showing the relationships between the variables of this model. Therefore, a process of modelling IT management will be presented later, with relevant evaluations.

The Term: IT Systems/Enterprise Architecture

According to ANSI and IEEE Std 1 471-2000 *the term architecture* [60] *means:*

- *basic structures of a system, its components and their mutual relationship, its reference to the environment, and the principles regulating its development.*

But according to TOGAF the term architecture has two meanings (*depending on the context*)*:*

- *a formal description of a system or its detailed plan on the components level, together with the method of its implementation* or
- *the structure of components, their mutual relationship and principles and guidelines of their development and implementation* (*compliant with ANSI/IEEE Std* 1471-2000).

In analysing the task of developing IT systems architectures, it is vital to refer to IT systems architectures based on an object-oriented approach [12].

The Term: Blocks of Architecture. *Architecture components with public interfaces of access to their functionality* [65]. *They are characterised by the following features:*

- ✓ *they are easy to implement and use*
- ✓ *they are easy to modify*
- ✓ *they use available technology and standards*
- ✓ *they can be created on the basis of other blocks*
- ✓ *they can constitute elements for other blocks*
- ✓ *they can be used again, exchanged and implemented in environments of different mutually dependent blocks.*

The term: Object-Oriented Architecture - *this type of architecture takes into account the initial division of the system into object patterns and objects and defines the relationships between them.*

The method of developing an object-oriented architecture is based on the following steps:

- setting user requirements – use cases
- identifying classes of objects

- setting and defining the attributes of objects
- determining the relationship between the classes of objects, *i.e.* generalisation and specialisation
- defining and identifying object methods
- defining the dynamic part of the system, *i.e.* the interaction between objects.

Object-oriented architectures are the basis of the development of most IT systems [61]. Monolithic client-server architectures are built with their help.

The term: Client-Server Architecture - *within this type of IT systems architecture the application is divided between the resources of the client workstation and the server network.*

In the case of a *thin client* architecture, server resources include the data and business logic of the application. In terms of a *fat client*, the layer of the business application is on the side of the client, which requires the use of high-performance servers.

The term: Layer Architecture - *within this architecture the system is divided into layers the integration of which is dependent on the applied software development environment – software platforms* (*e.g. J2EE/.NET*) [22].

This type of architecture is characterised by tightly coupled elements.

On the logical level in the architecture of the layer of presentation, processing and data management are separated by processes. It is worth noting the meaningfulness of creating specific layers and indicating the means of their separation. The application of such architectures can be based on a variety of supercomputing hardware. At a server level, both utility processing and data management can be performed (*e.g.* as two logically different servers or as two separate processes supported by two independent processors). A multi-tier architecture is an example of an open architecture in which it is possible to add new resources as client needs grow.

Computer systems based on multi-tier architectures are flexible and scalable - depending on the load, several versions of the system can be created with the same set of services offered by separate or duplicate components. Growing needs can be compensated by changing the configuration of the system, such as the migration of components in the system.

In the construction of a new architecture, it is worth considering the functionality of the system and treating it in terms of services. Then a representation of these services by software components must be assumed, reflecting the particular layers of the system.

The term: Software platform *is an integrated development environment for the management of software development processes, application code, memory, and security.*

The term: Development environment *allows for the creation of software code using a programming language.*

The term: J2EE platform – *an IT systems development platform supported by the SUN company, using the Java object-oriented language.*

The term: .NET framework – *an IT systems manufacturing platform supported by Microsoft. In the systems implementation process it is based on both object-oriented languages as well as structural languages. The developers working on the .NET framework develop software using a number of programming languages* (*within the standard*)*, and then apply a common environment called Common Language Runtime.*

Three levels of enterprise architecture development (of applications, data, and technology) model and organise the picture of a company and provide a platform for the evaluation of its resources during the manufacturing and implementation of IT systems. The assumed approach [64, 65] adds a fourth level to enterprise architecture, namely business. This addition seems obvious (especially in terms of light management methods and the added role of a business analyst), but combining these levels into one platform creates a new challenge. This solution allows business goals in IT applications to be taken into account, and properly expresses the principles of manufacturing and developing adequate levels of the enterprise.

The term: Service Oriented Architecture (SOA) *is a service-based architecture which defines services and the methods of communication between them.*

The term: Capability Maturity Model Integration (CMMI) *allows to define the organisation maturity by examining the organisation's manufacturing and management processes.*

The term: Cost Construction Model II (COCOMO II), based on COCOMO, *allows the costs, investments and schedule of an IT project to be effectively defined.*

The creation of the schedule and budget of a project, and the development of the software provider's investment strategies can be based on the COCOMO II model, which forms the basis for constructing a model of technology. It is, therefore, a *technology model* expressed in the form of a structured set of processes, as a 'process model' (such as the JAVA or .NET process models, the Java server process model, or the project management process model).

The technology model described above is conceptually distinct from the models of MITM and MSM (*technology management* and *selection*) which constitute the objective of this work.

2.3 Multi-level Sequential Model of Information Technology Selection – MSM

The set of variables and the multi-level ordered/sequential structure of the MSM model of IT selection is rooted in years of professional practice in information technology project management. Thus the structure includes an ordered selection of methods and IT tools for the management of information technology projects. The project management method is chosen first and is not changed during the project. Consequently, it is also assumed that the choice of the method is followed by a selection of tools to support management processes.

Changes in project management processes are followed by changes in management tools. If the tool is not replaced in its entirety, then its functionalities are selected according to the changes in the processes of project manufacturing. The consequence of such changes are thus new features. In this way, the project manager developes project management methods and functionalities of tools supporting project management.

In this approach, it is assumed that the decision variables, when choosing a method of IT project management, are: the maturity of the client and provider organisations as well as IT project negentropy, contained in the maturity capsule (and describing the level of project management in detail). A sequential model of selecting the methods and tools to support the management processes is shown in Figure 2.1.

The MSM model includes two sub-models: of method selection (MSMm) and of tool selection (MSMt), and it also includes the ability to change tools and their functionalities during the project. The model does not account for changes in management methods during the project. However, the change of tools and the modification of their functions are relatively common.

Both the models and the feedback mechanism showing the impact of changes in the project in the scope of the functionalities of the tools is presented in Figure 2.1. It shows a multi-level sequential MSM model of selecting methods and tools for IT project management.

The MSMm model of selecting management support methods (left side of Figure 2.1) is shown in detail in Figure 2.2, whereas the MSMt model of selecting functionalities and tools to support management (middle part of Figure 2.1) – is shown in detail in Figure 2.3. Both the models (MSMm and MSMt) are multi-level. The model of selecting methods for IT project management, (shown in Figure 2.2) includes four procedure levels of the estimation process (*i.e.* the evaluation

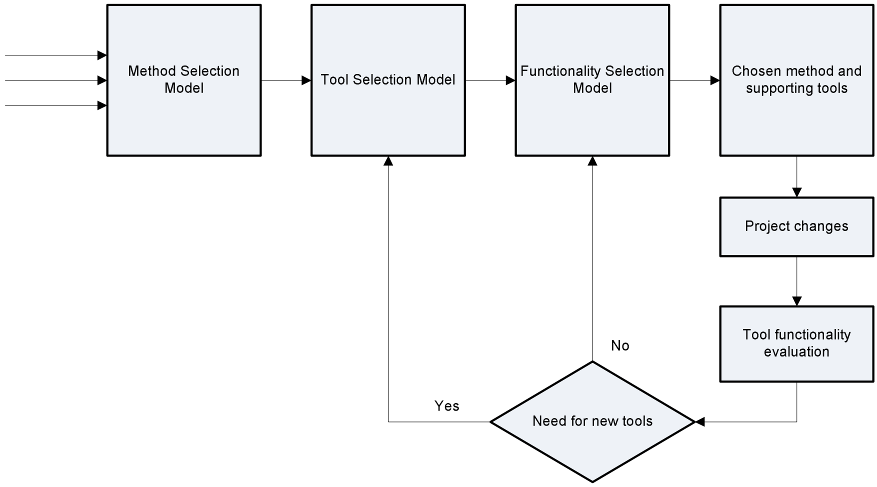


Fig. 2.1 Multi-level ordered/sequential model MSM of selecting methods and tools for IT project management

procedures) of the maturity of the organisations of the client and supplier/provider as well as the project negentropy, contained in the maturity capsule:

- **Level one** – a measuring system, classifying the project, the providers and the clients. The research aspect of provider maturity, project negentropy, and client matching and suitability are taken into account here. Depending on the research results of the organisation's maturity for executing the project, decisions are made regarding changes in the organisation which are necessary for the effective implementation of the project.
- **Level two** – suitability: possible decisions of project managers are analysed here, regarding changes in the organisation relevant to the implementation stage of the project.
- **Level three** – maturity: evaluation of the client is carried out at this stage and any changes to improve project management are suggested.
- **Level four –** the use of a system supporting decision-making in terms of the selection of project management methods. The system consists of independent software components (called *agents*). On the basis of knowledge of the quality of the provider organisation, the client organisation and the project negentropy, it supports the project manager in making his decisions.

After the introduction of the MSMm, the MSMt model of selecting complete tools or their functions (Figure 2.3) will be presented. It includes five procedural levels. In Figures 2.2 and 2.3, the brief description 'data of client, provider and project' means a specific evaluation of the maturity of the client and provider organisations as well as the scalar negentropy of the project.

Level one – the measuring system concerns any manufacturing team. The measurements relate to the degree of execution of the project schedule and budget. In the system presented in Figure 2.3 the measurements relate to the level of an IT project managed with the use of light methods (such as the execution of tasks from a list of requirements called SCRUM, *Scrum product backlog list*) or with the use of heavy methods (as a stage of the project using the PRINCE method, *i.e. Project in Controlled Environments*).

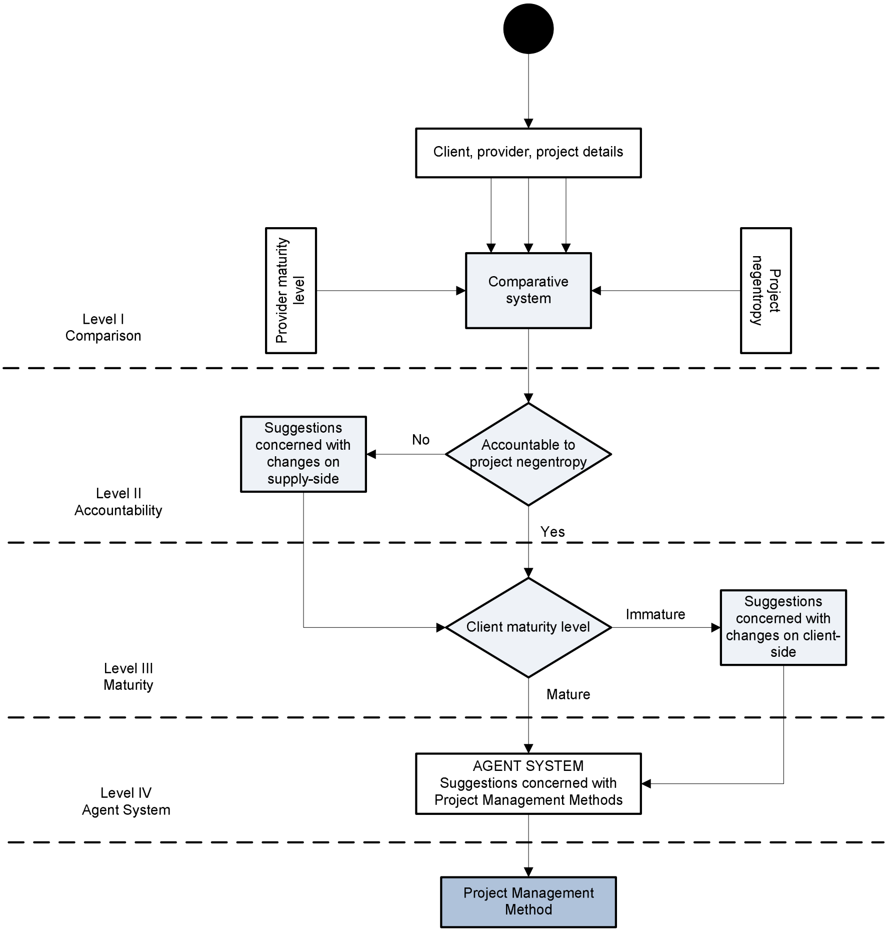


Fig. 2.2 Detailed model of selecting the project *management method* – MSMm

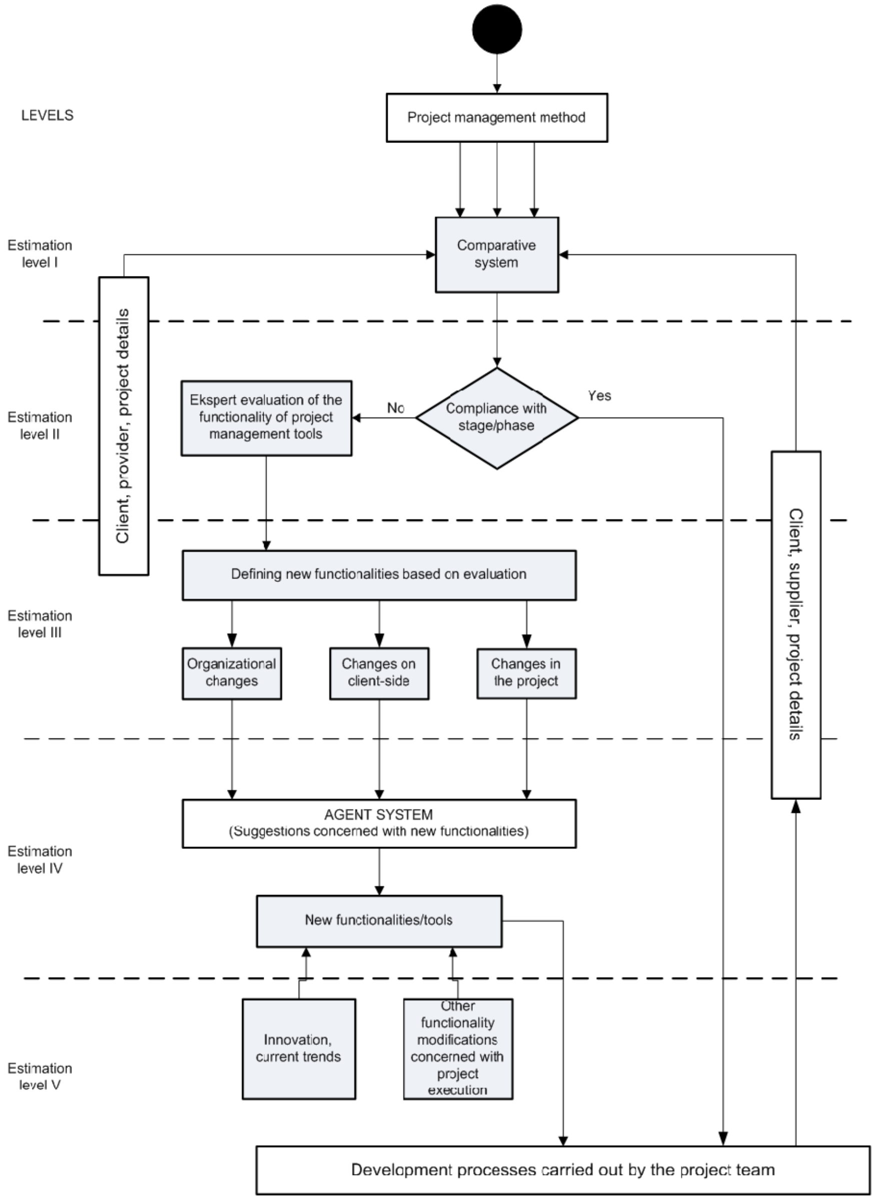


Fig. 2.3 Detailed multi-level model of selecting *IT project management tools* or changes in their functionalities – MSMt

Level two – the processes of tools evaluation used to support project management. Two cases are considered here. In the first one, the use of a project management tool (*e.g.* the OpenSource environment) allows for new services/functions to be added in the form of plug-ins. A plug-in is a module which can be attached to an existing system in order to extend its functionality, and it is implemented through the use of appropriate technology. The second case uses the functionality of a different tool. Both tools can function in cooperation (based on SOA architecture), or be used independently through constructing a data exchange environment. In both cases, an expert evaluation is required as well as identification of the causes of change (in the applied functions) which lie in the problems of project and organisation management (with respect to the organisations of the provider and the client).

Level three – defining new functionalities for the applied tool which supports project management. After an initial expert evaluation of suitability, the defining of new functionalities is associated with changes in the management of the provider/supplier and client organisations. The identification of the causes and consequences of these changes is based on general knowledge about managing projects. The cause of such changes could be, for example, the attitude of the client in terms of matching to the team, or client suitability.

Level four – the prognosis area of the agent-based system [6, 7]. It was assumed that in the examination of information technology for project management, these prognoses relate to new functionalities offered by IT tools. The analysis of various functions is necessary, as well as the implementation and presentation of the functionalities typical for project management methods (according to the sequence from heavy to light methods).

Level five – a proposal to modify the functionalities of IT management tools. These changes result from an analysis of the degree of utilisation of existing project management tools and from the emerging proposals for changes (*e.g.* to new management methods or tools under the influence of fashion) or from the absence of solutions in the agent-based system [48]. The proposals are submitted by the project manager at the team meeting of experts.

2.4 IT Systems Architectures

The description of architectures will begin with the presentation of an enterprise architecture development method based on the ADM. It includes the construction processes of: architecture vision and (on its basis) business architecture, applications, data and technology. Relevant documents as well as manufacturing and management processes have been introduced for these processes [63]. The description of processes and the corresponding documents will be based on the concept of project negentropy introduced in this book.

Architectures – architecture vision and the names of the subsequent architectures are consistent with the nomenclature adopted by *The Open Group*. The design of this vision is based on the resources of the company (mission, vision, strategy, and objectives of the enterprise) and the competences of the employees. A high-level description of the company’s technical and economic environments is created, as seen from the economic and technical perspective, taking its resources into account.

Notations expressing this vision (useful in the IPP) are: business scenario, business requirements documents, as well as specific expressions (from the field of the integration of business and technology objectives – such as modelling entities, or the conversion of processes into function diagrams) directly related to the construction of the architecture vision. In the process of constructing this vision, two groups of processes can be distinguished: *of manufacturing*, concerning the accepted methods of manufacturing and *of management*, related to the manufacturing method applied. In addition to the arranging/ordering effect of this classification, it is also used in the IPP process to support the construction of the MITM model. During the preparation of the architecture vision, the relevant documents are created in terms of manufacturing and management.

The manufacturing documents include:

- architecture development principles along with the identification of key personnel and their system/development requirements
- prepared and validated methods of architecture development.

The management documents include:

- the architecture vision (as described in the TOGAF standard), taking the initial and target business architecture, technology, data and applications into account
- methods of the creation and verification of business scenarios (including operation-specific constraints), including business processes and the corresponding business and technical environment, as well as a list of the project participants and their roles.

Before using the architecture vision in the manufacturing processes of a company, the architecture project must include [64] all the key business requirements and constraints which must be acceptable to the company. In architecture development, the project components contained in the Continuum are used, as well as the priorities of architecture developers in terms of their use of the components (within the proper process). Other architectural frameworks [63] are also taken into account (*e.g.* the Zachman framework [62]). The sequence of the processes of architecture vision development is presented below (for the IPP, which will be discussed later):

- defining the business goals of a company
- defining the architecture scope (within the defined project) *i.e.*:
  - ➢ the scope of the enterprise
  - ➢ the level of detail of the description and the enterprise scope description areas
  - ➢ the schedule for constructing the architecture of business processes
  - ➢ the architecture of resources which can be contained in the project Continuum (which were defined in the introduction):
    - ✓ resources created in the earlier stages of the ADM
    - ✓ other external resources (other frameworks, system models, *etc.*)

- defining the constraints of building the architecture of business processes
- identifying business requirements and the architecture vision
- developing business scenarios, including defining components:
  - the issue to solve and the existing business and technical environments
  - the goals of architecture development and the components of success (indicating processes/resources increasing the chances of project realisation)
  - project participants and their roles
- developing initial architectures under the ADM.

The above-mentioned manufacturing processes, essential in the development of the architecture vision, require appropriate management processes [64], in which IT Governance processes will be implemented to ensure the development of architectures in accordance with the ADM. The use of IT Governance processes assumes strategic management, involving the key persons of the company in developing the architecture vision. This process also verifies the previously adopted principles of architecture development and the key persons from the point of view of their usefulness for the implemented project.

Architectures – Business Architecture. As was the case in the above-discussed architecture vision, in the process of developing a business architecture we can distinguish the following: artifacts/products, documents necessary for their manufacturing and the manufacturing and management processes. Such a division of business architecture (products – documents – processes) is convenient in terms of evaluating the applicability of a business architecture in the IPP modelling. The availability of a particular business architecture of a company (built within other organisational processes, such as business planning, strategic business planning, business process reengineering, *etc.*) is the basis for the subsequent construction of types of instances (data, applications, and technology). In the development of such an architecture, business process reengineering (defined as a process which aims to radically change business processes) may become necessary, as it covers the entire company and is there to adapt its business processes to the requirements of IT systems. The scope of work in developing a business architecture depends largely on the company resources available and the opportunities to use them. These resources include business plans, development strategies and specific business requirements. If they are re-used, they should be reviewed and updated. After taking the above-mentioned documents (discussed in the description of the architecture vision) into account in the development of a business architecture, this development results in two business architectures, initial and target, which include:

- the organisation structure together with the business goals
- business functions, both those which reflect the main scope of the company and those detailed ones
- internal and external services provided by the company to its own clients

- business processes, including the means and methods of their realisation, as well as business roles, their changes and modifications
- a business data model presenting relationships between the company and its functions.

The following documents are prepared before the realisation of management processes:

- a current form referring to the preparation and realisation of business architecture (including technical requirements)
- accepted (in the architecture vision) architecture development principles including:
    - the construction (supporting processes) or application of the Continuum
    - the application of the architecture vision, including initial and target business architecture, data, applications and technology
    - results of a gap analysis.

A question arises: whether such a precise specification of documents will be useful in IPP? Will types of documents (management/manufacturing) be significant or rather their examples?

To develop a business architecture, the following steps must be realised (the sequence in which they are carried out depends on the state of a company and the availability of documents):

- development of the initial business architecture for the adopted scope and level of detail describing the architecture vision, which is dependent on the type of business, company size, and the level of business process descriptions – using data from the Continuum is recommended (as part of the proper process)
- identification of the existing resources: patterns, tools and experience:
    - the choice of business architecture resources (reference models, patterns, *etc.*) from the Continuum – taking the company indicators into account
    - the choice of business architecture views (*e.g.* of realised operations, financial management) and the evaluation of this architecture by the client and the provider
    - an evaluation of the appropriate tools and techniques which are to be used in the modelling process, and an analysis of the views (documents, spreadsheets, functionality models, business process models, model use cases, *etc.*)
- manufacturing the business architecture model:
    - incorporating client feedback through the use of business models or scenarios prepared for different levels of detail and hierarchy
    - the development of information requirements for the developed business functions (identification of roles and sources and

  assigning functions to them) – the level of these requirements depends on the complexity of the developed business architecture
  - controlling model quality, in terms of the degree of implementing principles and objectives and of including the requirements of the company
  - testing the architectures and their completeness against the set requirements
- choice of blocks for the construction of the business architecture model:
  - evaluating the requirements of the blocks and checking if they are available in the block library as well as establishing if they can be re-used (if suitable business blocks are not in the library, the decision to manufacture new ones)
- inspection and verification (done by clients) of the model of architecture taking blocks into account
- quality evaluation of the architecture model (*e.g.* efficiency, costs, size)
- developing the architecture of a complete business model:
  - choosing the manufacturing standards for each block, developing the possibility of their re-use (within the proper process) on the basis of reference models selected from the Continuum
  - when selecting architecture blocks, the creation of a cross-sectional version of the architecture before assessing the business objectives (justifying the need for application)
  - preparation of the final report after analysing the requirements and the possibility of mapping the architecture into the Continuum. The report would also contain the selection and evaluation of blocks which are to be included in the Continuum (for re-use and publishing via the architecture repository)
  - preparation of the business architecture development report
- tracking activity (people and places associated with key business functions)
- detailed descriptions of the business functions (processes) and their information needs
- management paths (to show the control level and the responsibility of partners)
- standards and guidelines showing working practices, legislation, funding, *etc.*
- records of experience resulting from the gap analysis and the corresponding descriptions.

The range of processes is significant. Therefore, the documents describing the manufacturing process can be divided into three groups: preparatory documents, main documents and documents of the transfer of business architecture. The basis for this classification is the applicability/fitness of the documents for the development process.

In the preparation of documents, the bottom-up approach is used, which takes the state of the IT resources (systems and data) of the company into account.

If there is no clear business architecture of a company, an analysis of the deficiencies (gap analysis) is performed to re-identify and evaluate business processes. It is important to examine the potential areas of deficiencies, which include:

- resources: personnel, information, financial resources (*e.g.* omission of key individuals in the selection for training, or buildings or premises in the selection of resources)
- the specification of certain processes or tools
- services and functions which were omitted during development or procurement.

Gap analysis may be a useful management process in the IPP. It may in fact show that some of key processing procedures should be ignored, or taken into account as particularly valuable.

Architectures – IT Systems Architecture. When developing enterprise architecture and having finished the development of a business architecture, the next step is the development of an IT systems architecture, including data and applications architectures. The processes of their development will be presented separately, although from the viewpoint of the IPP, they could be discussed together. The manufacturing and management processes used in the development of an IT architecture, and the manufacturing and management documents, will be discussed below.

In the course of developing an IT systems architecture, two basic documents of data architecture development are prepared:

- the architecture vision including business scenarios (the vision is verified on the basis of the business architecture)
- initial and target business, data and application architectures.

These documents are needed to verify architectures developed according to the ADM, where the sequence of manufacturing and management (and documentation) processes is as follows:

- business architecture development
- data architecture and application architecture development
- technology architecture development.

According to the TOGAF standard, the sequential development of components of IT systems architecture, data and applications, is not necessary. There is an opinion [23] that the influencing factor driving the development of an architecture is data. On the other hand, in the case of companies with ERP (*Enterprise Resource Planning*) systems, a different approach is used, which focuses on identifying the key applications underlying the development of architectures and integration methods. In the TOGAF standard, another mixed approach can be found, in which the initial data architecture development is followed by an application which processes and archives data. It should be noted, however, that the aim of this approach may be to produce application and data architectures, or just data or applications for business processes supported by IT. For the IPP, the sequence of architecture development is only relevant if the architecture development process increases (scalar) project negentropy. Project negentropy can also increase by using appropriate management processes:

- the requirements and the declarations of the provider referring to work on architectures
- the application of rules relating to data acquisition and architecture development
- ensuring the availability of the project Continuum, and more specifically – its artifact blocks (created in a supporting process)
- technical requirements relating to the development of IT systems architecture
- a gap analysis showing areas of business architecture changes, to which data and application requirements and technology limitations should be referred.

IT Systems Architecture – Data Architecture. When discussing IT systems architectures, the possibility of aggregating data and applications in the IPP was mentioned. To identify potential areas of aggregation, the main types of enterprise data and its sources are defined in data architecture development - paying attention to data structures and how they are aggregated. Selected Continuum resources (*e.g.* data models) can be applied in the IPP (within the proper process) in data architecture development. When developing data architectures of companies, manufacturing documents are required which contain:

- a physical data model
- a logical data model
- data management models
- data architecture (model, method, model development environment).

As in the case of the manufacturing documents, management documents are created here too. However, instead of analysing a set of available management documents in the IPP, it is better to analyse only their types (initial, main, transfer) such as:

- initial documents (principles and patterns for architecture development)
- main documents (for architecture development and implementation processes)
- transfer documents (*i.e.* implementation and gap analysis).

Two types of processes should also be mentioned: manufacturing and management.

Examples of systems architecture manufacturing processes include:

- preparation of the initial description of a data architecture in the scope necessary for creating a target data architecture
- construction of a business data model (entities, attributes, relationships) on the basis of ERD and FHD diagrams, facilitating the creation of data architecture views from the perspective of the client and the provider
- preparation of a logical data model (logical data views for different perspectives)
- creation of data management models, including:
  - views of availability and the level of data security (and the cycle of obtaining it)
  - a data matrix, including data entities and their business functions
- development of a data architecture
- selection of data blocks from the Continuum (within the proper process)
- verification of the model of architecture containing blocks
- analysis of the influence of the developed architectures (vision and business) on the developed data architecture:
  - evaluation of areas where the business architecture (*e.g.* commercial practices) or application architecture can require changes in the data architecture; in the case of significant changes, preparing a new architecture may become necessary
  - identification of existing limitations for the technology used in the data architecture
  - modification of the data architecture.

Examples of management processes may include:

- verification and approval of principles and patterns for architecture development (the application of opinions related to these principles and patterns, as well as to the tools for their manufacturing)
- review of business functions developed for the business architecture
- review and verification of quality criteria (such as efficiency, reliability, safety, integrity)
- performing a gap analysis and creating a report in such a way that the prepared data model includes all the goals of information processing in a company, and indicating, in particular, if the developed model:
  - includes data – where it is indispensable
  - indicates the deficiency of necessary data for the developed data architecture
  - shows the significant data which are unavailable
  - exposes the data not used in the construction of the application.

The implementation of the TOGAF standard in enterprise architecture development is faced with difficulties due to incomplete descriptions, data inconsistencies and unclear assumptions underlying the architecture development process. An example of such problems – in the development of architecture blocks in the (supporting) project Continuum - is the analysis of gaps (deficiencies) in the specification of requirements in the Continuum resources.

In general, the concept of system functionality represents the provider's point of view, and the requirements specification usually relates to the client's perspective. However, in many workplaces, they occur interchangeably, which allows project maturity to be consistently analysed (where the change in negentropy results from the state of the specification and the degree of the implementation of functions) along with the maturity of the client and the provider within the maturity capsule.

IT Systems Architecture – Application Architecture. The purpose of this development process is to identify the main types of applications used in a company for data processing and business support. Applications which meet the requirements of the company are shown, as well as the way in which these applications manage the data. No specific technology to be used in the application construction is indicated. From a business perspective, it is the business models and applications which are important, rather than technologies which change over time. Only a change in business needs implies changes in the architecture, which supports business processes through the appropriate model. Therefore, the Continuum resources are analysed to find suitable models for the application architecture. In the case of this architecture, a gap analysis should also be taken into account – to establish which applications have been ignored, or which processes have not been included in these omitted applications. In a company, a matrix model of gaps [64] is created, on the basis of the TOGAF standard, as was the case in the previously discussed IT systems, data and application architectures. When analysing the construction of an application architecture, from the point of view of the required data, the expected results and the particular steps of execution, it becomes evident that it bears a strong resemblance to a data architecture. For this reason, both architectures will be integrated in the IPP procedures.

IT Systems Architecture – Technology Architecture. The development of a technology architecture is the next stage in the development of enterprise architecture. As was the case when discussing the development of an IT systems architecture, below we will describe the (technology independent) development processes of a technology architecture, grouped and arranged in sequences/steps, as well as the management processes of its manufacturing. These groups of processes should be included in the initial processing, in determining the value of scalar project negentropy. The specification of architecture development process groups, and of the preparation of documents and management, provides a good basis for determining negentropy. In this situation it is convenient to apply the initial processing procedure which facilitates the evaluation of project negentropy.

It is understood that the development of a technology architecture is implemented on the basis of two main steps: the development of the initial technology architecture and the development of its target version. Initial and target technology architectures can be created from scratch, but they can also be obtained from the Continuum resources.

Architectures Included in the Continuum. The enterprise architecture development processes were discussed above, along with the possibilities to use data related to the manufacturing of products, preparing documents and the IPP. Within the proper process of architecture development, developers quite often refer to the Continuum. Below, the structure of this repository and its adaptation to the concept of the IPP will be discussed in detail. The Continuum contains typical enterprise architectures as well as ABBs – blocks dedicated to particular functions of a company. It is important in the construction of the IPP for the selection of IT technologies. Its structure (architecture elements/blocks and complete architectures) can in fact increase the value of (scalar) project negentropy. Using these blocks speeds up the development process and reduces the number of developmental errors.

The structure of the architectural resources in the Continuum is shown in Figure 2.4, which illustrates the development and collection of specific architectures (the repository contains the resources of many companies), based on generalised architectures (of higher levels) and enterprise/business-specific architectures (of the lowest level of generalisation).

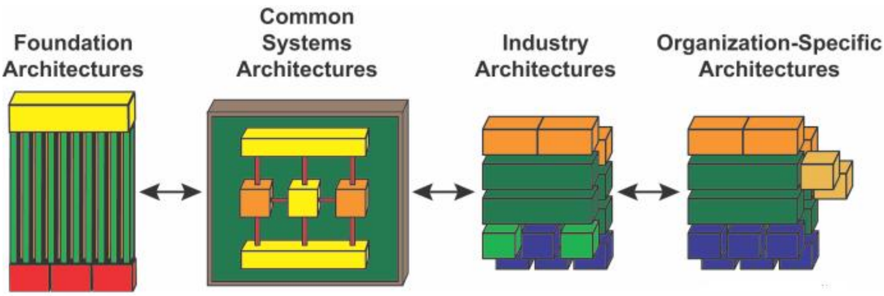


Fig. 2.4 Architectural resources in the Continuum arranged from general to specific

In the course of the project, its data is entered into the repository of enterprise architectures. The description of a new entry is compared with the existing descriptions of industry, common systems and generic/foundation architectures. If the resources contain an architecture which meets the requirements placed on the developed architecture of a company, it is automatically introduced to the enterprise architecture resources. In the absence of such a pattern, the process of developing an architecture for a company begins.

The architectures in the Continuum are thus collected in the following order: generic architecture, common systems architecture, industry architecture, and finally specific architectures of companies. The arrows in Figure 2.4 represent the two-way relationship between neighbouring types of architectures. The arrows to the right indicate a focus on the specific aspects of companies and business requirements, and the arrows to the left, on the more general architectures and their blocks, to the generic architectures. The real needs of a company and its business requirements are better represented by approaching (from left) to the right.

If an architecture for specific needs is searched for in the Continuum (at the stage of requirements analysis), and is not found there, the requirements are passed on to a higher level of generalisation, *i.e.* the architectures of common systems and the generic architectures. Then (in the proper project) the architectures available in the Continuum are confronted with the specific needs of the company, based on more general requirements adopted for a group of companies (a common enterprise architecture). Therefore, to improve the enterprise architecture selection process, the architectures located in a supporting process in the Continuum should be adequately assigned to one of the levels of generalisation/detail (Figure 2.4) in the functional description:

- a general description of company functions (generic architectures)
- more detailed functional descriptions of companies (common systems architecture and industry architecture)
- detailed descriptions of company functions (according to the enterprise architecture taxonomy existing within the Continuum) – when the level of detail in the functional description is high, the process of searching for/the selection of architecture is possible only if a common taxonomy for the Continuum is maintained.

Here are the specifications of the architecture resources mentioned above.

Generic Architecture. Because of its versatility, a generic/foundation architecture provides the best basis for the (proper) construction of architectures (those specific to businesses), within the ADM, and on the basis of the Continuum resources (rules and principles). It is based on the two main parts of enterprise architecture (the architecture-component part and the methodological part, introduced on p. 20), expressed in terms of TRM and SIB models and, according to The Open Group, has the following features:

- it reflects general business requirements as well as the enterprise architecture development technology and its blocks
- it treats solutions for developing architectures and their blocks as goods and services
- it provides a basis for the development of specific functions of a company in the form of common architectures
- it provides access to the architecture development environment according to the standards and principles of the Continuum
- it provides access for architects (according to the Continuum taxonomy) to a system of standards and guidelines.

The term: Standards Information Base Model, SIB – *a Continuum information standard, to assist with the development of dedicated enterprise architectures, on the basis of generic architectures.*

The term: Technical Reference Model, TRM – *a Continuum technical standard, taking into account software platforms, development environments, enterprise architectures and their blocks used in development.*

Common Systems Architecture. Common systems architectures are examples of Continuum architectures, each of which maps the needs of a group of companies (*i.e.* the common areas of their operation). For example, security architecture, management architecture, network architecture, *etc*. may be wholly or partially used in the development of the security architecture, management architecture or network architecture of a company in which enterprise architecture is being implemented. None of the blocks of architecture located in the Continuum includes all the functions of a company. However, they are complete in terms of specific areas (ease of management, networking, security, *etc*.). Other features of common architecture blocks:

- reflecting specific client requirements (but only in the sense of common elements)
- the possibility of defining new blocks for a particular area (described by the client) of the company's operation
- setting technical standards for the development of blocks according to the Continuum resources
- providing easy access to blocks (due to clear classification).

Industry Architecture. Common systems architectures, presented above, focus on common functions, whereas an industry architecture refers to common technological blocks (based on Java 2EE and .NET platforms) for a given industry. In terms of common architectures (systems), the division of functionality in terms of business areas helps developers select the architecture. The inclusion of an industry architecture in the IPP facilitates the application (inside the IPP) of finished blocks representing the various functions of the industry. An example of such a (local) Continuum block is the Petrotechnical Open Software Corporation data model, POSC.

When constructing the initial processing procedures (IPP), attention should also be paid to other features of an industry architecture which give the opportunity for:

- defining specific requirements and standards for a given industry
- manufacturing universal blocks for a particular industry
- classifying data and processes
- classifying applications and processes (for a specific industry), as well as business rules
- classifying guidelines for (self-) testing the industry architecture.

Enterprise Architecture. This is a group of architectures of individual companies or a group of companies (with a common IT system, groups of companies) which may be taken into account in the analysis of the IPP.

In addition to the basic structure of a company, its components and their mutual relationships, this architecture includes guidelines (a type of project document), which contain information on how to identify the Continuum resources and the methods of their use. The IEEE Std 1003.23 standard is an example of such guidelines, which includes the allocation of processes to software development cycles and a description of ways to document the process in which a client uses enterprise architectures as well as the services supporting this process. Based on the distinct architectures of various companies, enterprise architectures integrate their various processes. Clearly, when choosing (out of such a group) the architecture for a particular company, it is preferable to choose an architecture that best suits the company's needs.

Enterprise architectures (in addition to the basic structures of a company, its components and their relationships) include the guidelines of their use. The IEEE Std 1003.23 standard includes the identification of processes (with cycles of software development) and the description of ways to document the methods of using enterprise architectures and support services.

Enterprise architectures enclose a variety of different business processes (their distinct architectures are included in the Continuum). Therefore, if an enterprise architecture is to be re-used, it must be suitably adapted to current requirements.

Enterprise architectures (due to the descriptions of companies and standards used, contained in the Continuum) are characterised by the following features:

- they reflect requisites, blocks, applications and models of processes specific for a firm
- they support anticipated communication methods between companies
- they support management of the IT environment by providing criteria for selecting products, solutions and services, and by showing the appropriate means and tools
- they present evolutionary ways of supporting the development of an organisation and the formulation of new business needs.

Implementation Solutions. Components. In the TOGAF standard (version 8) it is assumed [63] that *implementation solutions* reflect the enterprise architectures as a set of system components.

In the project Continuum described above, as well as in IT, the philosophy of collecting models of systems (as enterprise architectures) and their technological implementation is commonly practised. Bearing in mind that the project Continuum is a repository which collects resources for multiple use by developers of architectures and IT systems, it is assumed [63] that implementation solutions (system components illustrated in Figure 2.5) are collected in the repository according to a structured division, which reflects the architectural resources in Figure 2.4, *i.e.* generic solutions (products and services), system solutions, solutions for an industry, and solutions for different organisations. Moving to the left in Figure 2.5, more general components are shown (those convenient for the provider, in particular at the highest level of generalisation; generic reference blocks represent finished products and services), and moving to the right, more detailed components can be seen (system and industry components and finally those dedicated to individual organisations).

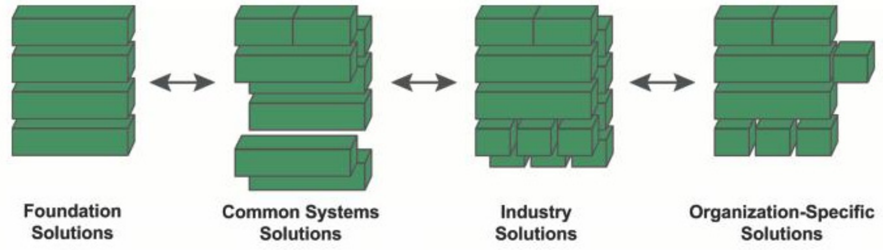


Fig. 2.5 Implementation solutions: components – from general to specific

Products and Services. In the concept of architecture implementation used in TOGAF, products and services are the main results of development. A separation of products from services may be related to the diverse profiles of client organisations.

System Solutions. These are a group of system components reflecting technological support for businesses. Examples of such universal components are: databases, IT infrastructure, operating systems, (IT infrastructure) resource management systems, monitoring systems and control of automation devices, *etc*. System solutions can support the existing IT systems of companies or be used both in the form of products and services. Such services are used only when the common systems architecture (Figure 2.4) is based on SOA (*i.e.* uses services).

Industry Solutions. These include a set of system solutions designed for specific industries. Their application facilitates the implementation of system solutions in the industry. The solutions proposed for implementation can also include the specifications of certain industry architectures. Such a specification is prepared on the basis of the requirements of individual companies of a selected industry for specific products, services and systems expected by the industry.

Dedicated Industry Solutions. These are characterised by high specialisation. Before they are introduced to the Continuum (in the supporting process of development) they must be assigned to a particular group of industry-specific solutions. The industry solutions described above are often analysed in terms of their application in the development of dedicated industry solutions for products and services in a given industry.

The Relationship between Architectures and the Continuum Components. The structure of the Continuum maintains the proper relationship between systems architectures and implementation solutions. Such a relationship is reflected in Figure 2.6, which shows the division of architectures (Figure 2.4), the division of implementation solutions (Figure 2.5) and the use of architectures and system components, through the search mechanisms of the resources of components/ solutions (Figure 2.6). The mechanisms are built on the basis of guidelines containing descriptions of the resource access procedures and their support via search algorithms.

The presented relationship between architectures and applied solutions (according to Figures 2.4 and 2.5) creates the way in which to use the Continuum resources (Figure 2.6). An example of such a relationship (constructed in the supporting process) is the relationship between the generic part of the Continuum architectures and the products and services placed inside the Continuum resources. For this reason, the generic architectures include guidelines for manufacturing products and services. A similar relationship exists between other resources of the project Continuum (architectures and components). The knowledge of these relationships facilitates the process of using the Continuum (its products and services).

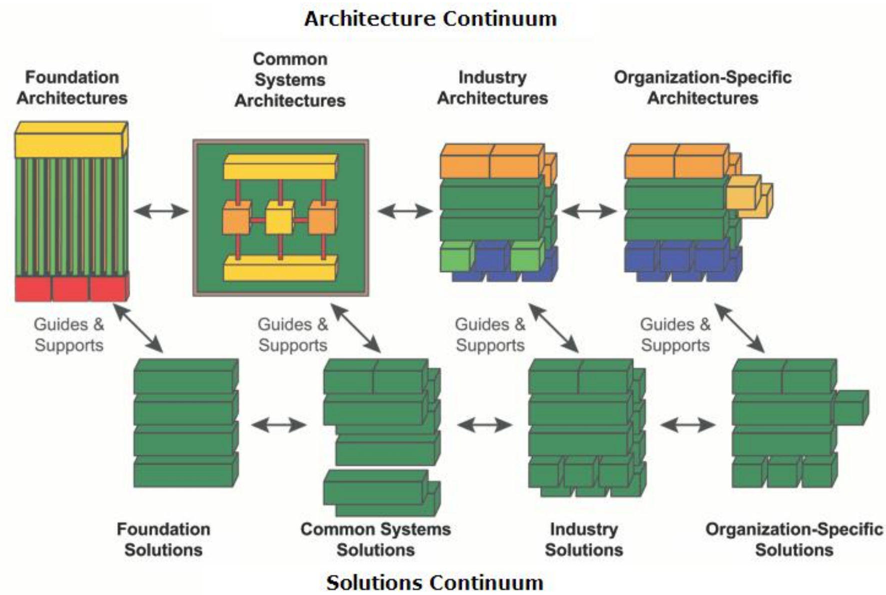


Fig. 2.6 The project Continuum: architectures and their concepts

Own Company. In the TOGAF standard, the basic way of describing one's own company is by its architecture. While developing it, the architectures contained in the Continuum should be considered. For example, when assessing the level of the architectures of one's own IT systems, it is recommended to use a generic architecture (contained in the Continuum). It is recommended to have architectures of common systems and management systems for developing them. Specific industry architectures, whose job is to represent the functionality of systems within a given industry, can be analysed in the IPP. Taking one's own company or its branches into account, a demand for a common architecture could be made, which is in accordance with the ADM. Then the process of developing the architecture of one's own company will mean adopting or adapting existing Continuum architectures or developing one from scratch.

2.5 Model of Information Technology Management MITM

Non-autonomous models process some input information. In the case of the multi-level selection of methods and IT tools, MSM (Figure 2.1), client and provider maturity and project negentropy, contained in the capsule, is such information.

The MITM output variables present growth in the functionality of the technology used in the development and the management of the teams or organisations of the provider and the client. To help developers to assess the input variables, we suggest a support procedure in the form of MITM processes based on the initial processing procedures (IPP) and the maturity capsule.

The description will begin with elementary IPP procedures, and will use the knowledge contained in such IT standards as TOGAF, ITIL and COBIT (see Section 2.2).

2.5.1 Methodology of Evaluating the Primary Negentropy of a Project

Let us focus first on the IPP related to evaluating (secondary) project negentropy which, defined by (2.2), expresses the degree of maturity of the project organization. This concept borrowed from cybernetics represents the state of the system, which should aim at the target point of saturation. In our approach, the degree of scalar negentropy/maturity of the project comes to the value of 5 – which describes the final state of scalar negentropy, corresponding to saturation in terms of technology and domain. Negentropy increases with the development of the project.

From this point of view, projects can be classified as:

- *embedded* (lack of applied information technology and domain knowledge)
- *semi-detached* (partial application of the latest information technologies and partial knowledge of the domain)
- *organic* (full application of IT and full knowledge of the domain).

This classification is consistent with the assumptions of the COCOMO model [4] and the division of projects into groups. This formalises the premise that, with the development of the project, not only does the knowledge of the project domain grow, but also the height of IT tools used increases. The consequence of positive changes in the level of domain knowledge and in the extent of the application of the latest technologies is the increase in project maturity during its evolution.

However, a question arises about how to allocate a specific level of domain knowledge and a specific height (level) of applied information technology to projects from higher groups (according to the COCOMO classification). This knowledge constitutes the basis for the estimation of primary project negentropy.

On the basis of knowledge acquired from the TOGAF standard resources, it can be assumed that the evaluation of the knowledge and understanding of the project domain and the applied technologies are: the manufactured artifacts and products of the IT system, as well as the project documents which the standard describes in detail. The quality of their manufacturing is also a derivative of the level of manufacturing and management processes. It is assumed (according to TOGAF) that the evaluation of such a classification in the analysis of project negentropy is based on the knowledge of the documents and architectures that arise during the development of a complete enterprise architecture.

Further in this work *project negentropy* will be expressed by the following variables: the *width of the repository* of architectures (a_t), which identifies the way to develop architectures, the *length of the catalogue of documents* (d_t), which determines the level of document manufacturing and the *height of the ITM section* (pr_t), which describes the manufacturing and management processes, and the IT technologies and good practices used in architecture development.

The selection of variables to describe the elements of primary negentropy is rooted in the experience of IT specialists, according to which the quality of the products (architectures and systems) is evaluated on the basis of the manufacturing processes. A detailed description of a product according to TOGAF suggests a different approach. Here we offer a hybrid solution in which the basis for the quality evaluation of the system (its architectures) are the processes of development, manufacturing, management, and the products (project and architecture documentation). At the present stage of the development of information technology, this solution emphasises the meaning of the process approach (the height of the ITM section) and the product approach (the length of the catalogue of documents and the width of the architecture repository). The inclusion of these three variables in the framework of project negentropy opens the path to a 'product-based' view of the achieved results of the development of architectures and IT systems.

Initial Documents

The main document is the contract for the project. A detailed and properly prepared contract means that the provider knows the matter in question and has knowledge on how to implement it in the framework of an IT system. The preparation of a contract for an *organic project* involves recognising the domain and preparing an order for the implementation of the IT system (which specifies the details and the quality criteria of the products during the implementation). On this basis, the level of detail of the preparation of the contract is established, as well as of the subsequent implementation of the project, including:

- defining a system for continuous monitoring and auditing of the manufacturing level and the subsequent maintenance of the system in terms of its functionality, in order to check its integrity and make changes

- complying to principles, standards and requirements applicable in manufacturing
- identifying risks in various aspects of development and implementation including: the internal development of the system in reference to the standards, principles, technologies, products and operational aspects of development related to the company's activity
- a set of processes and practices to ensure the accountability and implementation of procedures in terms of manufacturing, with the use of available architecture components.

Organic projects require an initial diagnosis of the domain and a client representative to realise the project. Then a contract for manufacturing the system, at various stages of the manufacturing process (*e.g.* when developing architectures) can be prepared - between the architect and the business users who will build and implement the specific elements. It is worth remembering that in such cases the ultimate goal is not simply the company and its systems, but a dynamic enterprise architecture, which allows the flexibility to change in response to changing conditions. Such preparation of a contract reflects a high level of domain knowledge.

An example of a contract for an organic project is a contract between the architecture developer and the manufacturers - a declaration of intent on the enterprise architecture development, entirely or in a substantial part with partner organisations, including the system integrators, the providers of the application and the service providers. The development of one or more architectures (of business, data, application, technology) can be outsourced, with the defined functions of the enterprise architecture and providing the supervision of target architecture together with the coordination and control of the overall effort. The supervision can be outsourced - although most companies prefer self-reliance in this aspect. Regardless of specific contracts for agreements and arrangements, most of the work is carried out under a contract which specifies the services, relevance and quality of the products (architectures), as well as the manufacturing processes which will be implemented jointly. After signing such a contract, the project can be considered as organic.

After defining the specification of the contract and its applicability and evaluating the level of domain knowledge and of the information technologies used in such an IT project, the processes of its implementation will be discussed. Those processes are usually treated as services. Their organisation is fundamental for projects to be classified as organic. In order to make such a classification, the management of services must be considered – in terms of the method of access to these services and of controlling their security. What should also be taken into account are the common services and the data normalisation method (*e.g.* accuracy, control consistency, data editing, encryption of transactions and management), as well as the place of storage. The method of data access should also be specified, considering the following:

- requirements in terms of access to data, files and messages as well as data management (entering, deleting, or selecting)

- type of data stored and the logic of its location
- possible application of the query language.

The state of data security should also be taken into account in terms of:

- ability to provide a security policy and guidelines for its control
- possibility for identification/authentication of data
- authorisation and data access control
- protection of confidential information
- control of access paths and preparation of an audit log
- external factors.

The use of the system in a company requires changes, representing the transformation taking place in the company.When making these changes the following should be taken into account:

- documenting the transformation of business processes
- managing versions of the system (software)
- frequency of generating versions of the system
- characterisation of tools for making changes in the software
- license conditions (of the system) and the possibility of making changes.

The analysis of the course of realising the IT services, in the context of the contractual conditions, shows the purposefulness of the preliminary evaluation of contracts. The classification of contracts presented in this Section emphasises the importance of a multi-faceted evaluation, especially from the point of view of the project environment. The evaluation of a contract (based on the description of the manufacturing processes and products) should allow at least an approximate estimate of the source negentropy of the project. In general, the evaluation of monovalent negentropy (in rough categories) is not easy. However, the same evaluation system (on a scale from 1 to 5) can be adopted for the evaluation of the contract and the services provided by the supplier, as was the case in the evaluation of provider organisation maturity. The classification of projects in terms of their current negentropy can be useful for scalar evaluation – similarly to the evaluation of provider organisation maturity (discussed further in Section 2.5.2).

Products/Artifacts and Their Documents

When evaluating project negentropy, all the factors which significantly affect its state should be taken into account, whether it is in absolute terms or relative, or dynamic: such as documents and the length of their catalogue and the products created during the development process (expressed in the width of the repository), and their production processes (manufacturing and management).

The analysis which follows here is of architectures, documents, and processes which relate to complex projects (unlike the component projects: IT systems, technologies, *etc.*) with low (final) negentropy (whose scalar value does not exceed 3 in their life cycle), which is the development of entire enterprise architecture (a set of architectures by TOGAF). The choice of this type of architecture is related to the intention of evaluating the suitability of the proposed *negentropic* (based on negentropy) classification of projects with architectures of high complexity.

In Section 2.4, by analysing the types of architecture, documents and manufacturing and management processes, we presented those which occur in the development of any IT systems architectures (the development of EA includes creating an IT system architecture). To describe current project negentropy, it is necessary to analyse the documents and the development processes of EA. Below we consider technology architecture development processes in detail.

Technology Architecture Development Processes

The processes of development from the initial to the target technology architecture are implemented in eight sequential steps (described below), which are to lead to:

- the evaluation of the previously developed architectures of business, applications and data, and of the preparation, made on this basis, of the description of the initial technology architecture; the scope and level of detail of the description depend on the level of the descriptions and the quality of the available components, as well as on the detailed description of the target technology architecture (where the degree of application of the components must be evaluated)
- the development of the target technology architecture.

Step one requires the following manufacturing documents [63] describing:

- the principles of the technology architecture development
- the technology architecture
- technology architecture blocks.

Within this step, the following manufacturing processes are also realised:

- collection of data referring to the existing IT systems and the constraints connected with the development of the technology architecture, in the form of the constraints document
- preparation of a list of separate functions unrelated to the technology architecture design
- verification of the correctness and functionality of the developed architecture
- conversion of the existing documentation of the systems into a description complying with the terminology for developing the basis of the technology architecture
- creation of the concept of blocks (components) of this architecture – ABBs (*Architecture Building Blocks*), the aim of which is a multi-faceted view of the developed technology architecture – through the involvement of specialists and by giving them access to an environment for multi-faceted evaluation, the process of development is facilitated

- defining the space for particular architectures – to improve the use of resources, and (through ABBs) to allow the collection of the necessary services and applications
- preparation of interfaces to communicate with the application architecture
- development and verification of the technology architecture
- preparation of test questions to authenticate the benefits of the technology architecture.

When building the technology architecture, the following management processes are needed:

- verification of the correctness of the technology architecture development principles, containing guidelines for the development and application of these principles; it is created on the basis of sample sets included in the TOGAF resources
- analysis of other frameworks for the development of architectures which enable the development of the technology architecture of a company with the use of functionality components included in the TRM standard
- analysis of the relationship between the manufacturing frameworks and the requirements posed on them
- restructuring of the technology architecture development team (including training on the subject), and being in the possession of the initial and the target technology architecture – the members of the team should have experience with architectures (ADM) and be able to adapt them to the requirements of the technology architecture development process; it is also important to determine the key points which constitute the basis for evaluating the effectiveness of this process
- documenting the manufactured architecture blocks using the architecture description language – such documents (stored in the repository of initial architectures or public IT resources) may be useful in the case of the re-use of developed architectures
- development of the technology architecture including the principles of its construction, the restrictions, the requirements analysis in terms of its construction, a list of key questions and the criteria for their selection.

In the <u>second step</u> (of the transition to the target technology architecture) the use of different reference models of architectures, blocks (multi-faceted views) and the selection of tools for their construction is considered. The purpose of this step is to perform an analysis of technology architectures taking into account diverse requirements and viewpoints – to ensure that the target technology architecture meets all the requirements placed on it.

In this case, it is worth using a proven business architecture, which provides the partners with an environment facilitating the selection and presentation of their own, most important musts (views, viewpoints) for the project. Ideally, the architecture should meet all the requirements of the project partners (since only then will the compatibility of the target IT systems architecture with the business architecture have sense). Documents describing the most important views will be necessary for this. The selection of viewpoints is an important element of the requirements and a success factor in creating a technology architecture. This can result from an analysis of the TOGAF resources and the needs and requirements of the company. In this step, the following manufacturing and management documents are prepared:

- the document of block modelling techniques and tools
- the technology architecture
- architecture blocks.

For these documents to be created, the following manufacturing processes must be applied:

- selection of technology architecture resources (reference models, development patterns, *etc.*) from the Continuum (on the basis of the application requirements)
- selection of technology architecture views (those which enable the architect to incorporate client requirements) – or at least the following viewpoints:
    - of the network hardware
    - of partner communication and resource processing
    - of the cost of the presentation.

Such choices can be made through "brainstorming" and developing requirements documents providing views of the entire system. A redundancy of views is the preferred solution. When developing the application architecture, the following management processes are used:

- selection of techniques and tools appropriate to the acquisition, modelling and analysis of viewpoints – depending on the level of knowledge of the developers and the perceptual capabilities of clients, documents, spreadsheets, or more sophisticated modelling tools and techniques (environments based on modelling standards) can be used
- identifying conflicts among the stakeholders representing various viewpoints on the architecture.

The third step is related to the development of architecture views within the defined blocks. In contrast to the first step (which defines the functionalities), this step defines groups of viewpoints unifying the functionalities developed in the second step. The restrictions adopted here, in terms of the development of the Continuum data architecture, can be used to verify the *general architecture model*, which takes into account system requirements (within the proper process). Such a general architecture model is the TRM, created in step 1 by the mapping of functions onto the service.

The construction of this model allows for the requirements of many organisations interested in the project, specific in terms of their operation, to be taken into account. If, therefore, there are different viewpoints (reflecting the functions of the system), then the TRM will reflect any technology architecture of a company. It is then easier to decide how to implement particular elements of system functionality in the technology architecture. By grouping the various functions of the system into blocks, it is easier to determine how these functions (and blocks) will be implemented in a specific technology (*e.g.* client-server).

However, a question arises as to how the system functionalities grouped within the TRM organise the system manufacturing processes, increasing the value of scalar project negentropy. Two approaches can be suggested: (1) the construction of the TRM (by taking into account and grouping the specific functions of enterprises) organising the repository resources and thus supporting architecture development, (2) artifacts built within the repository (development documents and architecture blocks) are the basis for identifying the IPP. In this book, we propose the first approach, *i.e.* the construction of a project repository which supports the development process of enterprise architecture. The second approach can be applied with a potential extension of the linguistic IPP model.

Therefore, this step will determine how the blocks created within the Continuum (in the supporting process) will be applied (*e.g.* as a set of tasks and services produced in steps 1 and 2). It is desirable to use the project Continuum blocks for the development of new enterprise architectures. The architecture and component blocks of the Continuum should be evaluated in terms of the scope and completeness of the implemented functions of a company (within the proper process).

In consequence, the third step refers to the preparation of the following artifacts:

- the target business and technology architecture
- appropriate views (viewpoints) contained in the Continuum
- architecture blocks in the Continuum
- constraints of the technology architecture
- proposals for changes in the technology architecture (in the Continuum), resulting from an analysis of views
- requirements documents referring to specific functions of a company
- new blocks which could be implemented in the event of changes in the requirements
- tests of the completeness of the technology architecture (with respect to requirements)

as well as other activity, such as:

- taking the specific requirements of the company into account – the production of models of these requirements or the modification of existing ones (with minor changes to the functioning of the company)
- checking the conditions for all the requirements referring to the construction of data and application architectures
- defining the names of the blocks when they are stored in the Continuum repository – in line with the procedures of configuration and change management [60] – for their re-use

- solving conflicts, after diagnosing them [63]
- checking if the requirements models are in accordance with the principles, objectives and constraints
- paying attention to changes in viewpoints for blocks selected from the Continuum and documents – before they are used in the development of architectures.

Step four involves the selection procedure (from the Continuum) of a portfolio of services necessary for the construction of the blocks manufactured in the third step (as part of the support process). In the selection of these services, we try to provide support for developed application blocks and later assemble them into a complete and fully functioning architecture. After taking into account the constraints of step two, the services selected from the Continuum relate to:

- specific elements of organisations or support for decisions made by them
- development of the existing and unchanging components of organisations.

Step four requires the preparation of the following:

- documents of the target technology architecture, which include the objective of its construction and the description of the portfolio of services required to achieve this objective in the structures of the organisation (also known as a detailed description of the organisation structure),

as well as the realisation of the following processes:

- inference as to the changes in technology architectures contained in the Continuum
- analysis of the similarity of the products in the adopted group of services
- analysis of the type of services and the similarity of groups in relation to the requirements
- preparation of the portfolio of services for each block and carrying out cross-checks for hard-to-define services
- making changes in the structure of the Continuum (new blocks).

If there are requirements for specialised services not included in the Continuum, the choice of other services available in the Web resources should be considered, *i.e.* pre-testing them to verify their usefulness in the development of enterprise architecture.

Step five should result in a statement, based on a cross-sectional analysis, that the business (and other) objectives of implementing the architecture have been achieved. The basic method is based on a list of questions about the technology architecture and on testing its completeness and applicability. The manufacturing and management documents prepared as part of this step include a specification of the target business and technology architectures.

The goal is achieved through the realisation of the following tasks:

- a formal review of the technology architecture model and its blocks, in cooperation with the client, to confirm that the business objectives contained in the architecture model have been carried out
- verification of the architecture completeness through the use of a list of key questions
- agreeing with the client on the procedure in the event of changes to the requirements relating to the IT system (deadlines and the scope of the changes).

Step six refers to developing the architecture selection principles and preparing the target architecture specification. The selection of these rules depends on the state of the enterprise architecture (or in the absence of it, on the state of the IT systems of the company) and on the general objectives posed on it, which should be consistent with the business objectives of the organisation. Here are some examples of rules for selecting these architectures, which cover (in the supporting process) the enterprise architecture, on the basis of development principles which supplement the description of architectures contained in the Continuum resources:

- they must meet the requirements of the organisations and legal requirements
- they should be available to the public in the Web resources
- they have been developed in view of a variety of sources and resources of the company
- their development should be based on the use of the Continuum resources
- they should be complete, clear, and preferably from the Continuum
- they should be easy to test and integrate with other architectures of the company
- they should be prepared in several versions, varied in terms of technology, then the easiest one can be selected – in the event of problems with integration
- they should be constructed based on the model of the company and regardless of the IT (in the creation of such models, great meaning is given to 'good functional requirements', reflecting complete and consistent [3, 56] characteristics of a future system which are necessary in the construction of IT systems fully reflecting the needs company), and also:
- new architectural requirements can not interfere with the demands posed on the existing, older systems of the company.

The realisation of these tasks (step 6) is based on knowledge of the target business and technology architectures, or the resources in the Continuum containing these architectures. Moreover, the purpose of step six is carried out by running the following develoment processes:

- brainstorming the selection criteria of the architecture (sources: previously used criteria for the existing system and their extrapolation to the new architectural elements)
- assessment by the sponsors of the current state of the architecture, in order to negotiate the requirements with respect to the target architecture,

and the development of the documents of:

- the target technology architecture
- requirements of the technology architecture (a description of the selection criteria).

Step seven refers to the construction of the full technology architecture. This is a multi-stage process in which the selected blocks and interface functions imply the method of employing the requirements. The target technology architecture can be developed in two stages:

- preparing the intermediate description of the architecture (*e.g.* pseudocodes, diagrams)
- supplementing (for completeness) the architecture description (*e.g.* adding diagrams of systems configuration, specification of the operating systems, network functions).

In the process of building a full technology architecture, the selection of blocks (from the Continuum) has a character similar to the selection of blocks in the construction of IT systems architecture. For the process of selecting the blocks (in the proper process) to correspond with the requisites of the technology architecture, the postulates of the enterprise architecture should be implemented in the form of services. This will facilitate the process of selecting blocks and the development of the technology architecture, and increase the negentropy of the project. In the blocks selection, all the services, both the existing ones and those being currently developed, should be taken into account according to the requirements of the company. With the selection of blocks (as services), the development process of the technology architecture becomes well-organised. The following examples of service levels (business requirements) can be enumerated:

- LEVEL 1 – implementation of security policies and management procedures, and basic functions and attributes of the IT system, with a semantic description
- LEVEL 2 – application of interfaces in the integration processes
- LEVEL 3 – inclusion of the Application Protocol Interface, API, data formats, protocols, hardware interfaces, and standards for their implementation
- LEVEL 4 – addition of maps of business processes, organisational units and strategies.

The details of constructing the blocks, implementing specific solutions as SBBs (*Solution Building Blocks*), and of creating interfaces for them should be included in the specification of the architecture. SBBs are a means of showing how the components of the target architecture can be acquired, developed or re-used. The architectures of SBBs should contain developed elements (of the Continuum), and each of them should go with a minimal specification description.

The realisation of step seven is based primarily on the documents of the target business architecture and technology architecture, architecture blocks, and SBBs. Here is the sequence in which to prepare the documents:

- a final document mapping the Continuum architecture onto the project
- the target (proper) technology architecture which meets the specified requirements with a description of the supporting mapping of the developed architectures onto the Continuum
- a report with the results of the development of the technology architecture (a description of the developed blocks, a description of the block development principles, a description of architectures obtained from the repository, and the principles of their use)
- justification of the decision to build the technology architecture blocks (based on the description of the technology architecture blocks in the Continuum)
- a document of the sequence and quality control of all the documents and reports prepared within step seven,

as well as the realisation of the supporting processes:

- presentation of the architecture created from the selected blocks and the identification of those blocks which can be re-used
- publication of the Continuum resources
- obtaining full documentation of all data formats, protocols, hardware interfaces, and interfaces for each block (API)
- making a full analysis of the overall architecture in relation to the business requirements
- preparation of a report on the design results and principles of the technology architecture
- determining the areas in which architectures (of business, data, applications) can induce changes in the technology architecture (if the changes are significant, it is necessary to modify architectures of business, data, and applications)
- modification of the technology architecture, if necessary
- selection of development standards for the technology architecture blocks and re-using some selected ones essential for constructing Continuum reference models (a safer method, reducing the possibility of error)
- extending the documents of technology architecture creation with development standards
- checking the level of work on the architecture on the basis of the documentation.

Step eight aims to validate the areas (both in the existing and in the target system), in which the specifications have not been taken into account. A key step in the validation of architectures is to consider the possibility of providing these specifications; a gap analysis in the developed specifications of the technology architecture. To complete this architecture with the missing specifications (which provide more of its details) a report is prepared, which:

- refers to the construction of the technology architecture, and contains a summary of key findings and conclusions, and charts and diagrams to aptly illustrate the state of the specifications
- contains criteria for selecting specifications for the developed technology architecture.

2.5.2 Methodology of Evaluating Provider Maturity in the IPP Model

After discussing the initial processing procedures, IPP, for project negentropy, we will move on to characterise them in reference to provider maturity. In the analysis of the initial processing procedures, the classification of projects, as accepted in COCOMO, was used for the external (project environment) evaluation. In this case, the internal classification (of the provider) can be used. Therefore, to evaluate the maturity of an organisation the CMMI model is suggested. This is essential in the manufacturing of systems for enterprises. The IPP requires an evaluation of the maturity made preferably on the basis of different criteria. One of these might be the effectiveness of the processes of change management, such as the improvement by using IT in processes leading to the expansion of the company [26]. Another criterion could be compliance with the process maturity model (CMMI). The use of the CMMI model brings the following benefits to the organisation and the project team:

- description of all the organisation's practices (to improve its processes)
- establishing the determinant of implementing changes
- establishing a proven framework of process improvement

For this reason, the use of CMMI models for the specification of the IPP is important in the (frequent) case of evaluating a provider organisation which has a low level of maturity.

To decide on the application of the CMMI model, its brief characteristics should be presented and then the suitability of this model (in whole or in part) for the IPP can be considered. This model makes it possible to evaluate an organisation in terms of its five levels of maturity in a given area. It also helps to select the practices on which the organisation should focus to get the greatest improvement. Many American institutions, such as the Software Engineering Institute (SEI), the Federal Research and Development Funding and the development of a centre sponsored by the U.S. Department of Defense and supported by Carnegie Mellon University, are involved in the development, expansion and maintenance of the following models:

- CMMI (Capability Maturity Model Integration)
- IPD-CMM (Integrated Product Development Capability Maturity Model)

- P-CMM (People Capability Maturity Model)
- SA-CMM (Software Acquisition Capability Maturity Model)
- SE-CMM (Systems Engineering Capability Maturity Model)
- SW-CMM (Software Capability Maturity Model).

The multiplicity of maturity models has led to their own problems. The SEI therefore decided to integrate all the models for the comprehensive evaluation of an organisation's maturity. In order to provide a means to manage this complexity, CMMI was created. According to the SEI, CMMI models include the best practices from the previous models in many important respects, in particular, such models enable organisations to:

- better express discrepancies between management and manufacturing to achieve business goals
- expand the scope of activities by adding the application and applicability evaluation (used by organisations) of software life cycles and requirements technology (in order to meet client expectations via the supplied product or service)
- combine applications (referring to measurements, and management of risk and providers, for instance)
- introduce more efficient management and manufacturing practices to implement the developed IT system
- implement additional organisational functions in teams and assess their impact on the development and the management of the manufacturing of products and services
- introduce manufacturing processes fully compliant with the relevant ISO standards.

The growing demand for software development methods (relevant for organisation maturity), resulted in some interest in a group of tools (which are an integral part of these methods), which are templates for the evaluation of the processes of development and architecture selection as well as the evaluation of manufacturing organisations. This approach can be used in the planning of team members' qualifications, in identifying training and development needs, *etc*., which are components of the development process of architectures. These were used to evaluate the maturity of companies developing enterprise architecture with the use of the Architecture Capability Maturity Model (ACMM), helpful in conducting internal evaluations of organisations. The ACMM proposes a framework for developing strategic architecture elements, in order to identify weak areas of the company and to indicate evolutionary paths for their improvement.

The ACMM model consists of three parts:

- the capability model of maturity for the design of an enterprise architecture
- the ACMM model concerning the architecture of particular companies

- characteristics (SCAMPI, scorecards, *etc*.) of particular processes of the development of company architectures (for different maturity levels discussed below).

The purpose of the first two parts is to provide solutions that support the application of the ACMM model to evaluate the maturity levels of the organisation (with justification). The third part is to present the results of architecture maturity which are to be reported to the CIO.

The ACMM document based on this model applies to one particular (current) level of the six levels of maturity and one (current) of the nine 'characteristics' of architecture.

In this document, the six *levels of architecture maturity* are: 0 – None (null), 1 – Initial, 2 – Repeatable (under development), 3 – Defined, 4 – Managed, 5 – Measured.

The nine *architecture characteristics* are: the degree of architecture designing, the degree of its development, and the levels of: business linkage, senior management involvement and team participation, communication necessary to manufacture the architecture, development security, architecture governance, project management, and investment and acquisition strategy.

Using those two measurements (maturity and architecture 'characteristics') to evaluate the maturity of the ACMM, the aim is to achieve a 'weighted average' architecture maturity and describe the achieved level of knowledge about each of the nine characteristics (aspects of the project's progress).

Maturity of the Architecture Development Processes. The following example shows the details of determining the level of maturity of enterprise architecture development processes.

Level 0: **No** maturity of enterprise architecture development processes.

Level 1: **Initial**: Informal architectures show the following characteristics during development:

- processes are *ad hoc* and localised, some architecture development processes are defined, in the formalisation of business processes a uniform method of development or the same technology do not apply, success depends on the efforts of team members
- architecture development processes are not formally documented
- limited awareness of team management or the need for participation in the development
- a small number of team members involved in the development of the architecture
- limited possibility of modifying documentation referring to architecture development by dispersed teams, limited communication about development processes of architectures and methods to improve these processes

- safety aspects of architecture development are available and defined *ad hoc*
- lack of or insignificant strategic planning, lack of growth of the architecture development team, no (full) compliance with the existing standards.

Level 2: **Repeatable**: Enterprise architecture is in the process of development, based on an iterative cycle of software development:

- preliminary architecture development processes are based on documented approaches
- views of the developed architectures are built according to accepted principles of development (TRM) and business connections for the clearly defined target architecture
- it is assumed that company functions are linked with business strategies of the Continuum
- there is an awareness of the need for managing architecture development
- manufacturing/development processes are divided into tasks
- documents of the development team are periodically updated
- management processes are based on several standards
- there is no formal management of manufacturing and data acquisition processes.

Level 3: **Defined**: This is characterised by specific architectures and the detailed process of architecture development assigned to them:

- teams (properly) define architectures, and then pass them over to business departments
- there are no gaps in the analysis and migration of the development plan, the TRM tools and profiles of development standards are applied, the aims and methods of IT are set
- the budget for the architecture development process has been planned
- managers of provider organisations are aware of the need for team management and the management of enterprise architecture development
- architectural standards management is applied
- most team members actively participate and/or accept the development process
- architecture development documents are regularly updated on the website
- safety standards of enterprise architecture development processes are fully implemented and integrated with enterprise architectures (Continuum)
- IT investment management procedures (initially determined) are documented
- there is a requirements acquisition strategy and measures to ensure compliance of the developed architecture with the architecture of the company (supporting project), and the cost of development processes are included in the project cost estimate.

Level 4: **Management**: Measurements are carried out here as well as the management of the enterprise architecture development process:

- the development process is indicative of a high level of culture in the provider organisation; indicators of the quality of manufacturing processes are used
- development documentation is updated on a regular basis (reflecting the current IT architecture) the architectures of business, data, application and technology are defined in accordance with the law and the state of matters (*de jure and de facto*)
- during regular updating documents are referred to the latest developments and standards
- the work of the investment planning and control centre is adjusted (on the basis of opinions and conclusions from the development process) to the current IT architecture
- the manufacturing team is directly involved in the architecture review process
- the entire team is actively involved in the architecture development process and accepts it
- performance indicators related to IT architecture security are recorded
- the organisation has experience in managing a variety of IT investments
- there are processes for managing discrepancies between the IT resources and the developed enterprise architecture
- all the planned purchases related to the development of the enterprise architecture are established and carried out by the team leader.

Level 5: **Optimisation** of the architecture development process is present on a continuous basis:

- coordinated action is implemented to optimise and continuously improve
- standards and processes focused on requirements are applied for optimisation
- the applied measurements of the development process are optimised and relate to business requirements, and business strategists are involved in the process of the continuous improvement of architecture development
- team leaders are involved in the optimisation of the development processes and use appropriate procedures
- architecture development documents are applicable to business decisions
- enterprise architecture security measures are taken into account in the management of the improvement process (in the development of the architecture)
- the company has defined processes for managing investments in the event of changes to investment objectives.

As a supplement (to determine the strengths and weaknesses of the organisation and to refer this evaluation to CMMI reference models – with the application of the best practices in CMMI) the SCAMPI method (*The Standard CMMI Appraisal*

Method for Process Improvement) is suggested. It is useful in an organisation for the development of various kinds of protocols (internal and external) and in the improvement of the architecture development processes (proper and support ones). According to the SCAMPI method, it is necessary to prepare documents describing the requirements, processes and best practices related to each of the processes [55]. The question arises whether the processes and best practices of SCAMPI (or other methods) can be used in the IPP similarly to the manufacturing and management processes of the organisation.

2.5.3 Methodology of Evaluating Client Maturity in the IPP Model

The type of IPP procedures discussed here are used to evaluate the maturity of the client organisation. For that purpose – as was the case with provider organisations – the CMMI model can be used, taking into account the specificity of the client. The *client organisation* or the *representative of the client* can also be a client, as is the case in most project management methods (such as SCRUM, RUP, *etc*.). Then the client (organisation, team, representative) is evaluated from the perspective of process maturity or the suitability and matching of its components [30]. A *suitable* client is classified, on the basis of the theory of team roles from TOGAF [65], as one whose experience and qualifications (in knowledge of the domain and the ability to formalise it) demonstrate the usefulness (or lack thereof) of cooperating with the IT systems provider. And *matching* client personality, classified on the basis of the theory of team roles, means the usefulness (or lack thereof) of carrying out IT projects along with the supplier.

Assuming that an essential component of IT systems development strategy is to set up the team for a given company, it is also assumed that its members will be appointed roles in accordance with the accepted criteria for suitability and matching. The costs of establishing and running the team are mostly offset by savings that the company receives as a result of the proper (appropriate and matched) selection of members to build a particular team. The team may be local, regional or global, and its character is related to the scope of approved activities. Large teams are composed of representatives on at least two levels: (1) the local one, consisting of representatives (experts) of the client, responsible for business processes, and (2) the global one, consisting of representatives of the client who are responsible for the company. The nature of work in the manufacturing of IT systems at a regional level includes selected features of the client's representatives at the local and global levels. In developing IT systems together with the provider, the client team (or the entire client organisation) participates in:

(i) manufacturing processes, *i.e.*:

- development, providing consistency of the architecture and its components
- identification of components for re-use
- taking into account the changing needs of the business

- aiming at solutions for the implementation of new technologies
- making sure that the architecture development is performed in accordance with the standards (Continuum)
- improving the maturity of architectures in the organisation
- providing conditions for making decisions about changes in the project;

(ii) managing processes (to check the compliance of the project with the company's goals), *i.e.*:

- monitoring and control of orders for enterprise architecture development
- ensuring the effective and consistent implementation of the architecture, according to the implementation processes of industry architectures (contained in the repository)
- resolving reported ambiguities, problems and conflicts
- determining functions, consistent with the strategy of the company and the applied IT.

Changes, which the client team may be subject to during the preparation of the IT system, might be caused by:

- introduction of a new CIO to the provider team, who changes the structure of the team
- a merger or acquisition within/outside the company, if it affects the company's functions
- the use of new IT by the provider, which is difficult for the client team to accept
- the provider proving that the client team does not see the business requirements clearly
- the provider establishing that the client team seeks to achieve competitive advantage through the use of new technologies (which can interfere with/hinder cooperation between the client and provider teams, as well as the obtaining of requirements, *etc*.)
- the provider preparing changes in the existing IT system (difficult to accept by the client)
- significant changes in the client's business activity or its rapid growth
- the client having new assumptions about complex or cross-sectional functional requisites.

The team, structured in this way, realises the project strongly supported by the CIO (on the provider's side) and other key members of the team. A better solution is to get corporate support. It is also important to win the support of the supervisory board (for the sake of its major impact on the vision and business objectives). Its members can help in evaluating the adjustment of the IT strategy to business objectives. With such support, the team constitutes a definite executive power at the group level: it is responsible for reviewing and maintaining the company's strategic architectures and their components. It should be noted that the corporate sponsor should also provide support for such a project (ideally, by becoming a member of the team), especially in the implementation phase of the company's architecture. Such a sponsor should declare its partaking as early as in the planning stage. Unfortunately, many companies do not apply such proceedings.

The recommended size of the client team is four to five permanent members and four to five temporary ones. To keep a reasonable team size and a good representation of the company, membership of the team could occur in turns. Note, however, that too much variation in the team can contribute to changes in the vision and implementation of the architecture.

A good solution is to adopt appropriate structures and rules of team member rotation. For example, after an initial effort to manufacture the system, the sponsor's executive body can evaluate the effectiveness of the system. Within the adopted structures, the roles and responsibilities assigned to them and the relationships between them are defined. The structures are built taking into account the best practices adopted to the forms and structures existing in the IT industry. The size of the organisation, its form, and the functions performed by IT are taken into account here. This is a good basis for the construction of the team, as well as for the management and development of architectures.

The structure of the client team (part of the client organisation) should be bound to adequate manufacturing and project management processes, including:

- team management at the global and local levels
- persons responsible for changing requirements
- working groups
- previous experience with the functioning of a team,

as well as its objectives:

- support for the manufacturing processes and their quality
- establishing a platform for mutual formal communication between team members (acceptance of actions on the basis of validated documents)
- provision of control mechanisms (for the effective implementation of the system)
- creating and maintaining appropriate relationships between business objectives and the objectives of the ordered IT system
- identification of discrepancies between system requirements prepared by the client organisation and their implementation by the provider organisation.

The characteristics of the client team presented above include a description of its structure and functioning. In this description we emphasised the processes (in which the client team is involved) of enterprise architecture development and the issue of managing these processes. These processes can be easily compared to the relevant processes of the provider organisation and used in evaluating the maturity of the client organisation. It should be noted that in terms of IT knowledge, the organisations of the client and the provider differ greatly. The client usually delegates its representative or a larger team to work with the provider. If a single representative is delegated, the maturity of the client organisation may have a weak influence on the conduct of the representative. However, if a team of representatives is delegated (as is often the case), the state of the maturity of the client organisation can have a significant and multiple impact on team maturity. In any event, the client organisation should be assessed differently to the provider one. It is suggested, therefore, that a different approach should be used in the IPP than was the case in evaluating the maturity of the provider organisation. A good approach, which takes into account the evaluation of client maturity and the state of one's own organisation/company, and which also takes into account the impact of 'a company, its processes and maturity level', is the COBIT standard. Its usefulness, including the manufacturing and management processes present in the client organisation, will be analysed further within the IPP processes.

2.5.4 Static Processing in the IPP Model

The Sections 2.5.1-3 show the elementary processes (of development and organisation), which were found to be useful in assessing the value of input variables in the MITM model. On this basis, a triple-channel (C-S-P) and three-level structure appeared, shown in Figure 2.7, which is a *static* part of the IPP model, for the incremental updates of the maturity capsule.

Level one allows the *identification* of processes in key areas (Figure 2.7) by assigning appropriate variables to them and by evaluating the processes in the client organisation according to the assumptions described in Section 2.5.3.

Level two performs the *analysis* of measurable variables selected for the maturity capsule C-S-P on level one. In the case of the client and provider organisations, the analysis refers only to processes identified on level one. In the case of negentropy, there is a conversion of (the level of) knowledge of the tools and domain into technology, processes and best practices. A division of key processes into two areas: production and management is also introduced.

Level three *prepares* statically compiled fundamental variables of the model of information technology management MITM in an incremental form.

The incremental variables obtained in the static part of the IPP in channels, form the basis for *dynamic processes* of the IPP, which are used to determine the absolute value of the level (quality) of management defined as the maturity capsule. The formal incremental description of variables and the mechanism of state variables, presented below, are used for this purpose. The increments are evaluated on a scale of 0:5, *i.e.* we continue the modelling based on the maturity scale (see maturity capsule: fuzzy, linguistic and sharp variables).

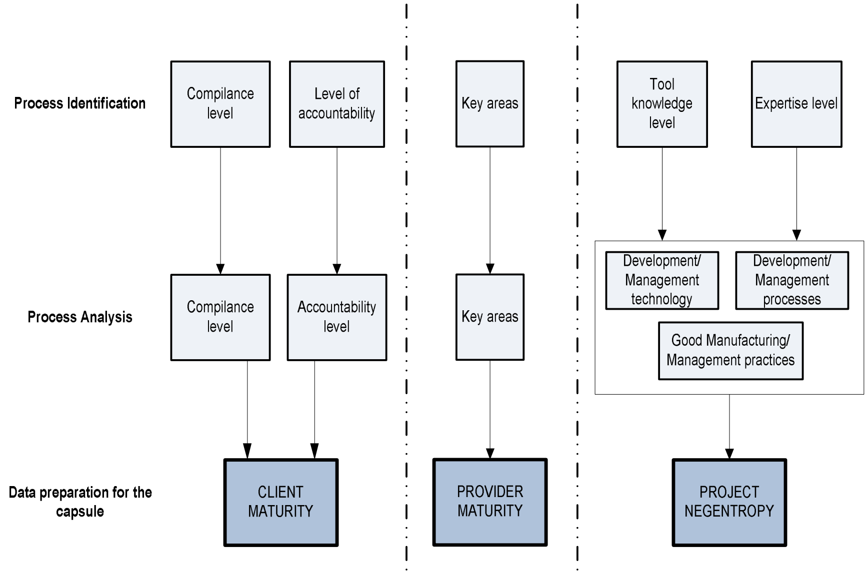


Fig. 2.7 Aggregation of static processes in the IPP model

2.5.5 Dynamic Implementation of the Maturity Capsule in the MITM Model

Below we present a general four-phase diagram of the MITM processes for determining the state of the project via the maturity capsule, which contains the fundamental variables of the ITM process. The final result of the state of the capsule will be the forecast on changes in IT services, *i.e.* in the functionality of the technology (methods and IT tools) of project management.

The parallel channels (C-S-P) in Figure 2.7 are integrated to the form of a static IPP system, which in Figure 2.8 constitutes the input for the dynamic submodel (maturity capsule). After the capsule, there is a decision-making submodel, which is a rule-based-linguistic implementation of the MSM model which converts its variables into the above-mentioned categories of IT services.

Step one initial processing IPP applies to static data (Sections 5.3.1-3).

(**s**) Channel S includes, adequately to (2.4), the aggregation procedures of the increased level of maturity of the supplier/provider organisation with the scalar variable (Δs_t), on the basis of the increased maturity of its key processes: in manufacturing ($\Delta ka1_t$) and in management ($\Delta ka2_t$).

(**p**) The increase in scalar project negentropy is defined similarly as (Δp_t), aggregating variables (ΔhT_t) and (Δhe_t), corresponding to the growth of applied management IT technologies and the level of expertise (the knowledge/ understanding of the project area). We assume that the increase in the level of the applied technology is due to the change or the introduction of new features in the IT used in the project. We also assume that the increase in the level of domain knowledge determines changes in the level of knowledge which is required in dealing with the provider in the creation of the *system metaphor* (a common image of the system seen by the provider and the client). This increase can be identified by definitions of new use cases, with the *system metaphor* in the form of diagrams of these cases. Both the increase in the level of domain knowledge and in IT is observed with the use of the IPP – for the (scalar) evaluation of project negentropy.

(**c**) The increase in client maturity (Δc_t) is associated with an increase in the level of matching (Δhcm_t) and suitability (Δhcc_t) of the client. The increase in client matching means the formation of new personal traits in the software development processes performed together with the provider. Whereas, the growth in client suitability refers to the increasing level of the client's knowledge of the domain (as a result of working with the provider on the system metaphor). These increments are determined according to the assumptions made in the chapter describing the IPP processes in terms of client organisation maturity.

Step two shows the direct effect of static channels, with the use of the management level increase function P_ϕ, on the maturity capsule in the form of a change (Δz_t) in the level of project management. This level is described by the vector $\mathbf{z}_t$, which represents the aggregated, three-element state of the capsule – adequately to its three coordinates. The input growth determines the changes in project management which are the consequence of the growth in the level of client maturity, provider organisation maturity and the (secondary) negentropy of the project.

Step three of processing constitutes the *maturity capsule* (a dynamic system), which reflects the maturity growth processes of the supplier (Δs_t) and the client (Δc_t) organisations, and the scalar project negentropy (Δp_t), represented by the increment (Δz_t), and the new state – the level of project management ($\mathbf{z}_t$). This process is realised with the help of function P_ω, which dynamically refers to the current increment $(\Delta \mathbf{z}_t)$ to the previous level of project management $(\mathbf{z}_{t-1})$, which expresses the quality of management in the previous calculation cycle.

The new level of project management is, therefore, a consequence of the increased maturity of the provider and client organisations, as well as of the project negentropy within the maturity capsule. The scalar measurement of management quality (z_t), resulting from normalisation to the interval <0, 5> (see Appendix 4), which is standard for this work, can be used for the general tool evaluation of the maturity capsule as well as the level of management. The target result of management is to increase the functionality of the tools used in project management.

Step four represents the *decision-making system* based on a rule-based description generating output variables, which are determined using the generally described function P_ψ [13-16]. These variables express the level of expected functionalities, specifically in terms of their growth Δy_t.

Changes in the functionality of the tools thus relate directly to the maturity capsule (C-S-P) and the generic change in the management of the provider team, changes in the applied IT and process changes of the client organisation.

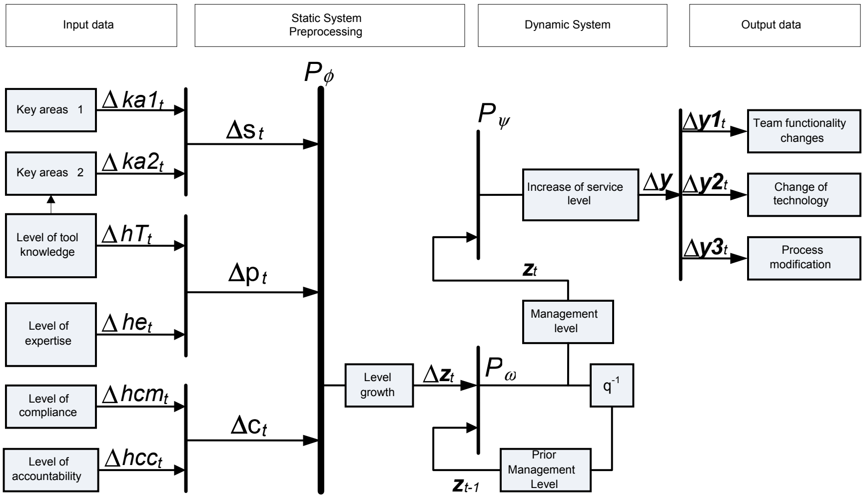


Fig. 8 The integrated dynamic MITM model (IPP - capsule - a decision-making system)

2.5.6 Matrix-Vector Implementation of the C-S-P Maturity Capsule in the MITM

First we will present a simplified general model of the initial processing procedures, which is an extension of the model from Figure 2.8 in a decision-making, fuzzy rule-based mechanism. Then we will show its detailed description which is based on properly defined variables and procedures and their processing [29, 38, 39]. Finally, we will present exemplary IT management models adapted for enterprise architecture development – with the use of vector descriptions of the IPP and the maturity capsule.

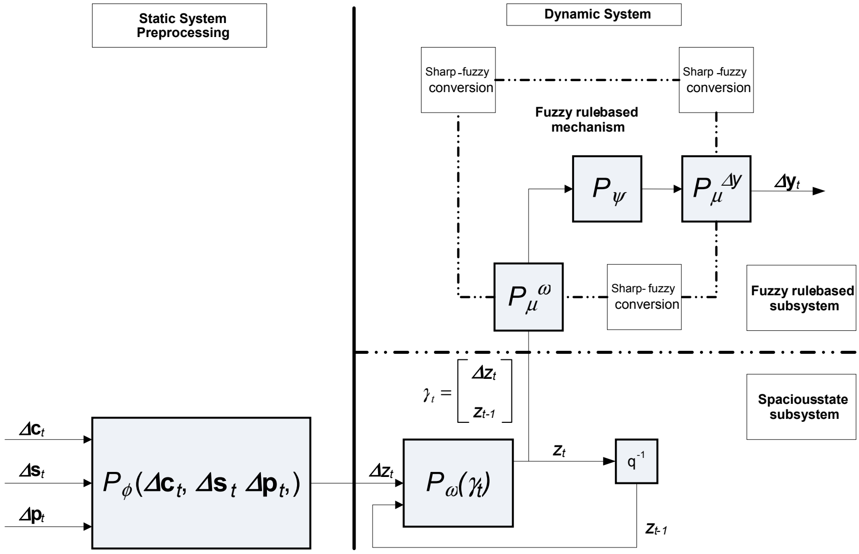


Fig. 9 Model MITM for the selection of IT technologies through a decision-making system

The general initial processing procedures (IPP) model is based on previous findings (see 2.5.4) in terms of processing. This model also includes a proposed IPP specification focused, in the case of project negentropy, on manufacturing and management processes (details are presented in Section 2.5.1). Here we suggest the division of project processes resulting from experience based on the implementation of many projects.

Below we describe how to measure the scalar negentropy of the project and how the processes of management and manufacturing intertwine. Figure 2.9 shows this division and common processes carried out in both areas, with best practices, processes and technologies.

The general IPP model presented here is the starting point for the presentation of the complete IPP model. This model contains areas of static (incremental), dynamic and decision-making (rule-based) processing. The scope of static and dynamic processing was discussed in 2.5.4 and 2.5.5. Whereas, the rule-based processing mechanisms will be discussed in detail later in this work, in Section 2.5.7.

Within the static part, the initial processing is connected to the previously introduced variables (Δc_t), (Δs_t), (Δp_t). The characteristics of the complete conversion process P_ϕ will be shown below on the basis of separate (independent) member functions P_ϕ^c, P_ϕ^s, P_ϕ^p:

$$P_\phi^c : \Delta \mathbf{c}_t \rightarrow \Delta z_t^c \qquad \text{and} \qquad P_\phi^c : \Delta z_t^c = \mathbf{\Pi}_c \cdot \Delta \mathbf{c}_t \tag{2.16}$$

$$P_\phi^s : \Delta \mathbf{s}_t \rightarrow \Delta z_t^s \qquad \text{and} \qquad P_\phi^s : \Delta z_t^s = \mathbf{\Pi}_s \cdot \Delta \mathbf{s}_t \tag{2.17}$$

$$P_\phi^p : \Delta \mathbf{p}_t \rightarrow \Delta z_t^p \qquad \text{and} \qquad P_\phi^p : \Delta z_t^p = \mathbf{\Pi}_p \cdot \Delta \mathbf{p}_t \tag{2.18}$$

$\Delta \mathbf{c}_t$ – client maturity growth vector

$\Delta \mathbf{s}_t$ – provider organisation maturity growth vector

$\Delta \mathbf{p}_t$ – vector of variable increment 'secondary project negentropy'

Δz_t^c – partial growth in the level of management resulting from client maturity growth

z – partial growth in the level of management caused by supplier maturity growth

Δz_t^p – partial growth in the level of management connected with project negentropy growth

t – project realisation time $t \in <1, t_{end}>$, where t_{end} refers to project completion time

$\mathbf{\Pi}_i$ – matrix (row) of partial methodological transformation, for $i \in \{c,s,p\}$, *i.e.* the impact of maturity growth of the capsule C-S-P on the growth in the level of management.

Determining the matrix coefficients of partial methodological transformation allows for the specification of the differentiated impact of the growth of client and provider maturity and the project negentropy growth on the increase in the level of management (details hereinafter).

Knowing the partial effect of each variable (2.16) - (2.18), the complete function P_ϕ can be defined, realising the transformation process of vector increments in the maturity capsule (client-supplier-project) into an increase in the level of project management:

$$P_\phi : \Delta \mathbf{z}_t = \left[\mathbf{\Pi}_c \Delta \mathbf{c}_t \qquad \mathbf{\Pi}_s \Delta \mathbf{s}_t \qquad \mathbf{\Pi}_p \Delta \mathbf{p}_t\right]^T \tag{2.19}$$

$\Delta \mathbf{z}_t$ – vector of management level growth (comprising such elements as Δz_t^c, Δz_t^s and Δz_t^p)

Δz_t – effective scalar evaluation of management level growth, $\Delta z_t \in <0,5>$, and

T – refers to matrix transposition.

In this model, it was assumed that the vector of the maturity growth $\Delta\boldsymbol{s}_t$, of the provider organisation is described with the help of the elements $\Delta ka1_t$ and $\Delta ka2_t$:

$$\Delta\mathbf{s}_t = \begin{bmatrix} \Delta ka1_t \\ \Delta ka2_t \end{bmatrix} \tag{2.20}$$

$\Delta ka1_t$ – variable of the growth in processes located in the key area 1, $\Delta ka1_t \in <0,5>$

$\Delta ka2_t$ – variable of the growth in processes located in the key area 2, $\Delta ka2_t \in <0,5>$

t – project realisation time $t \in <1, t_{end}>$, with project completion time t_{end}, while

Δs – will represent the scalar evaluation of provider maturity growth, $\Delta s_t \in <0,5>$.

The vector $\Delta\boldsymbol{c}_t$ of client maturity growth is described in a similar way, with the use of the element Δhcm_t (growth in matching) as well as Δhcc_t (growth in the level of suitability):

$$\Delta\mathbf{c}_t = \begin{bmatrix} \Delta hcm_t \\ \Delta hcc_t \end{bmatrix} \tag{2.21}$$

Δhcm_t – variable of growth in (the level of) client matching, $\Delta hcm_t \in <0,5>$

Δhcc_t – variable of growth in client competence suitability, $\Delta hcc_t \in <0,5>$, while

Δc_t – will refer to the scalar evaluation of client organisation maturity growth, $\Delta c_t \in <0,5>$.

The vector $\Delta\mathbf{p}_t$ of growth in secondary project negentropy will be described with the use of growth in the level of the IT technology (ΔhT_t) and the knowledge of the domain (Δhe_t):

$$\Delta\mathbf{p}_t = \begin{bmatrix} \Delta hT_t \\ \Delta he_t \end{bmatrix} \tag{2.22}$$

ΔhT_t – variable of growth in the level of the applied IT technologies, $\Delta hT_t \in <0,5>$

Δhe_t – variable of growth in the level of domain knowledge, $\Delta he_t \in <0,5>$, while

Δp_t – will offer a scalar evaluation of growth in secondary project negentropy, $\Delta p_t \in < 0{,}5 >$.

Thus, according to P_ϕ (2.19), growths in the client maturity (Δc_t) and provider maturity (Δs_t) as well as in the project negentropy (Δp_t) can be mapped onto the growth in the level of project management ($\Delta \mathbf{z}_t$ i Δz_t):

$$\begin{bmatrix} \Delta c_t \\ \Delta s_t \\ \Delta p_t \end{bmatrix} \rightarrow \Delta \mathbf{z}_t \text{ or } \begin{bmatrix} \Delta \mathbf{c}_t \\ \Delta \mathbf{s}_t \\ \Delta \mathbf{p}_t \end{bmatrix} \rightarrow \Delta \mathbf{z}_t \tag{2.23}$$

The above model requires matrix coefficients/rows $\mathbf{\Pi}_i$ of partial transformations [13-16, 24], for $i \in \{c,s,p\}$, to be established. Each of them contains different management coefficients α_j, β_j, γ_j, $j = 1{,}2$. An *aggregate methodological transformation matrix* $\mathbf{\Pi} = \begin{bmatrix}\mathbf{\Pi}_c^T & \mathbf{\Pi}_s^T & \mathbf{\Pi}_p^T\end{bmatrix}^T$, encompasses the full impact of the capsule elements (the maturity of the client and provider organisations and the project negentropy) onto the management level evaluation:

$$\mathbf{\Pi} = \begin{bmatrix} \mathbf{\Pi}_c \\ \mathbf{\Pi}_s \\ \mathbf{\Pi}_p \end{bmatrix} = \begin{bmatrix} \alpha_1 & \alpha_2 \\ \beta_1 & \beta_2 \\ \gamma_1 & \gamma_2 \end{bmatrix} \tag{2.24}$$

α_i – matrix coefficient of client organisation maturity transformation, $\alpha_i \in \mathcal{R}_{[0,1]}$

β_i – matrix coefficient of provider organisation maturity transformation, $\beta_i \in \mathcal{R}_{[0,1]}$

γ_i – matrix coefficient of project negentropy maturity transformation, $\gamma_i \in \mathcal{R}_{[0,1]}$, while

$\mathcal{R}_{[0,1]}$ – a set of real numbers from the closed interval <0, 1>.

Rows $\mathbf{\Pi}_i$ of this matrix reflect the impact of the maturity capsule elements on the level of information technology management (ITM). The values of these matrix coefficients need to be tuned according to the knowledge of the dynamic incremental effect of client, provider and project negentropy maturity (2.16) - (2.19) and (2.23) - (2.24) on the level $\Delta \mathbf{z}_t$ of management.

First, we will describe the method of evaluating the coefficient γ_1 in reference to the level of project negentropy (management of component architecture development, based on the model).

Component architecture development is based on the organisation of the manufacturing cycle, which allows the re-use of the source code. It is assumed that the components should be independent of the implementation platform and should enable cooperation between various applications. This allows the creation of multi-level and multi-layer applications. Such components increase the manufacturing team efficiency and the scalability of the manufactured systems. The component approach is especially valid when building a transparent and standardised structure in distributed information technology systems. This facilitates the management of IT systems development.

The value of the coefficient γ_1 is then high (close to 1) and can be adjusted to the applied component standard:

- CORBA (*Common Object Request Broker Architecture*) – developed and supported by the OMG consortium (*Object Management Group*)
- COM (*Component Object Model*) – developed and supported by Microsoft and
- EJB (*Enterprise Java Beans*) – developed and supported by Sun Microsystems.

CORBA architecture is a type of IT systems architecture which provides communication between objects functioning in various IT environments, such as any hardware platforms and operating systems, as well as components implemented in any programming language. Whereas, COM architecture defines the environment for creating communication interfaces for the components of IT systems along with a description of their development method. EJB architecture, on the other hand, defines the standard of Java applications based on multi-layer application components and communication interfaces manufactured with the use of the EJB application server. Selection of the value of the coefficient γ_1 depends on the experience of the team leader in the scope of the applied standard in the use of components.

The coefficient γ_1 depends on the architecture development method. If we apply a method based on the multiple-use of components and assume the existence of a large collection of software components (*e.g.* COTS, *Commercial off-the-shelf*) and their integration within the developed structures, then the coefficient γ_1 should be given a value of 1. In the process of architecture development, the initial stages are similar (the definition and the specification of system requirements), while the intermediate stages are very different. The component analysis is carried out on the basis of the evaluation of the specification of requirements and the possibility of applications. If there are not any suitable components, a trial modification of requirements should be carried out, in terms of the COTS components application, which is an important resource for organisations manufacturing IT systems.

Another way is to develop a system based on the features of available components and to complement the system structures with fragments of software (unavailable in the form of components, yet necessary for its operation). Then the coefficient γ_1 should take the value of 0.5. The manufacturing can also take place by attaching the available components to the finished system. The way of the IT system manufacturing obviously depends on the functionality of the available components and the system requirements. When developing large systems, a hybrid approach is applied (taking into account both the components and the developed fragments of software). The value of coefficient γ_1 should be fixed at the level of 0.75. The hybrid approach should therefore be based on classic cycles of IT systems development: incremental, incremental -iterative and spiral. In each of the mentioned approaches the following steps are performed:

- the architecture project, which aims to identify and document the components of the system and establish relationships between them
- system specification, within which an abstract specification of requirements and the functionality of individual components is prepared
- development of interfaces, which includes their structure and communication methods between them
- development of components, which defines the functionality of the components.

For developing architecture based on the *Model Driven Architecture* (MDA), the coefficients γ_1 and γ_2 should be lower. Managing the development of such an architecture is in fact much more difficult than the management of component architectures. MDA is a type of architecture in which the development-manufacturing team is focused on creating a system model, not on its implementation [37].

A PIM (*Platform Independent Model*) is then created to facilitate this – regardless of the existing technologies. It also supports the later manufacturing and development of models. Its main objective is to separate the layer of the system model from its implementation (technology). The application of the model is of great importance in the development of enterprise architectures and their implementation through SOA (as it provides the matching of an architecture to the needs of the company).

In the instance of developing MDA architectures in the CASE (*Computer Aided Software Engineering*) environment, the value of the coefficient γ_2 should be increased to 1 – which involves the use of *round-trip* engineering in such an environment, consisting of the use of *forward-engineering* and *reverse-engineering* mechanisms in the manufacturing processes. The forward-engineering mechanism is based on the functions of such an environment type as CASE, in which a fragment of the program code is created automatically on the basis of the developed system model. The changes made to the model can be implemented in the code. The reverse-engineering mechanism involves shaping the functionality of the system on the basis of the program code entered. The changes in the code should be reflected in the system model. The forward-engineering and the reverse-engineering mechanisms enable the realisation of round-trip engineering (and therefore the value of the coefficient γ_2 is high).

The values of the coefficients of the matrix of transformation of client maturity (α_1 and α_2) and provider maturity (β_1 and β_2) to the management level, are estimated similarly. These coefficients can be applied in the evaluation of responses to questions about the control objectives of the client and provider organisations.

Table 2.1 Coefficients α_1, β_1 to evaluate control objectives of the client and provider

Value of coefficient α_1, β_1	Linguistic evaluation
1	of key importance
0,5	very important
0,3	significant
0,15	negligible impact

Coefficients α_1 and β_1 can be used for the classification of the importance of questions which refer to control objectives for a given domain (see Appendices 2 and 3 for details). It has been assumed that the maximum value of these coefficients is 1. Other useful values of coefficients, based on the authors' experience, are explained in Table 2.1.

With respect to the coefficients α_2 and β_2 a similar evaluation was used as in the case of α_1 and β_1. These coefficients determine the degrees of certainty for evaluating the realisation of the control objective concerning the level of the organisation processes of the client and provider (see Appendices 2 and 3 for details). The values of the coefficients α_2 and β_2 used to evaluate the degree of certainty of the realisation of control objectives are presented in Table 2.2, as compiled based on the authors' own research.

The values of coefficients and their linguistic evaluation, presented in Table 2.2, have been developed by experts evaluating the organisations of the client and provider. From an external point of view, the linguistic evaluations ‘almost certain’ and ‘certain’ seem identical, while from the point of view of experts, they are different. Therefore, when analysing the effect of the degree of realisation of the control objectives on the maturity of the client and provider organisations, both a rough scale (0.1 to 1) is applied, as well as a linguistic one.

Table 2.2 Coefficients α_2, β_2 for the evaluation of the control objectives of the organisations

Coefficient value α_2, β_2	Linguistic evaluation
1	certain
0,5	almost certain
0,2	uncertain
0,1	doubtful

Having presented the aspect of selecting the values for the transformation matrix coefficients, we can now proceed to the presentation of the dynamic MITM model. It contains two systems: a static system and a fuzzy-linguistic system. These systems, in their initial structures, as presented in [11, 13-16, 21], were used to evaluate IT projects. Both systems are described with the use of two processing functions: the first P_ω used to evaluate levels of project management, and the other P_ψ used in a rule-based mechanism. The first case shows the vector-matrix description of the prior level and the new level of project management. In the second case, a linguistic description of the input and output variables was applied [11, 13-16, 30-32].

Within the dynamic system, first we will describe the static sub-system, whose task is to convert the variables using the function P_ω. It describes the dynamics of the transition (changes) in management levels, which were then used in the rule-based description (Figure 2.9):

$$P_\omega : (\Delta z_t, z_{t-1}) \rightarrow z_t \tag{2.25}$$

$\mathbf{z}_{t-1}$ is the vector of the project management level defined in the following way:

$$\mathbf{z}_{t-1} = \begin{bmatrix} z_{t-1}^{c} \\ z_{t-1}^{s} \\ z_{t-1}^{p} \end{bmatrix} \tag{2.26}$$

$\mathbf{z}_t$ is the vector of the new) level of project management, presented as:

$$\mathbf{z}_{t} = \begin{bmatrix} z_{t}^{c} \\ z_{t}^{s} \\ z_{t}^{p} \end{bmatrix} \tag{2.27}$$

While vector $\Delta\mathbf{z}_t$ is the vector of the growth in the level of project management.

Consequently, after the presentation of the static system, we will show the variables of the fuzzy rule-based sub-system. This is the basis of fuzzy modelling, which is subject to further investigation and will not be discussed in this book.

2.5.7 Linguistic Fuzzy Rule-Based Implementation of the Decisive MSM System

Let us first introduce the linguistic description. The fuzzy rule-based system variables are generated by function P_ψ of management functionality growth. It reflects the rule-based implementation of growth in project management functionality:

$$P_\psi : \mathbf{z}_t \rightarrow \Delta\mathbf{y}_t \tag{2.28}$$

$\mathbf{z}_t$ – vector of the new level of project management

$\Delta\mathbf{y}_t$ – vector of functionality growth which refers to the team of the provider organisation $\Delta y1_t$, to the IT applied $\Delta y2_t$, and to the manufacturing and management processes $\Delta y3_t$

Δy_t – scalar evaluation of growth in project management functionality, $\Delta y_t \in <1,5>$

z_t – scalar evaluation of the new level of project management, $z_t \in <1,5>$

$$\mathbf{\Delta y}_t = \begin{bmatrix} \Delta y1_t \\ \Delta y2_t \\ \Delta y3_t \end{bmatrix} \tag{2.29}$$

As there are no dynamic operations in the described fuzzy rule-based sub-system [20, 23, 25], using an economical description, the time index t can be omitted. Then the basic variables of the described fuzzy-rule-based sub-system (Figure 2.8), the variable increment vectors of the level of management and functionality, can be presented as follows:

$$\mathbf{\Delta z}_t = \begin{bmatrix} \Delta z_t^c \\ \Delta z_t^s \\ \Delta z_t^p \end{bmatrix} = \begin{bmatrix} \Delta z^c \\ \Delta z^s \\ \Delta z^p \end{bmatrix} = \mathbf{\Delta z} \tag{2.30}$$

$$\mathbf{\Delta y}_t = \begin{bmatrix} \Delta y1_t \\ \Delta y2_t \\ \Delta y3_t \end{bmatrix} = \begin{bmatrix} \Delta y1 \\ \Delta y2 \\ \Delta y3 \end{bmatrix} = \mathbf{\Delta y} \tag{2.31}$$

In determining the level of the applied methods and IT tools it has been assumed that the linguistic values are represented by values from the sets {0, 1, 2, 3, 4, 5} or {N, VS, S, M, B, VB}, where N (null, reference value), VS (very small or very low), S (small or low), M (medium), B (big or high or large), VB (very big or very high or very large).

For simplicity, it has been assumed that the linguistic values of the variables of the levels of management (z), and the output data (Δy) are described by the same symbols as rough values.

The linguistic model is based on the concept of processing (Figure 2.8) with the use of the function of increasing functionality P_ψ, which is in the form of a rule:

$$P_\psi : S_z \rightarrow S_{\Delta y} \tag{2.32}$$

S_z is a set of five, the Cartesian product of sets of linguistic values of management levels: $S_z = \delta \times \delta \times \delta \times \delta \times \delta = \delta^5$, where $\delta = \{1, 2, 3, 4, 5\}$, and similarly

$S_{\Delta y}$ describes the Cartesian product of sets of linguistic values of changes in the functionality level, $S_{\Delta y} = \rho \times \rho \times \rho \times \rho \times \rho = \rho^5$, $\rho = \{1, 2, 3, 4, 5\}$.

On the basis of a linguistic description with the use of the management functionality growth function P_ψ, the description can be presented in the form of a simple rule:

$$P_\psi : (z^1 \textbf{ IS } \mathrm{A}_{1j}) \textbf{ AND } (z^2 \textbf{ IS } \mathrm{A}_{2j}) \textbf{ AND } (z^3 \textbf{ IS } \mathrm{A}_{3j}) \Rightarrow (\Delta y^m \textbf{ IS } \mathrm{C}_{mn}) \tag{2.33}$$

where symbols A_{ij}, C_{mn} describe linguistic values which are elements of the matrices

$$\mathbf{A} = [\mathrm{A}_{ij}] \qquad\qquad \mathbf{C} = [\mathrm{C}_{mn}] \tag{2.34}$$

for $i = 1, 2, 3;\ j = 1, 2, 3, 4, 5;\ m = 1, 2, 3;\ n = 1, 2, 3, 4, 5$, and

$$\mathrm{A}_{ij} = a_j = j,\ i = 1, 2, 3, \text{ and } j = 1, 2, 3, 4, 5:$$

$$\mathrm{A} = \begin{bmatrix} \mathrm{A}_{11} & \mathrm{A}_{12} & \mathrm{A}_{13} & \mathrm{A}_{14} & \mathrm{A}_{15} \\ \mathrm{A}_{21} & \mathrm{A}_{22} & \mathrm{A}_{23} & \mathrm{A}_{24} & \mathrm{A}_{25} \\ \mathrm{A}_{31} & \mathrm{A}_{32} & \mathrm{A}_{33} & \mathrm{A}_{34} & \mathrm{A}_{35} \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_2 & a_3 & a_4 & a_5 \\ a_1 & a_2 & a_3 & a_4 & a_5 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \end{bmatrix}$$

and

$$\mathrm{C}_{mn} = c_n = n,\ m = 1, 2, 3, \text{ and } n = 1, 2, 3, 4, 5:$$

$$\mathrm{C} = \begin{bmatrix} \mathrm{C}_{11} & \mathrm{C}_{12} & \mathrm{C}_{13} & \mathrm{C}_{14} & \mathrm{C}_{15} \\ \mathrm{C}_{21} & \mathrm{C}_{22} & \mathrm{C}_{23} & \mathrm{C}_{24} & \mathrm{C}_{25} \\ \mathrm{C}_{31} & \mathrm{C}_{32} & \mathrm{C}_{33} & \mathrm{C}_{34} & \mathrm{C}_{35} \end{bmatrix} = \begin{bmatrix} c_1 & c_2 & c_3 & c_4 & c_5 \\ c_1 & c_2 & c_3 & c_4 & c_5 \\ c_1 & c_2 & c_3 & c_4 & c_5 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \end{bmatrix}$$

The above description on the level of rules closes this short presentation of the MITM model. It may serve as a basis for further development and implementation, for example with the use of the fuzzy description and the development of knowledge bases (*i.e.* for the project negentropy description/variable), as well as for putting the entire issue in the framework of the agent-based system [42, 47], or ontology [5, 44].

The previously presented general initial-processing model is entirely sufficient for the specification of variables in the IPP to determine the growth of maturity in the capsule C-S-P for the model MSM and its implementation as a decision-making system.

2.6 Summary and Conclusions

IT project management involves both manufacturing processes centred around software development, and management processes governing the development processes and the project team. As the keynote presented in the Preface states: management models should also take into account technology management processes (which is a rather rare approach).

A specified organisation is needed in manufacturing as well as in the management of a team and in development processes. Software development is characterised by a properly structured sequential course and iterative (cyclical) realisation. Team management processes require a specific hierarchy. In managing technologies, the procedure of selecting methods is hierarchical, and the selection of tools is sequential (a sampling process). Hence, the technology management model includes both hierarchical and sequential processes (MSM with MSMm and MSMt).

When modelling efforts are focused on managing technologies it is easy to distinguish the key (management) factors, such as the level of the client and that of the provider as well as the status of the project. These factors determine the predestination (success or failure) of the entire project. The concept of the maturity capsule, introduced in this work, integrates these key factors of the project and creates favourable conditions (in the form of an environment which provides suitable input data) to the practical implementation of the IT management model. In this way, the technology management comes down to the dynamic selection of methods and tools which are adequate for the current level of project maturity and for the state of development of the latest technologies.

The proposed way of modelling the process of project management, which integrates the model (capsule) of maturity with the model of selecting technology (exactly, IT management technologies, *i.e.* management methods and tools), is consistent with the current approach in IT of making project models independent from technologies (as technology is clearly a subordinate element).

Therefore, the multi-level sequential/ordered model (MSM) of information technology selection (management methods and tools) – in the context of this book – represents the objective model of information technology management MITM.

This Chapter has presented a multi-level sequential/ordered model of information technology selection. A major motivation for its creation was the rule of inconsistencies by Zadeh [59] referring to the possibility of modelling complex systems. It seems that the presented model is a practical compromise between the accuracy and usefulness of the description of the information technology selection process. This model was developed by moving from a general model (the selection of methods and tools) to a detailed model (including fine points of initial processing). The experience – necessary to establish the final applicability of the model – can be expensive and there will always be some doubt in this regard. Is it appropriate to construct a general technology management model – with variables for which values must be chosen based on heuristic rules and which in practice must be unequivocal (specific)? The authors' opinion can be summed up in two postulates: (1) frequency of use is a relevant criterion for evaluating the quality of the model, (2) the fuzzy modelling philosophy, based on membership functions, is an appropriate foundation for the universality of developed models. Accordingly, the structure of this Chapter consists of four main parts.

In the first part (Sections 2.1 and 2.2), the basic terms used in this Chapter were introduced. This description is for educational and utilitarian purposes (the terms used are applied in MSM models). For readers of the book who are students of computer science, a comprehensive list of terms and statements have been attached which are useful in the construction of the MITM model. Part two was devoted to the construction of the information technology management model MITM (Section 2.3) and to system architectures (Section 2.4).

In accordance with the Zadeh principle (trying to choose the right level of precision), the description of the model was based on diagrams used in control theory (which add clarity).

The Chapter contains a detailed discussion of the multi-level sequential model of information technology selection, which integrates the models of the selection of methods and tools for project management as well as the main features of both models (the selection of methods and tools). First, the model of selecting project management methods was presented (as the method constitutes the first stage of the project management process [23]), and then, on this basis – the tool selection model. The applied description also contains data underlying the basis of the processes (IPP) for setting the three model variables presented in the third chapter of this work.

In this work, we base our discussion on the concept known in physics and cybernetics, where entropy is the measure of the *uncertainty* of information. Negentropy is thus its contradiction and is the measure of the *degree of organisation*: it is therefore the difference between the maximum possible value of entropy, corresponding to the total disorganisation of the system, and its current value. Negentropy decreases as entropy increases, while a growth in the level of organisation (sorting, creation of new linkages, using outside information) corresponds to an increase in negentropy.

The presented results illustrate the complexity of development processes. Evaluations of the maturity variables of client and provider organisations are made by experts. However, the analysis of project negentropy turned out to be much more difficult. Assuming the prerequisite of initial processing (IPP) that during the realisation of the project the changes in negentropy reflect changes in maturity, the question remains of how to measure negentropy?

Engineering practice shows that the framework for such measurements is created by simple boundary conditions (the scale, the smallest and the largest value of negentropy). Therefore, measurements used to evaluate organisation maturity, with a range from 1 to 5 were applied for the evaluation of project negentropy.

Taking into account the issue of the IPP variables, on the basis of standards [41, 65] it was assumed that the following have an impact on the evaluation of project negentropy: the level of fabricating products (the level of documentation and the level of processes of manufacturing, such as the level of development processes) as well as the level of manufacturing management.

The remainder of this Chapter (Section 2.5) was focused on the IPP processes and models. The introduction of a formal description required a specific compendium of knowledge (Sections 2.2-2.4) in terms of analysis and applicability (for the IPP) of data contained in the IT standards. The vector-matrix descriptions (general and linguistic), presented in Sections 2.5.6 and 2.5.7, were based on examples of IT projects in which information technology standards were used, together with the authors' own knowledge and experience. The linguistic model is an attractive approach to project data processing and represents a reasonable compromise (in reference to Zadeh's principle) between its precision and its importance for technology management. Not without significance is the fact that this material presents a report of an important stage of research on the construction of a management model.

The initial processing procedures, IPP, shown in Figure 2.8 (Section 2.5.4) illustrate the basic concept of the ordered approach to the selection of information technology. This arrangement contains: the initial definition of the input variables, forming the information base for the maturity capsule, the initial processing of growth/increments in terms of the capsule, the system of a dynamic increase in the level of information technology management, and the conversion of the target management level into output variables, representing an increase in the three-layer functionality. This diagram highlights the need for continuous monitoring of the state (maturity) of the project and the state of technologies supporting this project. It is also a formal support and a tool for implementing successful IT projects.

This Chapter has fundamental conceptual relevance to the whole work. For it presents formal IPP models, the maturity capsule and the MSM, and the methods of their construction. The following Chapters (3 and 4) will focus on the description and the specifications of these models and on their practical verification.

3

Specification of Variables of the IPP Processes and of the Maturity Capsule

Chapter 2 was devoted to the initial processing procedures and their models. In this Chapter we will present the specification of processing variables for these models. First, we will refer to the negentropy of the project using classical knowledge of enterprise architecture development. Then we will discuss the concept of *provider organisation maturity* and the methods of its evaluation. We will show the applicability of the ITIL standard for the manufacturing organisation, especially in the context of the CMMI model used in the IPP. Finally, we will provide the specification, evaluation and application of the *client organisation maturity* variable.

The data for the specification of variables is discussed in the appendices, making it easier to review and evaluate the applicability of IT standards. The Chapter is concluded with the verification of the proposed method/specification from the point of view of further consideration.

3.1 Specification and Measurements of the Primary Negentropy of a Project

The general model of the IPP is based on the input variables defined in Section 2.3. It takes into account the IPP specification (Section 2.5), which is focused on knowledge of the domain and the technology, and the variables (Section 2.5.1): the width of the repository of architectures, the length of the documentation catalogue and the height of the ITM section. This division, which is the result of years of experience gained in implementing IT projects, takes into account the need for a precise definition of the coordinates of secondary project negentropy, expressed by the formula (2.2): the height of the applied technology of IT project management and the level of expertise (the knowledge of the area). The IPP processes carried out in the negentropy space (the width, the length and the height), and the mapping of this space onto two variables allows the specification of the secondary negentropy. This leads to fixing the scalar tertiary negentropy.

Z. Kowalczuk and C. Orłowski, *Advanced Modeling of Management Processes in Information Technology*, Studies in Computational Intelligence 518,
DOI: 10.1007/978-3-642-40877-9_3, © Springer-Verlag Berlin Heidelberg 2014

The case of new enterprise architecture development, presented below, refers to a group of projects with (initially) low negentropy. It is modelled with the use of the IPP and evaluated according to project negentropy, which – in its tertiary form – allows for an unequivocal evaluation of the comprehensive issue of IT systems architecture development.

The evaluation of the enterprise architecture development process comprises of three variables. In each dimension of the 3-dimensional space of process and product quality (Fig. 3.1): the width of the architecture development repository (a_t), the length of the documentation catalogue (d_t) and the management (pr_t) measured by the height of the ITM section (which must not be confused with the management process as such), an overall evaluation was applied:

$$\bar{\mathbf{p}}_t = \begin{bmatrix} pr_t \\ a_t \\ d_t \end{bmatrix} \tag{3.1}$$

It should be noted that in the evaluation of the above-mentioned *primary negentropy*, we take into account both the proper development processes and the supporting ones.

The first variable (axis in Fig. 3.1) 'the width of the architecture development repository' involves the significance of architectures (and their range/set) in dealing with a client (as in TOGAF). This variable takes the following values: initial (Continuum) and final (ADM). We assume that the width of the architecture development repository includes a set of architectures created in the process of enterprise architecture development.

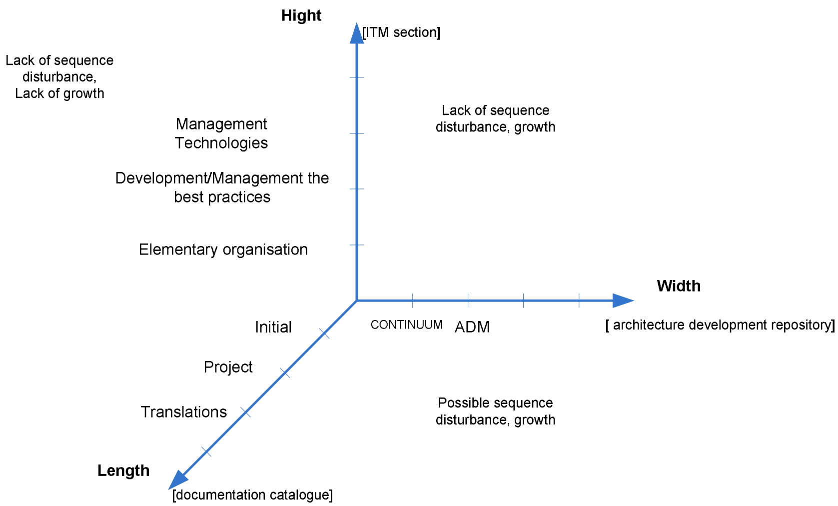


Fig. 3.1 Project negentropy in the three-dimensional space of the quality of development processes and their products: the width of the architecture development repository (a_t), the length of the documentation catalogue (d_t) and the height of the ITM section (pr_t)

The adopted specification, according to the standard (ADM – Contunuum) is typical only when developing architectures with the *bottom-up* approach. When developing architectures with the *top-down* approach (enterprise architecture as a whole), the initial construction of the Continuum, on the basis of which the ADM processes is developed, is a better option.

In this way, the control of the manufacturing processes is supported at the level of decision-makers/strategists, rather than at the operational level. Therefore, the description presented in Fig. 3.1 begins with the creation of the Continuum and its resources (as part of the supporting process), and only then are the architectures, developed within the ADM, introduced.

The second variable (Fig. 3.1) is the length of the documentation catalogue developed for the project, which includes the initial documents, documents of the developed architectures, as well as transfer documents prepared in iterative cycles. The length of the catalogue is simply the number of documents produced during the architecture development process.

The third variable is the height of the ITM section. Values identifying *elementary organisation of development* (manufacturing processes and technologies, and management processes), *best practices*, as well as *IT management technologies* are assigned to this. Figure 3.2 shows the classification of the ITM section of the project and shows the common processes carried out both in areas of manufacturing and management. We assume that in the definition of the ITM section we take into account the levels of the manufacturing and management processes.

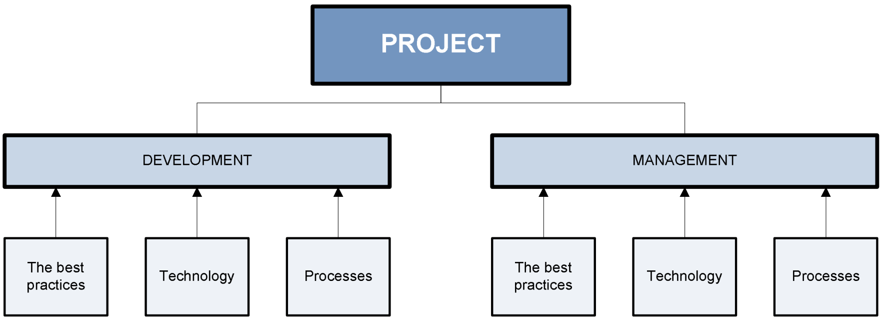


Fig. 3.2 The project: its processes, technologies and best practices applied

The use of information technology constitutes the highest level of support for the production (manufacturing and management) processes, and determines the success of the project. Best practices (of ADM) are an intermediary solution relating to the framework and the principles of architecture development. An example of manufacturing processes (of architecture development using best practices) is shown in Fig. 3.3. They are implemented in three loops, with the use of the Zachman framework and/or the ADM. The Zachman frame is an example (based on best practices) of a tool (open utility) to support manufacturing processes (which was developed by an IBM employee in the form of a sheet and allows the different levels of enterprise architecture to be defined). An example of such best practices in management is the use of the heavy method, when the project management processes are carried out iteratively at these levels:

- the selection of the architecture development methods and solutions
- the analysis of the need for data migration
- EA Governance – management processes to monitor and control EA development processes, in order to reach compliance between the architecture and the TOGAF/ADM standard (such compliance ensures corporate governance in the company), whereas the Zachman framework is a support tool for EA Governance (except the ADM)
- changes (dynamic loop).

The adopted variables (the width of the architecture development repository, the length of the documentation catalogue and the height of the ITM section) will receive a formal description. Because the model of IT management MITM refers to changes in the project, we will analyse these variables in terms of changes (increments) in the width of the architecture development repository, the length of the documentation catalogue and the height of the ITM section.

To define these changes we will first consider the three levels of description (Figs. 3.4-3.5):

- I – artifacts: the width of the architecture repository and the length of the documentation catalogue: expressing the quality of the developed architectures and documentation (in the repository)
- II – the height of the ITM section describing the level/quality of management processes
- III – negentropy aggregating the quality of the IT architecture development processes.

The term: The increment in the length of the documentation catalogue *is a measure of progress in the degree of realisation of an IT project. We assume that in the enterprise architecture development process, the number of documents representing this increment is an important indicator of the level of architecture development management.*

The term: The increment in the width of the architecture repository *is a measure of progress in the degree of realisation of an IT project. We assume that in developing enterprise architecture, the number of the developed architectures, or the increment in the number developed is an important indicator of the level of architecture development management.*

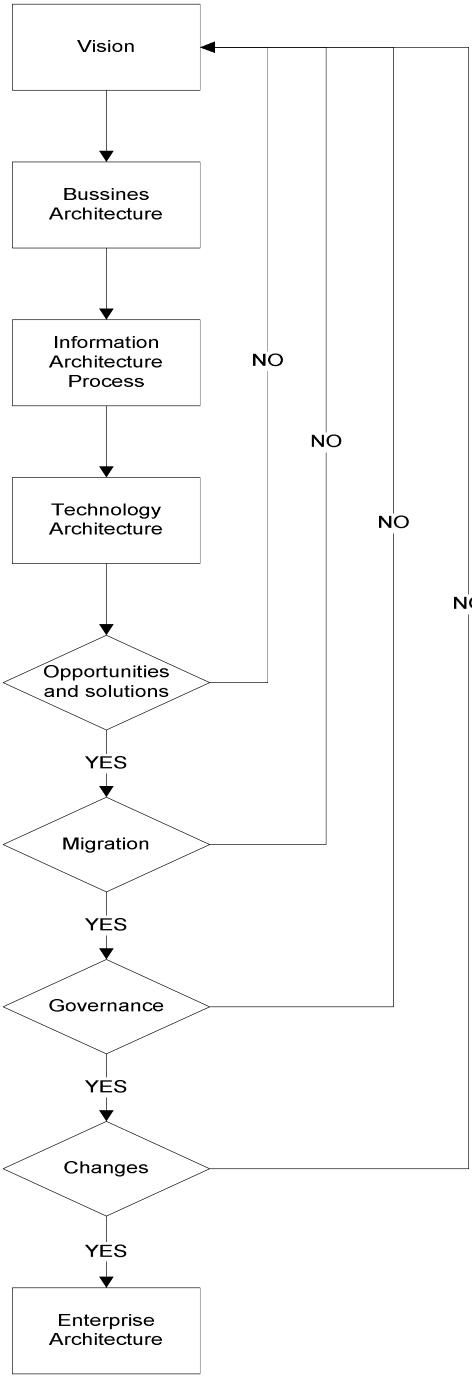


Fig. 3.3 Architecture development processes from the project negentropy perspective

The term: The increment in the height of the ITM section *is a measure of progress in the degree of realisation of an IT project. We assume that the number of manufacturing processes is a good indicator of the level of architecture development management.*

The term: The increase in the level of best practices *is a measure of progress in the degree of realisation of an IT project. Thus we assume that the number of best practices applied is an important indicator of the level of architecture development management.*

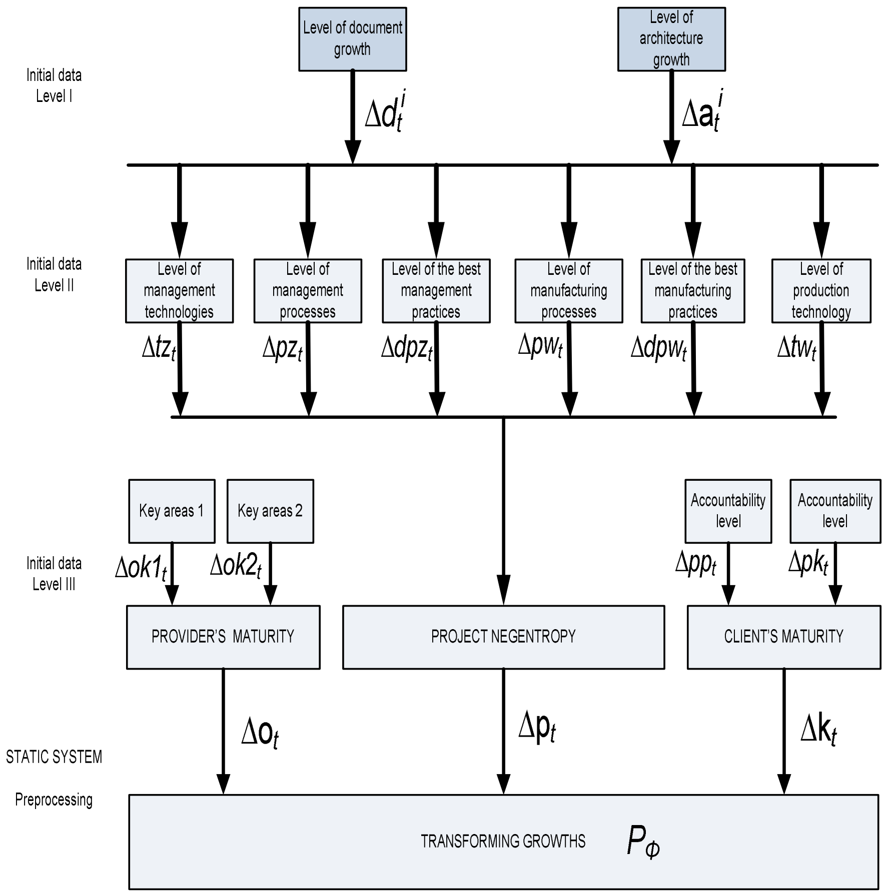


Fig. 3.4 Maturity factors affecting the level of project management

3.2 Source Specification of Project Negentropy

When analysing the issue of project negentropy in terms of IT standards (sub-Section 2.2), it is easy to notice the existence of eight source components of negentropy. In this way, the lowest (fourth) level of negentropy is revealed, which is not deeply analysed here. The discussion below will be narrowed down to provide the source ingredients and their transition to the three-dimensional (primary) negentropy and the two-dimensional (secondary) negentropy.

3.2.1 *Reference to Primary Negentropy*

Let us first consider the six elements describing the manufacturing technologies and project management, represented by the following vector associated with the ITM section:

$$\Delta \mathbf{pr}_t = \begin{bmatrix} \Delta hdT_t \\ \Delta hmT_t \\ \Delta pd_t \\ \Delta pm_t \\ \Delta gpd_t \\ \Delta gpm_t \end{bmatrix} \tag{3.2}$$

$\Delta \mathbf{pr}_t$ – six-dimensional vector of (technological) growth in the ITM section

Δpr_t – scalar evaluation of growth in the height (prt) of the ITM section, $\Delta pr_t \in <1,5>$

ΔhdT_t – variable of growth in development technology, $\Delta hdT_t \in <1,5>$

ΔhmT_t – variable of growth in management technology, $\Delta hmT_t \in <1,5>$

Δpd_t – variable of the increment in development processes, $\Delta pd_t \in <1,5>$

Δpm_t – variable of the increment in management processes, $\Delta pm_t \in <1,5>$

Δgpd_t – variable of the growth of good practices in development, $\Delta gpd_t \in <1,5>$

Δgpm_t – variable of the growth of good practices in management, $\Delta gpm_t \in <1,5>$

t – project realisation time, t ∈ <1, tend>

tend – project completion time.

For the scalar evaluation of Δpr_t the vector of technological growth (3.2), you can easily use the adequate and simple extension of the formulas of scalarisation given in Appendix 4.

Having achieved a scalar evaluation of the growth in management quality (the height of the ITM section), the applicable increment of three-dimensional primary negentropy (3.1) can be expressed as a vector

$$\Delta\bar{\mathbf{pr}}_t = \begin{bmatrix} \Delta pr_t \\ \Delta a_t \\ \Delta d_t \end{bmatrix} \tag{3.3}$$

$\Delta\overline{\mathbf{p}}_t$ – three-dimensional vector of the increment in the primary negentropy of the project

Δpr_t – scalar evaluation of growth in the height of the ITM section (=*hmtt*), $\Delta pr_t \in <1,5>$

Δa_t – variable of the increment in the width of the architecture repository, $\Delta a_t \in <1,5>$

Δd_t – variable of the rise in the length of the documentation catalogue, $\Delta d_t \in <1,5>$.

The vector above should be associated with the progress in the three-dimensional space of project negentropy (Fig. 3.1) described by three axes: the width of the repository, the length of the documentation catalogue, and the height of the ITM section.

3.2.2 Reference to Secondary Negentropy

For reference and transition to secondary negentropy, we suggest describing a detailed forecast of the growth in project artifacts (treated as a form of evaluation of the level of expertise related to the project which is being developed) in one vector form $\mathbf{\Delta ar}_t$ expressing the increments in the length of the documentation catalogue and the width of the architecture repository:

$$\Delta\mathbf{ar}_t = \begin{bmatrix} \Delta a_t \\ \Delta d_t \end{bmatrix} \tag{3.4}$$

$\Delta\mathbf{ar}_t$ – two-dimensional vector of growth in project artifacts (project expertise)

Δar_t – scalar evaluation of growth in artifacts ar_t (of the level of expertise he_t), $\Delta ar_t \in <1,5>$

Δa_t – variable of the increment in the width of the architecture repository, $\Delta a_t \in <1{,}5>$

Δd_t – variable of the increment in the length of the documentation catalogue, $\Delta d_t \in <1{,}5>$.

The vector (3.4) undergoing scalarisation (Appendix 4), leads to a scalar evaluation of the increment in the evaluation of the size of the project artifact Δar_t (Δhe_t), and in combination with knowledge of the height of the ITM section Δpr_t = (ΔhT_t), it allows the establishment of a two-dimensional representation of project negentropy (2.2).

Thus, for the conditions presented above, the vector of the increment in the secondary negentropy of the project (2.2) will be described here with the use of components of artifacts (the width of the repository and the length of the documentation catalogue, expressed in Fig. 3.5, respectively as the 'repository' and 'documentation'), and the height of the ITM section (marked in Fig. 3.5 as 'management'):

$$\Delta \mathbf{p}_t = \begin{bmatrix} \Delta pr_t \\ \Delta ar_t \end{bmatrix} \tag{3.5}$$

$\Delta \mathbf{p}_t$ – two-dimensional vector of the increment in project negentropy

Δp_t – scalar (target/tertiary) evaluation of the increment in project negentropy, $\Delta p_t \in <1{,}5>$

Δpr_t – scalar variable of the increment in the height pr_t of technology (ITM), $\Delta pr_t \in <1{,}5>$

Δar_t – scalar variable of artifact growth ar_t (of the level of expertise he_t), $\Delta ar_t \in <1{,}5>$.

In consequence of this process, based on the source negentropy (3.2), the primary negentropy described in a 3D space is subject to mapping onto the secondary negentropy given in a 2D space. The secondary negentropy is defined by two components: the height of the applied information technology and the level of expertise (knowledge) of the project domain. The applied transition (8D-3D-2D-1D) shows how you can simplify the structure of negentropy. Based on the source negentropy, it is easy to 'make a precise measurement' – establishing the value of one-dimensional (scalar, target) *tertiary negentropy*, as shown in Fig. 3.5.

Having accepted the above assumptions, the further processing of variables and their values can proceed according to the general rules described in Section 2.5.1. These assumptions also help us to map the three variables of the primary negentropy (a_t, d_t, pr_t) onto the two variables of the secondary negentropy (Fig. 3.5) expressed by the formula (2.2) and referred to in the diagram as 'technology' and 'expertise'.

The increments of the *secondary negentropy* vector are, according to the formula (3.5), described in a 2D space, which simplifies the structure of negentropy (2.2), and allows the height of the ITM section to be assigned to the applied information technologies and the number of artifacts (documents of the developed architectures), determined by the length of the documentation catalogue and the width of the project repository, to the current expertise (the knowledge of the project domain), as well as allowing for the transition (3D-2D).

Adopting such rules, we will first define the basic concepts of architecture and we will present the information systems architecture described in the TOGAF standard.

As the concept of project negentropy is a novelty, the three following points are devoted to it, which show ways of creating and selecting the media of negentropy (impact factors).

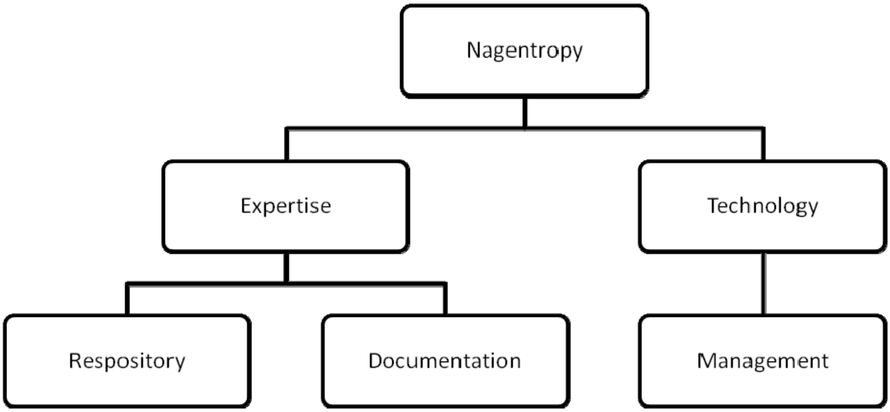


Fig. 3.5 Hierarchical, three-layer structure of negentropy

Section 2.5, devoted to the description of elementary channels (IPP), has presented definitions related to the maturity capsule. This entails the standardisation of the description (including the structure) of individual measures (not only measures should be common). Consequently, referring to the definition of project negentropy given there, below we are going to explain the measures and the impact factors. To enhance the functionality and applicability of the C-S-P capsule in the MITM model, we introduce the standardisation of these factors.

3.3 Compendium of Knowledge for the Specification of Negentropy

A more accurate assessment of the structures of negentropy, introduced above, requires delving into the TOGAF standard resources, as it is the essential source of knowledge for architecture developers and the creators of documentation. The processes described there apply to the development of ADM architecture and the construction of the Continuum. First, within the supporting processes, the Continuum repository is built (including the SIB and the TRM), and then the proper development processes are implemented, *i.e.* the ADM development, relating to the different architectures (vision, business processes, information systems and technology).

Quality is a fundamental concept for this book. In most cases considered here, it is understood in the ordinary *procedural* sense, or in the *structural* sense, which means that we accept the quality of an artifact on the basis of it meeting all the structural requirements. Negentropy, specified in terms of structure and parameters, is such a generic term for this book. Another classic example in developing architecture and information systems is the quality of the *documents*, verified only on the basis of maintaining their structure (*i.e.* the requirements relating to the hierarchical list of elements of a given document). The *documentation catalogue* is also assessed structurally.

Moreover, to make the quality analysis complete and to properly organise the terms used in this work, a clear-cut definition is needed of the very important *quantitative* relationships between the quality of the manufactured artifacts (products) and documents, and their number or the degree or level of their implementation (development):

(1) <u>number</u> refers to the amount of the manufactured items (artifacts or documents) in the (rough) scale of natural numbers (example: Tables 3.1 and 3.2)

(2) <u>degree</u> refers to the same information expressed in relation to an accessible range (maximum) in the (rough) scale of real numbers (usually in percentages) (see: Tables 4.6 and 4.12), while

(3) <u>level</u> represents the degree expressed linguistically:
   a. in a trivalent scale: YES/NO/PARTIALLY (Table 3.2, the quality of the ITM section)
   b. in a pentavalent scale (*e.g.* Tables 3.1 and 3.2, EA evaluation) with reference to null.

The granularity of evaluations (effectively 3 to 6 levels) results from practice.

In the partial evaluation of architectures and documents (in Table 3.1 – the number of created documents), the pentavalent scale is commonly used (beyond the reference level – void/null/ 0%), in which 20% corresponds to ‘very low’, 40% – ‘low’, 50% – ‘medium’, 70% – ‘big’, and 90% – ‘very big’). If those who developed the architecture provided two manufacturing documents out of the required four (Section 2.3), the documentation is at the level of 'medium'.

The ‘null’ (0%) reference level is usually ignored or equated with the lowest level of the scale (i.e. we are talking about the length of a scale in terms of non-zero levels).

In the case of estimating the (applicability level of) management technology, a three-grade evaluation is used temporarily (Table 3.2): (NO) for not applying management technology, (YES) where management technologies are used (both those included in the Continuum and external applications), (PARTIALLY) in the case of applying IT only from the Continuum resources or using solely external applications.

In order to standardise the results (of document and architecture evaluations) a conversion of the trivalent scale is necessary into the ‘full’ hexavalent one. Assuming that NO corresponds to the values of ‘null’ or ‘very low’, and YES to ‘very high’, the state PARTIALLY must be broken up into ‘small’, ‘medium’ or ‘big’, adequately to a separate expertise, which may be based on the importance of the developed documents for the process of architecture development.

The applied mixed (rough-linguistic) description is the implementation of the Zadeh rule of inconsistencies, widely used in this work, which juxtaposes the precision of a description with its importance. This description is efficient (which is significant for those who manage projects) in that it facilitates the classification of the quality of the analysed objects (prepared documents and manufactured products and artifacts) and provides useful evaluations of the state of the project.

3.3.1 Construction Technology of the Continuum

This report refers to extreme evaluations of the variable of the architecture repository width (a_t):

1. the lowest, with the value of 1, corresponding to the development of enterprise architecture without keeping the architecture development sequence and relevant documents
2. the highest, with the value of 5, for proper architecture development – based on the Continuum and with a correct architecture development sequence and relevant documents.

The issue of evaluations will also be discussed at the end of this Chapter.

Architecture of the Continuum

On the basis of the TOGAF description, in terms of the manufacturing and management of enterprise architecture development (ADM) as well as the Continuum, we propose a new concept of the maturity: *negentropy* of architecture development. The aim is to present the (fairly complex) TOGAF standard in a form which is more accessible to the client and the provider. Experience connected with standards (TOGAF, *etc.*) shows that the applicability of such tools increases when a client-friendly linguistic description is created. On the basis of the system of rules which was discussed in Section 2.5.7 we will now propose a conversion of the TOGAF standard into a linguistic description. It is a result of the analysis of the TOGAF standard and own experience in the implementation of IT projects [24, 27, 28, 34, 36]. It is also closely connected to the concept of selecting project management methods, presented in Chapter 2.

The description has been divided with regard to the concept of (a_t, d_t, pr_t): the width of the architecture development repository, the length of the documentation catalogue and the height of the ITM section, presented in Chapter 2. The description reflects the level of architecture development and enables companies to predict the possibilities of project realisation in terms of enterprise architecture development.

After the introduction of the linguistic description of the construction of the Continuum, we will present the development process of enterprise architecture in a similar way. We will draw attention to the need to keep a sequence within project development and management (the application of the ADM process in building the Continuum within the supporting process).

The Main Cycle and the Development Sequence. As companies often undertake single attempts at selecting particular architectures that implement processes, it is necessary to maintain appropriate relationships in the development process of architectures and their models. First of all, the enterprise architecture development process should take into account the existence of a cycle of two basic processes: the supporting one (the construction of the repository-Continuum) and the proper one (ADM). It is recommended, therefore, that the development and implementation of the ADM, the different architectures (vision, business processes, information systems and technology), follow the development of the construction of the project repository (the Continuum). According to the TOGAF standard, the Continuum has two reference models: TRM and SIB, and a detailed model of III-RM. The TRM model is a set of generic architectures, while the SIB is a set of information standards about the architecture development methods within the TRM. The III-RM model, which is a subset of the TRM, has a specific place.

One might ask whether these models should be used at all and in what sequence of the development process? The classical top-down approach to software development suggests that the construction of the Continuum should take into account the creation of the SIB model as well as the definition of standards for generic architecture development within the TRM model.

What results from this is the fundamentality of the *main project cycle* for enterprise architecture development (Fig. 1.1), including the construction of the Continuum (supporting process) and the development (proper process) of the ADM architectures. To complete the analysis, the instance of the unstructured creation of the Continuum should also be considered.

The first sign that the construction of the Continuum has such a character is the lack of data about the enterprise architecture development (no SIB). No initiation of the manufacturing process affects the development process of the ADM and means that the project has low negentropy. It also means that the relationships between the level of SIB manufacturing (supporting process) and the level of the development (in the proper process) of generic architectures, on the basis of the

TRM and III-RM, are not maintained. The TOGAF standard suggests maintaining appropriate relationships between patterns of projects – of generic architectures, common systems, industry architectures and specific enterprise architectures. The failure to maintain logical hierarchical relationships or the lack of taxonomy definitions indicates the small negentropy of such projects.

The Quality of Architecture. The *structural quality* (level) of the Continuum architecture is influenced by the level of the structure of two basic models (SIB and TRM) as well as their components/sub-models (*e.g.* III TRM model), being its resources. The quality of architecture development is thus measured by the quality of the development of all the components (as defined in the TOGAF specification) of the supporting processes, but also by the level of proper architecture development (ADM). It is assumed that the TRM model is complete if it contains not only a complete taxonomic description, but also its graphic version. The III-RM (sub-)model, which refers to the model of the information exchange environment, is treated similarly.

According to TOGAF, a complete description means that, in its resources, a company has a common system architecture, as well as industry and enterprise architectures, based on a generic architecture (SIB and TRM models). Complete descriptions should also take into account the adopted development methods, using products, services and both system and industrial solutions.

A perfect solution for a company is to have (in the Continuum) an industrial solution which meets the company’s business needs. TOGAF also defines the components that form the basis for the construction of the Continuum for the adopted structures, such as models, views, blocks (including those of the domain) and business scenarios to enable the construction and modification of the listed structures. Therefore, when evaluating the quality of architectures, all their elements should be considered. The state characterising the Continuum can be expressed linguistically as: fully complete, complete, partially complete, mostly complete or incomplete. This five-element division is adequate to its fuzzy analysis later.

It should also be determined whether and how the completeness of the Continuum translates into its applicability for an enterprise. The most useful, from the point of view of project negentropy, are dedicated architectures containing a set of methods for their development. It is important to have dedicated architectures, implementation solutions, and system components in the Continuum resources for the EA projects with a high negentropy.

It is much more difficult to identify cases of small and medium negentropy and the adequacy of the Continuum resources for such projects. Therefore, it is worth (as part of the supporting process) evaluating the quality of the repository as a whole, taking a hierarchical approach to estimating the level of its parts: the quality and availability in the Continuum of its two main models (TRM, SIB) – in step 1, then the common systems of companies – in step 2, the industry – in step 3, the enterprise – in step 4, and finally – its entirety – in step 5 (Figs. 2.5 and 2.6).

The implementation of each of these steps requires the availability of an implementation environment which affects the quality of individual products. The quality of the prepared models is also evaluated with the use of expert weights. This means, for example, that if experts evaluate the impact of manufacturing the SIB and the TRM on streamlining the enterprise architecture development process (in terms of its negentropy) in a similar way, the two resources are assigned to the same weight. Figure 3.6 shows the quality of the Continuum resources taken into account.

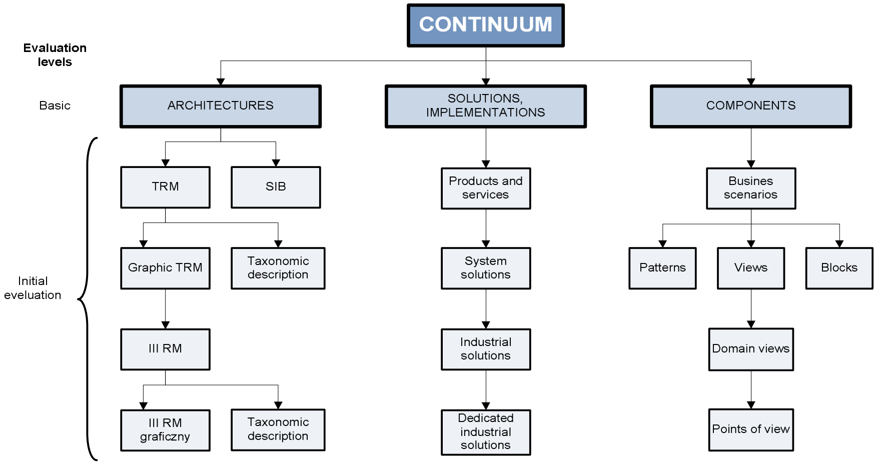


Fig. 3.6 The method of evaluating the quality of architecture development

The method of evaluating the quality of architecture development, presented in Fig. 3.6, includes three main components of the Continuum: architectures, their implementation solutions and components which form the basis for the construction of architectures. The scope of their implementation determines the degree of the adjustment of the Continuum to the needs of the company, which can be tuned by the afore-mentioned weights (determined by experts).

For such an estimation, an initial evaluation of the level of architecture specification is necessary, in terms of the width of the repository (a_t). As before, expert evaluation mechanisms are recommended here, (*i.e.* the percentage, and assignment of weights to both components). The quality of the III-RM model is evaluated alike: taking into account the level of implementation of the TRM model and paying attention to the components of the evaluation. For simplicity, the evaluation presented in Fig. 3.6 is narrowed down to two levels (basic and initial evaluation).

The Continuum Documentation

As was the case with explaining the construction of the Continuum, now we will present the resources which make up the Continuum documentation catalogue (with its length as d_t) prepared as part of the supporting process (Fig. 3.7), taking into account the appropriate quality and the structures of the documentation catalogue (Continuum) and individual documents.

The Structure of the Documentation Catalogue. Prior to determining the sequence of creating documents, their quality and the structure of their catalogue (*i.e.* the structure of the Continuum documentation in general) must be well-defined. The structure (determining the quality of the documentation catalogue) presented in Fig. 3.7 and the sequence from left to right (input documents – development documents – output documents) are obvious, thus, what remains to be determined is the preparation degree of the documentation (the length of the 'catalogue').

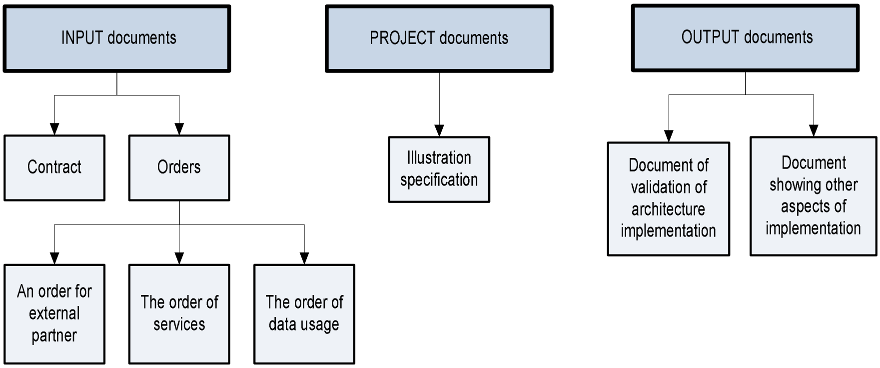


Fig. 3.7 Specification of documents manufactured for the Continuum

The input documents include the contract and the orders placed with external partners during the ADM processes. The participation of the development documents in the project implementation is significant. For example, taking only the development documents into account, their structural specification is presented in Fig. 3.8. The output documents, including the description of the system validation processes are presented to the project partners when the project is completed.

The Degree of Preparation of the Continuum Documents. This quantitative criterion for evaluating documentation closes the quality specification of the Continuum.

The input documents (as well as other materials and project data, important for the completeness of the Continuum) also require qualitative evaluation. A list of tasks which should be included in the ADM development contract (an example of an initial document) can be very helpful. Before describing the method of determining the level of preparation of this document, it is worth reviewing the tasks which should be included in the contract:

- determination of the number and specification of the developed architectures including costs
- preparation of a statement of intent together with the partners
- ensuring the operation of the system and the constant monitoring of the level of architectures
- compliance with the rules of the ADM and the use of appropriate components
- adoption of rules for determining risk in various aspects of development and implementation.

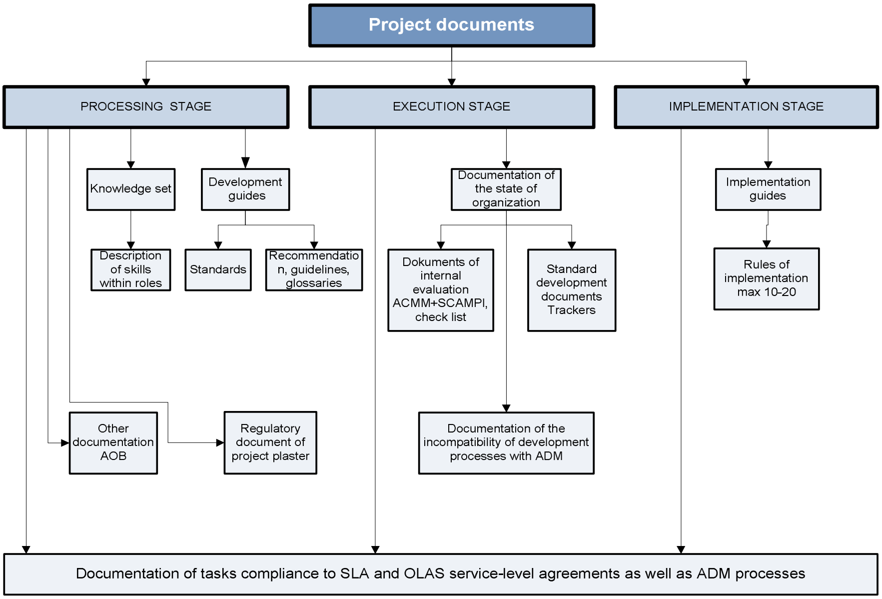


Fig. 3.8 The Continuum development documents

Therefore, full realisation of the initial document requires the preparation of five documents. A smaller number of documents provided means a lower level of documentation development. In such a situation, a detailed hierarchical description of the document development method should be included (see Table 3.1 for details on structural and quantitative evaluation).

The way contracts are prepared by the client should be evaluated similarly (see Section 2.5.3 for details). The preparation of the contract by the client and the provider is of course essential; orders for subsystems (such as the CRM and SCM systems as subsystems of the ERP) placed by the client are optional and depend mainly on the specifications of the contract. Thus, the subject of evaluation should not be the number of orders, but rather the way they are prepared. For example, if the contract assumes the preparation of ten orders (4 for services, 6 for resources), then the structural quality of the preparation of all ten orders should be the subject of evaluation.

When considering the output documents, those of a validating nature should be taken into account, as stated in the TOGAF standard. Neither their number nor type is specified. It would be reasonable to evaluate the need for these documents in relation to the contract, on the basis of best practices. The situation is different with the development documents. Their specification in TOGAF is precise, which means that their meaning is greater. Fig. 3.8 shows the hierarchical system of the developed documents, with three stages and three levels of description: the TOGAF division into stages and the levels complying with the solution proposed in this work.

Documents prepared at the stage of process modelling include: guides for manufacturing, sets of competences for the provider team members, as well as documentation of the compliance of the performed tasks with the ADM processes, contracts and services. These documents are necessary at the development stage and are the basis for initiating architecture development according to the ADM. They facilitate the development of enterprise architecture. The development processes of enterprise architecture are monitored (EA Governance) to check their compliance with TOGAF and its model of development, which is the ADM. The development documents of (EA) architecture are prepared for the benefit of a current review of the project.

If problems with realisation occur, it is worth considering the maturity evaluation of the provider organisation. It may be useful to identify development problems and the quantitative measure of maturity (obtained via SCAMPI) should facilitate the diagnosis. Although these documents are not standard, they are very useful in organisations with a low level of maturity.

The need for developing *compliance documents* should also be noted, as they describe the compliance processes of EA Governance and their status. They are created at the stage of contract development, and their applicability is greatest in the initial stage of the project (when the negentropy of manufacturing processes is the lowest). To increase negentropy, there should be constant references to the ADM. It is worth verifying the degree of project implementation in view of the objectives stated in the contract, by creating relevant documents for this purpose.

Thus, in terms of the quality of documents, there are two types: mandatory documentation and current documentation (describing the current state of processes and organisations). The first type should be evaluated according to the degree of development of documents (based on their number). The second type is more challenging. Taking into account the number of documents in difficult cases (*e.g.* in less mature organisations and processes), when the number of created documents is large, would wrongly indicate high quality of the documentation created. Therefore, it is proposed that such an evaluation be used only if there are no problems with the implementation of the manufacturing process.

The ITM Section of the Continuum

The starting point for the description of the ITM section of the Continuum and its height (pr_t) is Fig. 3.1. According to the figure, there are hierarchical areas of the manufacturing and management of the enterprise architecture development, defining the lowest manufacturing processes (which indicate a small negentropy of the project) to the highest areas of management technology (associated with a big project negentropy). The following description includes detailed characteristics of the different areas, taking into account two features: sequencing and the level of architecture development.

Managing Sequence. The sequence of the ITM section of the Continuum is associated with the order of management and enterprise architecture development processes with the use of best practices (groups of processes used by IT project managers). The need for processes and technologies of manufacturing and management in enterprise architecture development is pointed out. In the development process of enterprise architecture, as shown in Fig. 3.1, best practices and management technologies should be used sequentially, along the axis of the system coordinates (the axis of the ITM section). Failure to maintain the sequence of processes (*elementary organisation*) is the basis for classifying the project (EA development) to the group of projects with low negentropy. The development of enterprise architecture, according to the TOGAF standard, should consider changes in business processes, the introduction of new technologies, standards and manufacturing standards at the level of the Continuum. It should also take into account changes in the maturity of domain architectures and in constructing and identifying the Continuum components (as part of the supporting process). Ignorance of these changes is identified with low project negentropy (based on enterprise architecture).

Another area is the best practices used in manufacturing and management. According to TOGAF, they constitute an ordered set of rules and standards for the development of enterprise architectures. They refer to the use of IT in the development of architectures (as we also do in this book), and IT resources within the enterprise. They present rules governing the process of architecture development and implementation, as well as the initial development of objectives and guidelines relating to the development of information systems. These practices should also be subject to a formal program of development and certification. If best practices are not applied in manufacturing, then the development of EA is classified as a project with low negentropy.

Best practices reflect the experience of managers in the manufacturing of IT systems. A specific set of best management practices are Governance procedures. Their task is to support managers in monitoring the ADM processes and their compliance with the norms of TOGAF.

The Continuum resources do not contain detailed guidelines for the use of specific management technologies in supporting the development of EA. This is connected with the concept of separating the management from the technology (as is the case in manufacturing).

Determining an appropriate management sequence allows EA development to be treated as a project with high negentropy. If the adopted sequence is not maintained, the allocated evaluation of negentropy is reduced. It is low if the manufacturing processes are inappropriately defined (badly allocated resources: time, budget and people). Higher project negentropy is attributed if best management practices are used in manufacturing processes.

The Degree of Implementation of Management Processes. Assuming that this degree is the second indicator which informs the decision maker about the progress of the processes within the ITM section, their specification should be provided, as well as their evaluation method. Since the TOGAF standard describes these processes in detail, here we will focus only on their evaluation narrowed down to some examples of ITM sections. The degree of implementation of the processes is an important element of project negentropy. So if the team leader is able to carry out such an evaluation, they will also be able to make an evaluation of project negentropy. The examples will focus on a review (under Governance) of the compliance of the development processes with the best practices contained in the Continuum.

When describing current development processes, the project manager can refer to the Governance processes contained in the Continuum, which relate to the development, monitoring and control of documents necessary for the preparation of the contract and to ensure compliance of the current processes with the processes in the Continuum.

Monitoring the compliance of EA development (in Governance) includes these processes:

1. detection of errors in the development of an architecture
2. application of best practices in the development of an architecture
3. reviewing compliance with enterprise architecture (based on standards)
4. showing the possibility of modifying the standards
5. determination of specific applications/services used in the infrastructure of companies
6. development of a document of cooperation strategy of the EA development teams
7. application of new IT (methods and tools) in projects
8. management of communications of project partners
9. decomposition of orders (*e.g.* the inclusion of COTS and product documentation)
10. detection of significant differences between the degree of using the Continuum resources (development level) and the degree of provider maturity, in the EA development process.

Using the concept of the degree of process implementation and its metrics, we suggest the quantitative and hierarchical approach, i.e. to estimate the number (or a degree) of completed individual processes, the weights determining the worth of these processes and the overall weighted measure of the degree of implementation of the manufacturing and management processes of the company. This three-stage procedure facilitates the evaluation of the degree.

When evaluating the degree of implementation of the above-mentioned management and manufacturing processes, we shall begin with process 1: the evaluation of the error detection process. This depends on the number of system functions at the level of business architecture and the methods for their specification. In the absence of such functions, the vision of the system should be considered, and if it does not exist – the functional specification of the system architecture (applications and data). For process 2, the basis for evaluation should be a list (if it does not exist, it needs to be created) of best practices applied in the development of enterprise architecture. For process 3, the adopted standards should be related to examining compliance and the degree of implementation should be analysed (a simple comparative evaluation of the degree of implementation and a presentation of implementation proposals in line with a given standard). Process 4 has a logical evaluation (YES or NO) and is not subject to quantitative evaluation.

Process 5 should be evaluated in terms of the number of services or specific applications required for the enterprise. It is worth noting that their number does not have a direct impact on project negentropy because the business requirements can be met by one or more applications. Generally, however, a smaller number of applications means higher project negentropy. Process 6 (development of cooperation strategies of teams developing business architecture) can be evaluated from the point of view of its occurrence (YES / NO). Process 7 (the application of new technologies) must be defined differently (when the logical evaluation YES/NO does not apply). There are no measures indicating any relationship between the number of applied technologies and their impact on project negentropy. Therefore, the level of the applied technologies does not have to be evaluated. A similar situation occurs in the case of process 8. Process 9 refers to the specification of the principles for preparing procurement of software components. It is difficult to state whether the number of principles has a direct impact on the value of project negentropy, or the contrary effect. Process 10, in which significant differences between the degree of provider maturity and the level of architecture development are determined, should be treated in a similar way due to the fact that achieving an unambiguous evaluation is also difficult. It can be assumed (as in the evaluation of other processes) that a difference favours or limits project negentropy. Best management practices suggest that the smaller the difference between provider maturity and the level of development, the higher project negentropy is.

Although most of the above processes are difficult to evaluate in a quantitative way, it is worthwhile to look for an unequivocal measure. It seems that the application of best practices in relation to the measurements of these processes is the best option (for the evaluation of the above process hierarchy). Applying an expert evaluation/classification of these processes (as described by TOGAF) is also a recommended solution.

Other management processes which undergo evaluation are presented below:

1. approval of the contract for architecture development and adoption of formal procedures for its acceptance
2. approval of schedules, Service Level Agreements, and allocated requirements (*e.g.* for new forms of services)
3. ensuring that all relevant information regarding architecture implementation, which is stated in the contract, has been published in as reports available in the Continuum
4. creation and maintenance of the relationship between architecture objectives, strategies for architectures and their implementation, and the strategic goals of a company.

Process 1 has a single occurrence (or may not occur). Its evaluation is simple - both in terms of its occurrence, and the terms of its weights and measures. The evaluation of process 2 is similar, except for the addition of new processes in the course of the project, when the values of the weights need to be changed, taking into account all the processes. The evaluation of process 3 is more difficult and requires a definition of the concept of 'all the relevant information'. Similarly, many concepts need to be established for the classification of process 4. The identification of such processes should take place gradually, while medium or large negentropy should be attributed to the project.

The TOGAF standard (see Section 2.2) characterises previously described management processes treated as Governance. Monitoring of the degree of compliance (z_TOGAFt) should take place both at the level of SLAs (providing Governance services) and at the operational level of the organisation. Projects of various sizes are monitored on the basis of a series of questions (checklists) posed to members of the manufacturing team by the CAO. For larger projects, the review involves collecting and processing knowledge about the functions of the EA, which, in the largest projects, refers to all the teams involved in the project. However, will such a far-reaching specification (as in monitoring) be used by the manager of a project with small scalar negentropy? In the development process, the adoption of a reasonable approach (taking into account specified processes and their evaluation according to the hierarchy stated above) to assess the quality of the realisation of Governance processes is extremely important.

Both best management practices and management processes require the extraction of a manufacturing team for the implementation of EA on the basis of the ADM. Such a team can use the approach proposed here, formalising the data of the TOGAF standard.

3.3.2 Development Technology of Enterprise Architecture

As with the analysis of the Continuum repository, architectures, documents, and the ITM section of the enterprise architecture will be evaluated with the use of variables (a_t, d_t, pr_t).

ADM Architecture

The evaluation (for the benefit of project negentropy) of architectures developed in the ADM will be carried out in a similar way as was the case with the Continuum. First of all, the variable of ADM development (a_t) will be evaluated and its values associated with the quality of ADM architectures will be considered.

Development Sequence. We will first consider the question of how the architecture development sequence behaves within the ADM. For many companies, enterprise architecture development, according to the entire development specification shown in TOGAF, is not possible. Therefore, we will present architectures which are developed as part of the ADM, and the development sequence accepted there (Fig. 3.9).

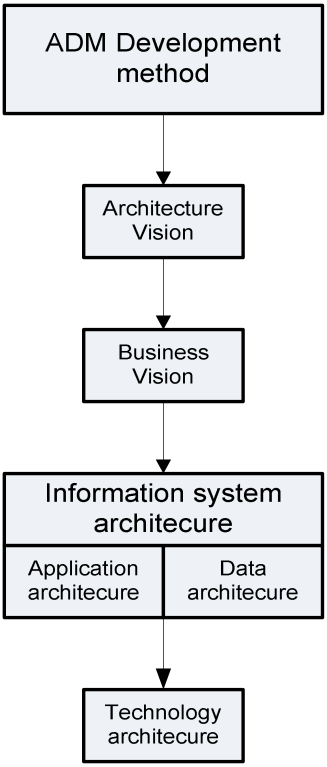


Fig. 3.9 Enterprise architecture components and development sequence

Architectures can be developed selectively, although it can be difficult to first develop technology architecture and then business architecture. However, if a company decides to implement EA using the *bottom-up* approach, then such a solution can be accepted.

Quality of Development. Six levels are used to evaluate the quality of architecture development: null, very low, low, medium, high and very high, with the following assumptions:

- architectures developed at the very low and low levels are those in which none of the architectures has been accepted, their structure is not complete, and some of them have not been developed at all (it is assumed that if only an information systems architecture has been implemented, the level of these architectures is considered to be low)
- architectures developed at a medium level are those in which only the business and information systems architectures have been developed, or all architectures have been developed but most of them have not been accepted

- architectures developed at a high and very high level are those in which all types of architectures have been developed (very high) or the majority of them have been accepted (high).

The five-element set of quality levels of architecture development is a result of combining the available expert knowledge with the authors' experience in implementing IT projects. The assumed method of evaluating the level of architecture development is slightly different (because it contains the classification of architectures into groups) from the one used for areas (Section 3.1) and documents (Section 3.2). This division was also used in the approach proposed in Chapter 2 and in the presentation of the development quality classification method in terms of the evaluation of project negentropy.

Isolation of the five levels of architecture development is also connected to the conditions imposed on the implementation of architectures (as specified in TOGAF), which include: the scope of the project, the scope of architecture development, the level of detail and the time scale. They should be applied to the adopted level of architecture development. We assume that these requirements will be classified linguistically (as very small, small, medium, big or very big). It is obvious that different levels of development can be assigned to different requirements (*e.g.* in the short term, the development of only two architectures: business and information systems, as representatives of the four enterprise architectures developed in accordance with the ADM).

Enterprise Architecture Documents

For the description of the enterprise architecture development standard to be complete, an evaluation of the documentation quality of enterprise architecture development is required. Therefore, assuming that the ADM development sequence in creating the documentation is maintained, we will evaluate only the degree of its development by determining the length of the documentation catalogue d_t. In this way, the description of architecture development will be completed by the scalar evaluation of project negentropy.

ADM Document Development Sequence. As is the case with the construction of the Continuum, in the case of the ADM there is also a need to create documents in a specific order. It is difficult to imagine that enterprise architecture development could be undertaken without the existence of initial documents, such as the contract and the orders sent by the client to the provider. The development of project documentation is a heavy method domain [43]. With a light approach to enterprise architecture development, this documentation can be omitted (as the creation of documentation to close a small enterprise project need not be relevant).

Degree of Document Preparation. Documents (developed in an appropriate sequence) are important from the point of view of the EA development process, and their degree of development is correlated with the meaning attributed to them (for the EA development process). Documents required for enterprise architecture development and its management according to TOGAF will be discussed quantitatively, disaggregated by the ADM processes.

The architecture vision project entails the development of documents referring to the initial and the target business architecture. With management based on heavy methods [53], the set manufacturing and management documents describe this vision and the verification method of business scenarios. The management documents define the roles of key persons, and present gap and business requirements analyses for the development of this architecture.

In documentation used in the construction of information systems architecture, five documents are used in accordance with TOGAF, including two manufacturing documents relating to business data models and the company data, and three management documents concerning rules, manufacturing patterns and a gap analysis. A similar approach to the development of documents is applied in the case of application architectures.

A similar method is used in evaluating the quality of documents used in the development of technology architecture. In this case (as in the evaluation of the ADM architectures and processes) documents assigned to the development processes should be considered.

In the first step of the technology architecture development process, one management document is used, relating to the construction principles, as well as two manufacturing documents describing the models and blocks of the developed architecture. In the second step, one management document and two manufacturing documents are also developed. In the third step, four manufacturing documents related to the products are used and one management document describing the change in technology architecture. In the fourth step, there is one management document and one manufacturing document, both relating to the technology architecture development process. In the fifth step, a list of questions to the client is used (instead of the traditional specification). In the sixth step, the basic document defines the selection criterion and the final specification. Manufacturing documents in this step describe the target technology architecture and project indications. The seventh step has three manufacturing documents and two management documents relating to the use of the Continuum resources. The eighth step focuses on the final document of technology architecture development, while the management documents refer to the analysis of gaps resulting from the development process.

To provide more details for discussion, Tables 3.1 and 3.2 show an example of an evaluation based on estimates of manufactured artifacts, taking into account manufacturing and management documents and evaluating them quantitatively (separately for each architecture).

The prepared list of manufacturing and management documents for enterprise architecture (made on the basis of the material in Section 3.1 and Appendix 1: TOGAF description), provides an evaluation of the degree of realisation of the development processes. Table 3.1 shows the qualitative classification of the manufacturing and management processes by evaluating the number of documents which describe these processes. Establishing the number of documents prepared for the project, in relation to the number of required documents (as defined by TOGAF), is easier than defining the number of processes; however, it allows an easy estimation of the degree to which the manufacturing and management processes have been realised.

Table 3.1 Evaluation of the management and manufacturing documents of EA

Documents/ Architectures	Vision	Business	IT systems	Technology	EA (entirety)
Manufacturing documents	Number of prepared documents	Number of prepared documents	Number of prepared documents	Number of prepared documents	Very small/ Small/ Medium/ Big/Very big
Management documents	Number of prepared documents	Number of prepared documents	Number of prepared documents	Number of prepared documents	Very small/ Small/ Medium/ Big/Very big
Architecture evaluation	Very small/ Small/ Medium/ Big/ Very big	Very small/ Small/ Medium/ Big/ Very big	Very small/ Small/ Medium/ Big/ Very big	Very small/ Small/ Medium/ Big/ Very big	

The EA column of Table 3.1 contains a cumulative pentavalent linguistic evaluation developed on the basis of partial evaluations (vision, business, IT systems and technology) via: the conversion of numbers into a degree (%), the choice of the lowest evaluation (along a given row), and its linguistic description (according to the given description). Such a cautious approach reflects the principle that it is the worst developed architecture which determines the success or failure of the EA development process. The documents of the different architectures (the last row in Table 3.1) were evaluated in a similar way, considering the lowest evaluations.

The preparation level of management documents can be determined in the same way. Although here the final evaluation may be more difficult, as it requires the manufacturing and management documents to be determined in percentages. If their number is comparable, we can talk about a joint evaluation (1, 2, ..., 5). However, if the number of manufacturing documents is low (2), and the number of management documents is high (4), then the question is how to evaluate the preparation level of the management documents? A situation in which many management documents, but only a few manufacturing documents were prepared, should raise doubts about the effectiveness of the management methods (a low number of manufacturing documents suggests insignificant development efforts). There is also a question of whether the level of project management can be evaluated only on the basis of the number of documents created? The experience and knowledge of IT project management indicates that it can not. Therefore, we recommend conducting an evaluation of the quality of the documentation, as a component of the target scalar negentropy of the project, to be carried out on the basis of expert knowledge, as was the case in the evaluation of the Continuum documents [43-45].

The ITM Section of Enterprise Architecture

After discussing architectures, we will proceed to discuss the ITM section of the ADM to estimate the variable (prt). The ITM section includes manufacturing and management processes and best practices obtained from the Continuum as well as manufacturing technologies (own or borrowed), for which the Continuum is the source. Below we present the evaluation method of the quality of the architecture development process in specific areas.

Table 3.2 Evaluation of the ITM: the level of architecture management and manufacturing

The ITM section/ Architectures	**Vision**	**Business**	**IT systems**	**Technology**	**EA (entirety)**
Processes of manufacturing+ management+ technologies	Number of completed processes	Number of completed processes	Number of completed processes	Number of completed processes	Very small/ Small/ Medium/ Big/ Very big
Best practices (the Continuum resources + own resources)	Y/N/P – Yes/No/ Partially	Y/N/P – Yes/No/ Partially	Y/N/P – Yes/No/ Partially	Y/N/P – Yes/No/ Partially	Very small/ Small/ Medium/ Big/ Very big
Management technologies (the Continuum resources + outside applications)	Y/N/P – Yes/No/ Partially	Y/N/P – Yes/No/ Partially	Y/N/P – Yes/No/ Partially	Y/N/P – Yes/No/ Partially	Very small/ Small/ Medium/ Big/ Very big
Architecture evaluation	Very small/ Small/ Medium/ Big/ Very big	Very small/ Small/ Medium/ Big/ Very big	Very small/ Small/ Medium/ Big/ Very big	Very small/ Small/ Medium/ Big/ Very big	

Management Sequence. It is extremely important to retain the architecture development sequence, shown in Table 3.2 (in terms of the *ITM*, *i.e.* along the advancement axis of the *ITM* section), which refers to the level of support for management processes. This sequence begins with own management processes, goes through the use of best practices (own or from the Continuum), and ends with the use of management technology (from external applications or from the depot). Use of the sequence is associated with the postulate of a preliminary analysis of the manufacturing processes before the application of best practices for their management.

Let us consider a case [46] in which a manager applies best practices to predict the manufacturing processes. Management technologies are the highest level of process management – therefore, they are used only after the manufacturing processes and best practices are defined. It should be noted that the TOGAF standard does not give specific technologies, but assumes that the project manager will select them adequately to the scalar value of project negentropy. The sequence presented in Fig. 3.1 is therefore a natural evolution starting with *elementary organisation of development*, and ending with the integration of *management technology*.

Level of Process Realisation. When evaluating the level of realisation of manufacturing and management processes, a division of areas was carried out, shown in Table 3.2 (Fig. 3.9) which takes into account the type of the developed architecture. As was explained at the beginning of sub-Section 3.4, the quality of the processes is expressed in terms of levels described linguistically: YES, NO, PARTIALLY, and Null/Very small/Small/Medium/Big /Very big .

The three-stage management technology evaluation means: no technology used (NO), technology applied (YES), the use of technology from the Continuum resources or a borrowed application (PARTIALLY). The EA column of Table 3.2 (resultant of such architecture types as vision, business, IT systems and technology) contains a cumulative pentavalent evaluation.

In Table 3.2, the first line shows the number of completed (reflecting the quality of implementation) manufacturing and management processes (the number was expressed as a collective linguistic evaluation of the realisation level). These processes were related to particular architectures developed within the ADM. The vision architecture takes into account the processes suggested by TOGAF which are used in the development of the vision of EA. According to this standard, six manufacturing processes and three management processes are defined, related to Governance, and three – related to the management of the development of a vision architecture. A CAO should examine all of these processes and decide whether to use them as a whole or their parts to build a vision architecture. It is relatively simple to establish the number of required processes. However, it is much more difficult to evaluate the degree of their implementation. It is therefore suggested that only the number of processes should be taken into account, without analysing this degree. The project manager should therefore evaluate the initiated processes and qualify them as completely realised or failed.

The level of the manufacturing and management processes in the construction of a business architecture should be analysed in a similar way. TOGAF defines seven major manufacturing processes, and (beyond Governance) two main management processes focused on a gap analysis. A similar solution should be used here as was the case with the architecture vision (a subjective assessment of processes and an objective assessment of the number of completed ones).

In the construction of an information systems architecture, the management and manufacturing processes should be divided in terms of data and application architectures. For data architecture, nine major manufacturing processes and four management processes are defined. The method of evaluation is similar to the one described above. Application architecture is treated with a similar approach to that of data architecture development (in this case TOGAF does not specify the processes of manufacturing and management).

The evaluation of the degree of realisation of the technology architecture development processes looks different (these processes are iterative). The first step includes nine manufacturing processes and six management processes. Step two involves two manufacturing processes and two management processes. Step three includes seven manufacturing processes and three management processes. Step four involves only four management processes. Step five involves two manufacturing processes and one management process. Step six involves two manufacturing processes and overall management processes. In the seventh step the manufacturing processes focus on obtaining documentation and on modifying the technology architecture with the use of five processes, and on management with the use of three. Step eight covers manufacturing processes focusing on the preparation of documents which contain a description of the technology architecture.

The above specification provides an evaluation of the degree of realisation of all the processes within enterprise architecture development. It is therefore necessary to take into account the evaluations obtained for each (developed) architecture and on this basis to generate a quality evaluation of the entire system. We recommend that a principle is adopted, stating that the set of architecture development processes and the set of management processes are positively evaluated on the linguistic scale (very small/small/medium/big/very big level of realisation). It is also advisable that the managers (based on their own experience) should assign high priority to the architectures which are most relevant to the objectives of the company, and, in evaluating the degree of realisation – they should take them into account.

The specification given above does not apply to best practices. Their general nature makes them difficult to quantify (especially in terms of applicability for specific architectures). The suggestion therefore is to count the number of entries to catalogues included in the Continuum which contain best practices if the Continuum specification divides best practices according to individual development and management processes. In the overall evaluation of best practices, the example given in Table 3.2 should be taken into account and those practices should be chosen from the Continuum which have a significant impact on the development. Weights can also be used to express the priorities assigned to practices by the project manager. In this way, the method which takes into account the identification of entries to the catalogues with best practices and which uses the weights of these practices, may also provide a basis for a quantitative evaluation of best practices.

The estimation of information technologies used by a company looks slightly different. The TOGAF specification does not include the type of technology and does not provide IT solutions which could be useful in the development and implementation of enterprise architecture. Therefore, it is advisable to create information technology resources in the Continuum (within the supporting process), which – as is the case with processes and best practices – will facilitate decision-making regarding the selection of appropriate technologies. The construction of such resources should take into account their functional specification and their selection criteria for developing enterprise architecture. It is unlikely that different technologies could be used for the management of each of the developed architectures (many technologies supporting IT systems means that the project is complex and difficult to manage). However, certain functions of technologies (like a web server) can be used. The selected function can therefore imply a type of technology (taking into account the dedicated features of the developed architecture).

The last line in Table 3.2 was created (like in Table 3.1), by evaluating the artifacts of particular architectures on the basis of their lowest evaluations. For such an evaluation of different architectures (in terms of the processes of the ITM section) an appropriate linguistic conversion from three to six levels is required (as explained in the introduction to Section 3.4).

The evaluation of manufacturing processes has already been discussed in detail. The question now is how such partial evaluations affect the overall evaluation of the EA development process (on a scale from 0 to 5). Another issue is about which areas affect the evaluation of the degree of realisation of EA within the ADM and in what way. The process arrangement shown in Fig. 3.1 (from elementary organization to management technology) reflects the fundamental importance of the manufacturing processes. The degree of realisation of these processes significantly affects the quality of the EA product and the applicability of management technology. If the manager decides to apply specific manufacturing technologies, he also have to use specific management technologies. With this distinction (into technologies of manufacturing and management), the position of best practices is unclear, as they represent a bridge between those two technologies. Therefore, when choosing the evaluation method of the degree of realisation of enterprise architecture, it is appropriate to take into account not only the manufacturing and management technologies, but also best practices – assuming, however, that the higher level (*e.g.* management technologies) includes the use of the lower level (*e.g.* best practices). If the manufacturing level is low, and the level of best practices is high, then the level of management technologies used should match the (low) level of manufacturing, and not the (high) level of best practices.

Thus, the demand for an equilibrium of proportionality of management and manufacturing technologies is obligatory here. The creation of other links between manufacturing technologies, best practices and management technologies is the responsibility of the project manager or an expert (in order to achieve an appropriate level of enterprise architecture development).

3.3.3 Measurement Specification of the Primary Negentropy of a Project

For the definition of project negentropy presented previously, (3.1) the measures for its evaluation must be selected. This specification should therefore include three variables:

- width of the architecture repository - a_t
- length of the documentation catalogue - d_t
- height of the *ITM* section – pr_t .

For the evaluation of each of theses variables, the procedure presented in Fig. 3.10 is applied.

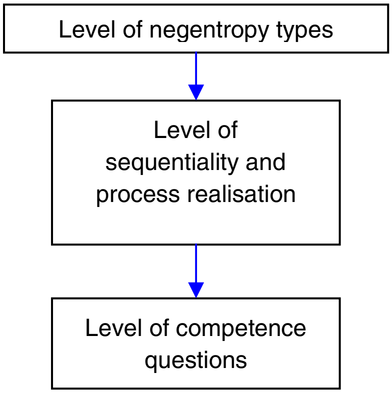


Fig. 3.10 Specification levels of the *project negentropy* variable

To determine the negentropy variable ‘width of the architecture repository’, the components of the project which affect its value were identified, with the assumption of two non-zero levels:

1. development level ‘the Continuum’ – *a1*
2. development level ‘the ADM’ – *a2*.

The symbols (*a1, a2*) represent positive values of the variable a_t (a negentropy component) evaluated in tests. Specific levels of maintaining the project realisation sequence, which are defined by sets of competence questions, were equally assigned to each of these values. The questions – on the basis of the response evaluation – are for the direct verification of which of the levels (*a1, a2*) is maintained. The number of competence questions for determining the development level (width of the repository) is shown in Table 3.3 (questions in Appendix 1).

Table 3.3 The number of competence questions for determining the width a_t

Level of development	Number of competence questions
a1	7
a2	3

For the evaluation of the ‘length of the documentation catalogue’ three values will be used:

1. low completion level of the preparation of documentation, ‘initial documents’ – *d1*
2. medium level of the preparation of documentation, ‘development documents’ – *d2*
3. high level of the preparation of project documentation, ‘transfer documents’ – *d3*.

The competence questions for establishing the value of the variable are presented in Table 3.4.

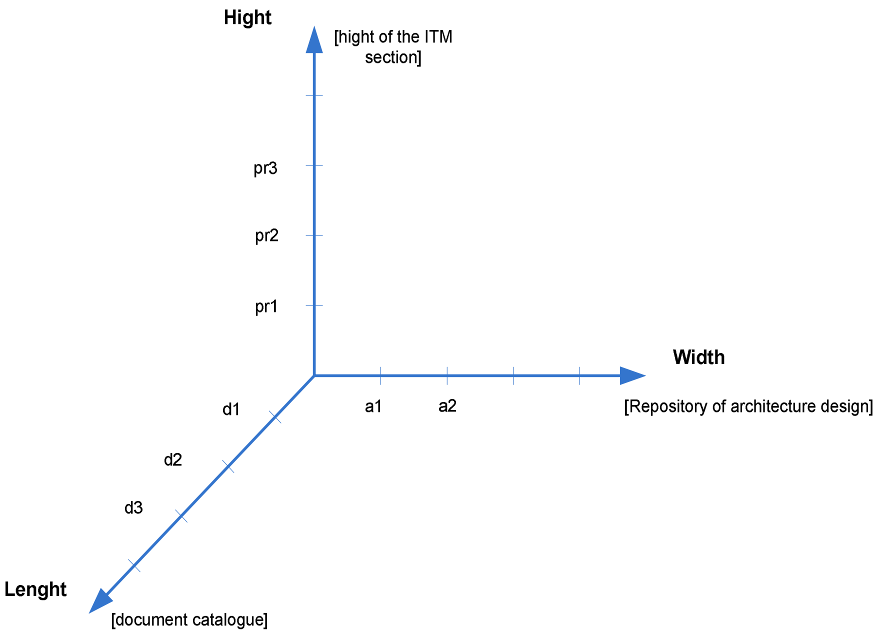


Fig. 3.11 Three-dimensional description of absolute project negentropy

Table 3.4 The number of questions for establishing the length of the documentation catalogue

Level of documentation	Number of competence questions
d1	4
d2	6
d3	5

To establish the third variable of negentropy: ‘height of the *ITM* section’, with the use of competence questions typified in Table 3.5, three non-zero values were assumed (Fig. 3.11):

1. level of processes and technologies of manufacturing and processes of management – *pr1*
2. quality (level) of best practices of manufacturing and management – *pr2*
3. height of management technology – *pr3*.

Table 3.5 The number of questions for the incremental establishment of the height of ITM

Level of technology	Number of competence questions
pr1	4
pr2	6
pr3	5

The information on the number of competence questions, included in Table 3.5, refers to the path of obtaining the first level *pr1* (starting from 0) in four stages (based on four competence questions), the *pr2* level in six stages (competence questions) and the *pr3* level in five stages (of competence questions).

Taking the above specification of the project negentropy variable into account, and relying on Fig. 3.11, the specification processes can be represented as a three-dimensional processing procedure. The specification of input variables proposed for large projects in the IPP model (small projects can be evaluated on the basis of knowledge of the domain and its tools) should therefore be extended by the description of the variable 'project negentropy' presented above.

An important conceptual novelty is the quantitative assessment of technological advancement (including information technologies, and manufacturing and management technologies), which is expressed by the variable 'height of the *ITM* section', and by the amount of project resources (the repository and the documents), described as 'project artifacts'. For practical reasons, secondary project negentropy (Δp_t), defined by the state of the applied technologies and the knowledge of the project domain (2.2), will be evaluated by the incremental value (2.22) as described by (3.5), in which the measure of technology is the height of the *ITM* section, while the expertise is measured by the size of the project artifacts.

In summary, according to the assumptions shown in Fig. 3.12, the vector of the initial prognosis of project negentropy should take into account the IPP specifications given in the diagram. The variables in the first layer: the height of the ITM section, the width of the architecture development repository and the length of the documentation catalogue (pr_t, a_t, d_t), are subject to evaluation. This involves determining progress in the technological processes of manufacturing and management (with best practices and technologies), as well as in the creation of the repository and the preparation of documents, appropriately on a scale of the level of technology, development and documentation. This is done with the use of competence questions posed to the provider organisation and the answers given to these questions. Formal vector descriptions of the decomposition and aggregation of variables according to specified levels are presented below. Whenever a conversion is necessary (from the multi-dimensional space), the method described in Appendix 4 can be used.

The analysis of changes in an IT project is crucial for management. Project negentropy (3.1) and (2.2) is determined by the increments (3.3) and (3.5), evaluating the progress in terms of the project negentropy coordinate, *i.e.* in the processes and technologies, in the preparation of documents and architecture development (Fig. 3.1), according to the specification of negentropy variables (3.3) on two or three levels for each of the three variables (Fig. 3.1 and 3.12).

Accordingly, first (Fig. 3.13) the evolution of the project is expressed according to (3.3) as a prognosis of the increment in the vector (three-dimensional) primary negentropy by separate increments in terms of the height of the *ITM* section, the length of the documentation catalogue and the width of the repository (it should be noted that in the formula (3.5) we have a two-dimensional prognosis of negentropy based on the height of technology and the size of the project artifacts: documents and architecture). In the case of the height of the *ITM* section/technology, the competence questions help to evaluate the progress in the use of ITM technology on a scale from zero (0) through *pr1* to *pr3* (Fig. 3.13 and Table 3.5). A similar procedure will be applied in the case of the other two variables of project negentropy (Fig. 3.12): the width of the architecture repository (a_t) and the length of the documentation catalogue (d_t). Figure 3.13 shows the 3D space of the considered increments representing the progress in the realisation of the coordinates of 3-dimensional negentropy/maturity of the project from Fig. 3.11.

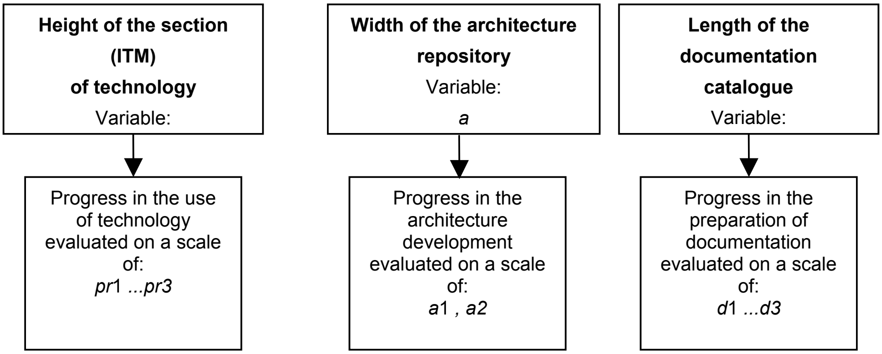


Fig. 3.12 The specification of the variable of project negentropy

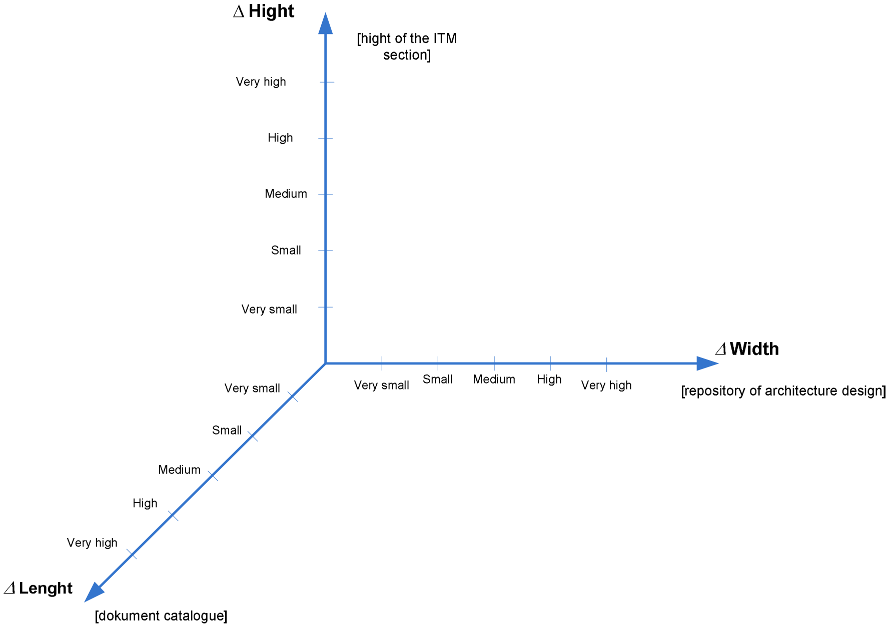


Fig. 3.13 IT management: Model of tracking progress in project negentropy coordinates in linguistic categories

Such progress can be uniformly evaluated in terms of linguistic values from the set {N, VS, S, M, B, VB}, where N (null), VS (very small or very low), S (small or low), M (medium), B (big or high), VB (very big or very high). The values of this increment (progress) result from evaluations obtained in response to competence questions (Table 3.5) posed to teams realizing an IT project (in a fixed time cycle of the project). As long as the progress in the implementation of technology application processes is rated as N, VS, S, or M, there is no transition. It is an increment evaluation rated as B, or VB (Fig 3.13) that causes a change involving a transition to a higher level of the *ITM* section (from 0 to *pr1*, *pr1* to *pr2* or *pr2* to *pr3*) on Figure 3.11 illustrating three-dimensional project negentropy used in the IPP processes. The progress in architecture development and the preparation of documents is evaluated and implemented in a similar way (Fig 3.13), where only the highest evaluations (B, or VB) determine the 'transition' to a greater width of the architecture repository or length of the documentation catalogue. Outside the decision-making category (in terms of changes in fragmentary negentropy/maturity) assigned to this progress, there is also the opportunity to follow the current progress (in anticipation of a change in the levels of pr_t, a_t, d_t).

The described method for the evaluation of project negentropy was used in studying the project negentropy and was the basis for the verification of the MITM model (Section 4.3.1).

3.4 Specification of Provider Maturity

In Section 2.5.2 of the work devoted to the IPP processes and the C-S-P maturity capsule, for the analysis of the variable 'provider organisation maturity' the universal CMMI model and the dedicated ITIL standard were used, while the key processes of manufacturing and management, and the maturity of the provider organisation were expressed by variables subject to the IPP.

The weak point of the standard is the fact that it is dedicated to specific provider organisations (implementing specialised, *e.g.* embedded systems). One can therefore continue the use of CMMI and apply it in the IPP. The IPP standard, ITIL, can also be used (which we recommend), as it is dedicated to modern information technology organisations providing services to both internal and external customers.

We therefore propose a solution which uses the ITIL standard while the CMMI model is a complement in the scope of the maturity evaluation of the provider organisation. The appropriateness of ITIL is a consequence of the state of information technology, in which information systems operate isolated information silos, blocks, people, processes and infrastructure in the realisation of the operations of a company.

If we also consider that fact that in a typical company (IBM study) [46, 49] only 13 to 28 % are key operations which have an impact (significant business-wise) on the company, then it should be considered whether the complete integration of information silos makes sense. In addition, another 55 to 74% of activities are less important in terms of business, while 32 to 47% are actions necessary to sustain the company but not directly related to its business. Therefore, the decision to implement other measures may not have a financial justification. Still, if other measures are implemented, they should be based on the SOA architecture, the construction of which is efficient in provider organisations focused on services.

A proper evaluation of the organisation can be effectively performed with the use of the ITIL standard (this is why the CMMI model will not be used here). The choice of method for maturity evaluation is primarily the responsibility of the CIO (*Chief Information Officer*). This is the person who defines the areas of integration of *information silos*, which are isolated areas of business (accounting, human resources, sales) deeply imbued with specific knowledge, made available only to authorised users. The selection of systems supporting the integration processes is carried out after a preliminary verification of the importance of the integration functions of these tools, in terms of business requirements. A process model of support services must be defined. The IT Service Support Model should also be defined for the evaluation of the service manufacturing processes according to the ITIL standard. The application of the IBM model (*IT Process Model*) [41] is also possible to evaluate the relevance of these processes and to support services – the creation of *On Demand Business*.

It is worth considering how IT should be introduced to these areas. The answer to this question helps to clarify the application of ITIL to evaluate provider organisations. The knowledge about the organisation (processes) is important then. A solution can be accepted in which, in relation to the less important areas and processes, the implementation can be outsourced (at lower costs) and such a task is treated as less important for the development of the company. Then the need to evaluate the provider organisation disappears. An example of such activities can be the implementation of systems (human resources management, logistics, finance and accounting). Activity with the greatest impact on the business effect, such as the implementation of an IT system, should be supported by the internal organisation. In this case, the application of ITIL acquires business significance. The external organisation often chooses the combination of both models of implementation: own activity and outsourcing. The concept of *outsourcing* IT services should be understood as a type of service involving commissioning the realisation of a project of a parent company from outside.

Here are some rules connected with establishing the model of implementation:

- *autonomy* ("Do it yourself") – a model for mature organisations, integrating for themself via the internal transformation of their infrastructure with the use of own resources
- *out-tasking* – a model for organisations with their own IT infrastructure, but looking for outside support for the development of their own IT infrastructure
- *outsourcing* – a model for organisations which use outside IT services and application management.

The term *out-tasking* of IT services is a type of service involving commissioning tasks related to the implementation of a given project of the parent organisation from outside.

From the viewpoint of the CIO, the choice of the model implies a valuation of the proposed IT solutions. The ease of the financial evaluation is related to the maturity of the provider. Two forms of payment are used: *standard* (fixed in advance) and smooth. The costs of the smooth approach are dependent on the number of users, the income derived from sales, the method in which the application is used, *etc*. Financial means devoted to the IT development are directly linked to business results, and indirectly to the effectiveness of the infrastructure. With the use of ITIL, the operating costs of the organisation can be determined with the TCO (*Total Cost Ownership*) method described below. To determine these costs, the concept of the maturity of the corresponding manufacturing and management processes must be defined.

The term: Total Cost Ownership – a method for evaluating IT investments which evaluates the costs of IT infrastructure before and after changes in a company.

When choosing the ITIL standard, it is worth referring to Gartner whose reports, when writing about the aggregation of data, indicate that [19] changing conditions in the business environment contribute to the creation of new factors stimulating companies to assume a strategic focus on data integration. Business factors such as immediate access to a market, or the ability to quickly change processes and business models, and necessitate good data management. Simplification of the IT processes and infrastructure is desirable to achieve transparency, which requires consistency and completeness of the data that describes the performance and operation of a company. Achieving this effect requires the perception of own organisation through ITIL.

Study of the maturity of the organisation (by its staff and experts) and the ITIL standard analysis lead to the conclusion [4] that the quantitative maturity evaluation should be based on the 'most critical' processes. Their choice also benefits the evolution of the organisation, which brings the need for further evaluation of the state and assessment of the evolution factors. Although the ITIL standard sets out the framework for the evaluation of the state of organisation, it does not indicate the change factors. Therefore, by introducing the variable 'provider organisation maturity', special attention is paid to the transition between states.

Assuming an increase of maturity, we will make the specification of the variable 'provider organisation maturity' based on the ITIL standard – assuming that the philosophy of ITIL can be implemented in the CMMI model. The formalisation of this variable on the basis of both standards increases the development flexibility for the IPP application.

In both methods of specifying this variable, the *top-down* approach will be used: both in reference to the *ITM* section (based on ITIL) and to enterprise domains (based on COBIT). Later, we will describe the processes of the provider organisation, the competence questions for ITIL, and the control objectives and the possibilities of their evaluation with COBIT. In this way, the similarities of the two standards will be emphasised, and their description will be organised.

Appendix 2 (Chapter 7) provides detailed information on methods for evaluating provider organisation maturity, necessary to explain new concepts and structures. The applied description of the standard allows for the separation of those parts of the content which have practical significance, from those whose evaluation is more difficult (for managers and other users applying this standard to evaluate the organisation). Presenting, in this Section, a complete description of the standard and the standard data useful in the specifications of this variable would confuse the message of this work.

Provider organisation maturity, based on the ITIL standard as described in Appendix 3, will be expressed in variables, representing different areas of services (Fig. 3.5):

- *Service Delivery – sd*
- *Service Support – ss*
- *Planning-to-Implement Service Management – pism*
- *Application Management – am*
- *ICT Infrastructure Management – ict*
- *Security Management – sm*
- *Business Perspective – bp.*

The values of these variables are fixed according to the three-level procedure of Figure 3.14.

Establishing the value of the first variable (service delivery) involves the identification of the elements which significantly affect the maturity evaluation of the service delivery method, which are arranged in the following sequence of management levels:

1. *Service Level Management – sd1*
2. *Capacity Management – sd2*
3. *Service Continuity Management – sd3*
4. *Availability Management – sd4*
5. *IT Financial Management – sd5.*

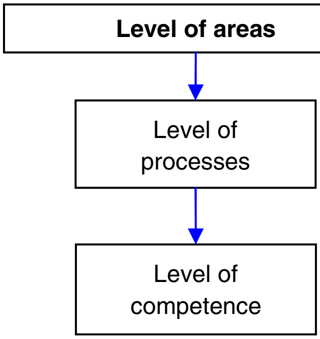


Fig. 3.14 Specification of the variable *provider organisation maturity*

It should be noted that as with negentropy (the level of which is determined incrementally so that we can track the progress of the transition between levels), here also the binary transition classification (*i.e.* the approval of the next level) happens after the increment has been evaluated as high, with the help of a different set of tests assigned to the analysed transition. The number of questions for the acceptance of the individual levels of the variable (*sd*) service delivery (*sd1*, ... *sd5*), and more specifically, the possibility to upgrade to these levels, are given in Table 3.6.

Table 3.6 The number of competence questions to fix the value of the *service delivery* variable

Service delivery level	Number of competence questions
sd1	3
sd2	3
sd3	13
sd4	3
sd5	4

The information on the number of competence questions, given in Table 3.6, refers to the path of achieving (starting from 0) the first level of *sd1* (on the basis of 3 competence questions), the *sd2* level in three stages (3 competence questions), the *sd3* level (13 competence questions), the *sd4* level (3 competence questions) and the *sd5* level (based on 4 competence questions).

In order to determine the value of the next variable referring to service support, we assume five values associated with the elements which affect the maturity evaluation of the service support method, listed in order of increasing management levels:

1. *Configuration Management – ss1*
2. *Incident Management – ss2*
3. *Problem Management – ss3*
4. *Change Management – ss4*
5. *Release Management – ss5*.

The method of achieving different levels of maturity (*i.e.* the value of the *ss* variable) of service support is the same as with the ‘service delivery’ variable (Table 3.6). The number of questions to determine the value (*ss1*, ... *ss5*) of the variable (level transfer) are in Table 3.7.

Table 3.7 The number of questions to determine the value of the *service support* variable

Level of service support	Number of competence questions
ss1	4
ss2	5
ss3	11
ss4	4
ss5	5

In the surveyed organisations [4], during the implementation of the ITIL standard version 2, special attention was paid to the processes of service delivery and support, and to their evaluation with the use of competence questions. In reference to other processes and variables presented below, we will narrow the discussion to the definition of the available values, without specifying the number of competence questions.

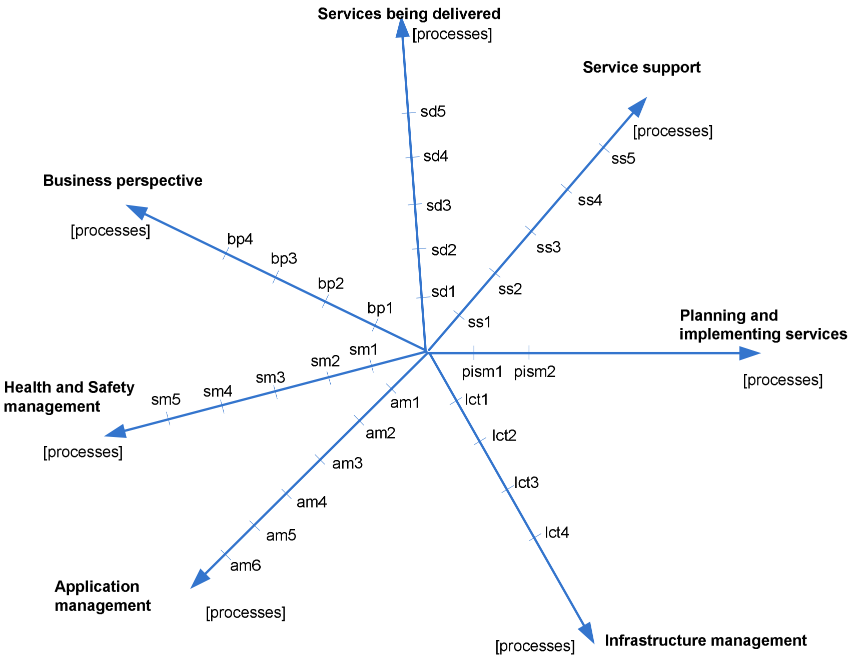


Fig. 3.15 Seven-dimensional description of provider organisation maturity

For determining the value of the third variable, (*Planning-to-Implement Service Management*) we assume only two management levels (indicated in Fig. 3.15):

1. *Service Planning Management – pism1*
2. *Service Implementing Management – pism2*.

For determining the value of the fourth variable '*ICT Infrastructure Management*', we assume four levels of quantisation (Fig. 3.15):

1. *Design and Plan – ict1*
2. *Deploy – ict2*
3. *Operate – ict3*
4. *Technical Support – ict4*.

For fixing the value of the fifth variable (*Application Management*), six management levels were singled out (compare Fig. 3.15):

1. *Require – am1*
2. *Design – am2*
3. *Build – am3*
4. *Deploy – am4*

5. *Operate – am5*
6. *Optimise – am6*.

To evaluate the variable '*Security Management*' five levels were assumed:

1. *Design and Plan – sm1*
2. *Implement – sm2*
3. *Evaluate – sm3*
4. *Maintain – sm4*
5. *Control – sm5*.

To determine the seventh variable (*Business Perspective*) four levels suffice:

1. *Business Continuity Management – bp1*
2. *Partnership And Outsourcing – bp2*
3. *Surviving Changes – bp3*
4. *Radical Change Transformation – bp4*.

In view of the above description, the specification processes of the 'provider organisation maturity' can be represented as a seven-dimensional variable processing procedure (Fig. 3.16).

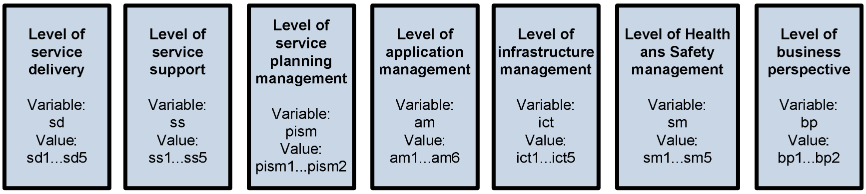


Fig. 3.16 Specification of the *provider organisation* maturity variable

Taking the above considerations into account, it is clear that the input variable specification processes (in the IPP, Section 2.5.2), need to be adjusted to the requirements of the provider organisation and to be integrated with the process division procedures (in accordance with ITIL).

As described above, in order to comply with the maturity provider formula (2.4), we assume that the maturity variable $ka1_t$ of the key IT systems manufacturing processes, integrates two variables: one of maturity in the area of service support (ss_t) and the other of maturity in the area of service delivery (sp_t). While the maturity variable $ka2_t$ of the key manufacturing management processes aggregate the maturity variables of plan-to-implement service management ($pism_t$), applications (am_t), infrastructure (ict_t), security (sm_t), and the maturity of determining the business perspective (bp_t). The aggregation method needed for this (conversion of maturity with dimensionality reduction) can be carried out according to the method described in Appendix 4.

According to Fig. 3.16 (process division), the vector of provider organisation maturity should take into account the transition between the values given in the diagram. Therefore, while evaluating the change in provider organisation maturity, the progress in the realisation of this process should be considered from the viewpoint of each of the variables presented in Fig. 3.16. This progress is evaluated directly through its (competence) assessment (two cases characterised in Table 3.6. and 3.7, other variables will not be described here).

Figure 3.17 shows a graphical interpretation of the considered increments representing progress in realising the different coordinates of the seven-dimensional organisation maturity from Figure 3.15. Such progress can be evaluated uniformly (like project negentropy) as linguistic values from the set {VS, S, M, B, VB}, or {1, 2, 3, 4, 5}. The values of the progress are based on the evaluation of answers given to competence questions by IT project teams (in a particular cycle). As long as the progress of the project is evaluated as VS, S or M, there is no transition. It is an increment evaluation rated as B or VB (Fig. 3.17) which leads to the transition to a higher level in service delivery (from 0 to *sd1*, from *sd1* to *sd2* or from *sd2* to *sd3* and higher, in Figure 3.17 illustrating the seven-dimensional organisation maturity in the IPP). Progress in the other six processes is evaluated and realised in a similar way, where only the highest evaluations (B, or VB) (Figure 3.17) determine the jump transition to higher levels. Thus, as with negentropy, apart from the possibility of making a decision (in terms of progress in provider maturity), we can also observe the current changes – in a quantified-linguistic way.

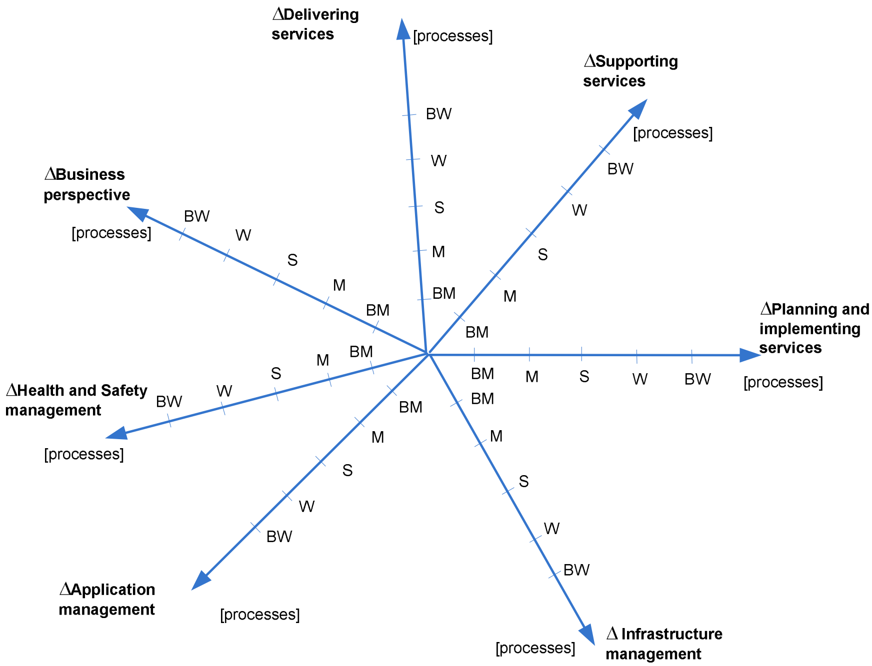


Fig. 3.17 Linguistic specification of progress (increments) in the coordinates of provider organisation maturity in the MITM model

The described method of evaluating organisation maturity will be used to verify the developed MITM model (Section 4.3.1).

3.5 Specification of Client Maturity

In the section on the initial processing of the ‘client organisation maturity’ variable, suitability and matching were discussed, as variables subject to the IPP. They are applied for small client organisations. Determining the values of these variables is then relatively simple and is based on specific knowledge of the client as a representative of the organisation. This occurs when the client organisation delegates one person or group to take part in talks with the provider/supplier on the realisation of an information technology project.

The values of these variables are determined in the process of analysis [23], in which suitability is assessed on the basis of the qualifications, experience and references of the potential co-worker, as well as an interview. It is easy to transfer this feature to the client or their representative. Qualifications are necessary when communicating with the provider’s representative. References describe the experience of the workers and their applicability in the project. The interview, in turn, reveals the competence for the co-operation between the client and the provider in determining the potential requirements for the system and its manufacturer.

It is also advisable [23] to evaluate client matching. What is meant by this is the client's abilities and openness to co-operate in the realisation of a project, which stem from their personal characteristics. It is recommended that a professional psychologist states whether the client’s representative is well-matched. It would appear that a combination of matching and suitability should characterise every mature client. However, experience from the implementation of IT projects (Table 3.8) shows that, in terms of cooperation with a client, matching is more important. Furthermore, as can be seen in Table 3.8, it is apparent that suitability, qualifications and professional knowledge make such a partner less flexible and, unfortunately, this means that the combination of suitability and matching of the client’s representatives does not always benefit long-term client – provider co-operation (Table 3.8).

Table 3.8 Classification of qualities of a potential representative of a client

	Matched	**Unmatched**
Appropriate	**Disappointment:** Ideal candidates move to a more lucrative position.	**Real problems:** dismatched employees. Reluctant to depart and they become difficult.
Improper	**Succes:** Surprisingly matched. They perform excellently in a randomly found job. They are content and remained on position	**No problems:** completely mismatched. They relieve themselves.

It could be argued that, in the event of long-term co-operation of the client (client's representative) with the provider, the client can mature along with the project (if the client possesses no qualifications at the beginning of the project), in the case of an IT project, it is not recommended that the client's representative is not suitable. Therefore, in the case of IT projects, clients should be classified as matching and appropriate. It also seems that the quantitative (acute) classification is of less importance and thus, the linguistic approach should be applied with the values of {null, very low, low, medium, high, very high} for each of the variables described in Figure 3.18.

Therefore, the maturity specification of the team or the client's representative comes down to collecting evaluations in the form of variables and their linguistic values. The record in the form of rough values would have formal significance, but be of little practical applicability. Most of these evaluations are made subjectively by the evaluation team, which greatly reduces the applicability of a model. Therefore, the specification (system of variables and their values) used in the modelling of the IT selection process for small client teams is a compromise solution.

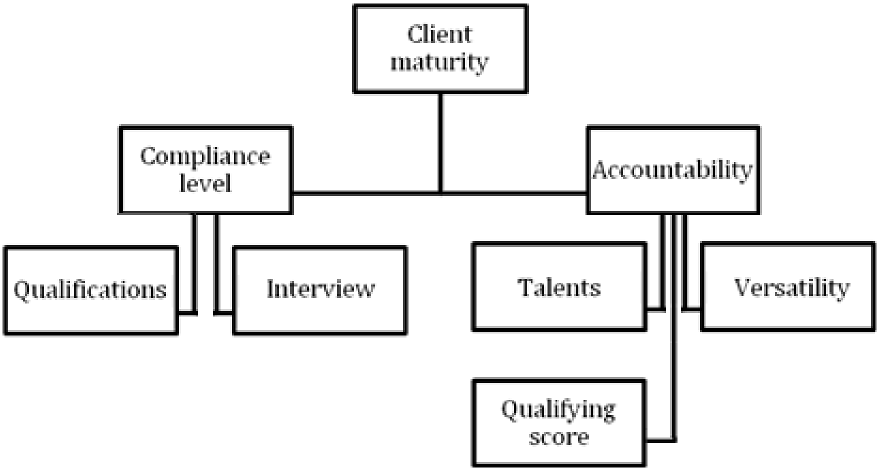


Fig. 3.18 Client maturity and its variables

A different approach to evaluating client maturity should be used in the case of a large client organisation, especially one with an IT infrastructure. Then it is difficult to describe such companies as people. Instead of team members, the evaluation should refer to processes which are carried out by them. The realisation method of the manufacturing and management processes stems from the organisational culture. Thus, the maturity evaluation of the client organisation should also take into account the method of cooperation between the members of this company.

In order to evaluate the maturity of the client organisation, several standards were taken into account. Both the CMMI model, discussed in reference to organisations, and the ITIL standard designed for the maturity evaluation of the provider were analysed. A set of standards proposed by the ISO could constitute another system for evaluating the maturity level of the client organisation. These standards were created in 1987 and then revised in 1994, in order to adapt to technological changes. The 9000 series standards are universal and organise the management of projects and the entire organisation so that the products manufactured by this organisation are of high quality. These standards do not provide hints for evaluating the maturity of an organisation, but on the basis of its ongoing processes it is possible to evaluate whether the firm is mature.

COBIT (*Control Objectives for Information and Related Technology*) is another standard used in this work. It is an open standard for controlling business processes, which has been developed by the IT Governance Institute and published by ISACF (*Information Systems Audit and Control Foundation*). COBIT uses generally accepted standards for controlling the application of best practices to support company needs, management and monitoring. The actions proposed in COBIT are designed to improve organisation maturity. Therefore, it has been proposed that the specification of the variable of client maturity should be based on this standard.

According to the concept of COBIT, we assume that the purpose of its use is to determine whether IT correspond to the business processes of a company and its resources. This evaluation philosophy can be applied to client organisations in which IT systems are being implemented. Thus, 4 domains, 34 control processes and 318 specific control objectives are named. A detailed description of the domains, control processes and control objectives are given in Appendix 3. The result of this control is to set management guidelines which include: maturity models, critical success factors (CSF), key goal indicators (KGI), and key performance indicators (KPI).

The application of the COBIT and TOGAF standards in the specification of the variable of client organisation maturity aims at evaluating the organisation to show its managers in which areas of IT management changes are essential. The application of a standard supports the process of defining the goals which the organisation seeks to achieve (by comparing the own control practices with best practices used in the IT environment). The standard is dedicated both for the management (to control the condition and the level of investment in the company), as well as for future users (to control the level of IT processes carried out internally or externally).

The COBIT standard should therefore be used to evaluate client organisation maturity. Appendix 3 includes details of COBIT, but only those which are necessary to explain the concepts and structures discussed here (avoiding the full specification). This allows for highlighting the content which is more practical for managers and users of IT systems who use this standard in the evaluation of the organisation.

Below we will define the variables (as in the case of project negentropy and provider organisation maturity) for a client organisation in which IT systems operate. The client represents a company based on outsourcing or out-tasking or is an employee of such an organisation (these concepts have already been discussed in the section about the provider).

After the presentation of variables specifying client maturity (as was the case with the specifications of the provider) we will proceed to determine the values of these variables according to the three-level procedure shown in Figure 3.19. The first level includes a decomposition of the adopted approach to organisation maturity referring to four variables labelled with names of domains, as shown in Figure 3.20. The second level includes the processes allocated to them, the number of which depends on the domain. These processes are then evaluated at the third level in terms of control objectives using competence questions.

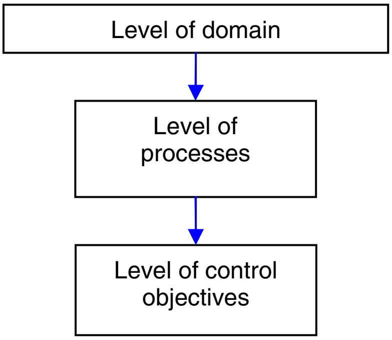


Fig. 3.19 Specification method of the *client organisation maturity* variable

The client organisation maturity evaluation was expressed by four variables corresponding to the domains, as shown in Figure 3.20: planning and organisation (*po*), acquisition and implementation (*ai*), delivery and support (*ds*) and monitoring (*m*).

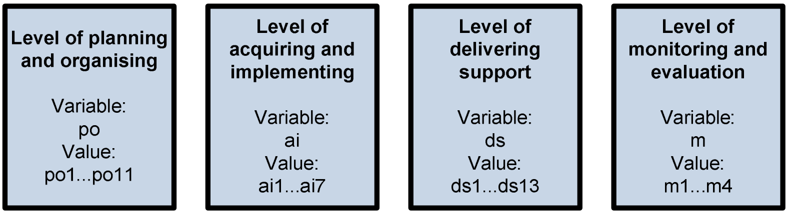


Fig. 3.20 Specification of client organisation maturity carried out on the basis of COBIT

Via an orthogonal spatial aggregation, these four variables form the client organisation maturity vector, in the four-dimensional maturity space as shown in Figure 3.21.

In order to evaluate the quality of the first domain of client organisation maturity, through the (quantitative) determination of the value of the 'planning and organisation' variable (*po*), we distinguish 11 levels:

1. defining the strategic plan – *po1*
2. defining the information architecture – *po2*

3. determining technological directions – *po3*
4. defining the IT processes – organisation and relationships – *po4*
5. IT investments management – *po*5
6. management through communicating goals and directions – *po6*
7. management of human resources in IT – *po7*
8. process quality management – *po8*
9. estimation and IT risk management – *po9*
10. project management – *po10*
11. product quality management – *po11*.

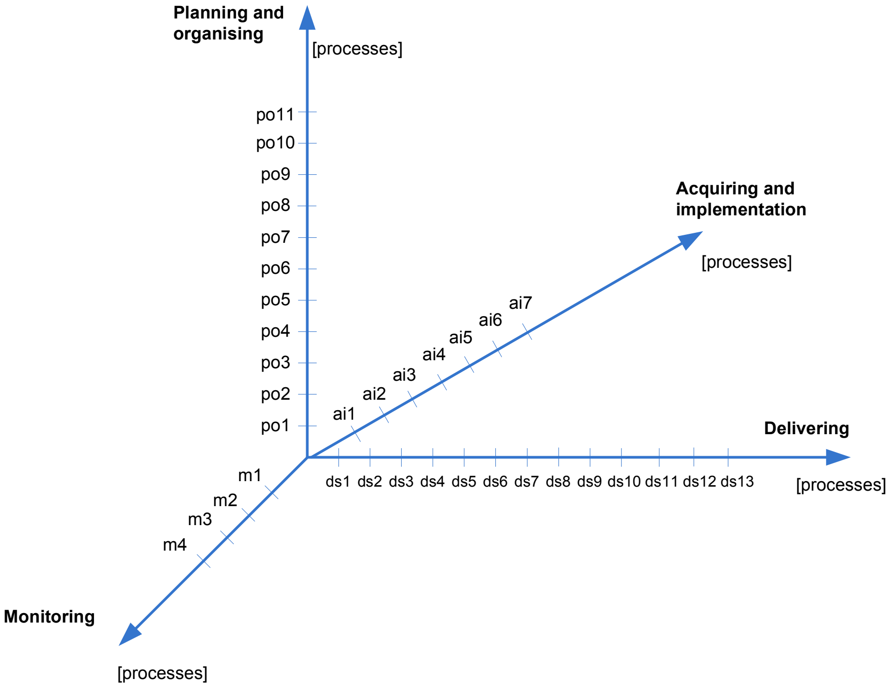


Fig. 3.21 Four-dimensional description of client organisation maturity in the MITM

It should be noted that, as in the case of negentropy (the level of which is determined in an incremental (jump) way with the possibility of tracking progress), here we can also see a binary classification of the transition (the level is either approved or not) only after a high evaluation of the increment, obtained through questions assigned to that level. The number of questions for the acceptance of individual levels of the variable (*po*) of service delivery (*po1*, ..., *po11*) is given in Table 3.9. As was the case with the provider organisation, the information on the number of competence questions refers to the path of achieving specific levels of maturity (*e.g.* the first level, *po1*, is achieved on the basis of 6 competence questions).

Table 3.9 The number of control objectives for the planning and organisation domain

Level of *po* variable	Number of control objectives
po1	6
po2	4
po3	5
po4	11
po5	4
po6	5
po7	7
po8	6
po9	6
po10	14
po11	5

To determine client organisation maturity within the acquisition and implementation domain (*ai*), suitable impact factors are needed, which determine the following 7 values – levels:

1. identifying automatic solutions – *ai1*
2. acquiring and maintaining applications – *ai2*
3. technological infrastructure of the acquisition plan – *ai3*
4. ensuring operation and applicability – *ai4*
5. acquisition of IT resources – *ai5*
6. change management – *ai6*
7. installation and accreditation of solutions and changes – *ai7*.

Table 3.10 presents the number of competence questions concerning the scope to which the client organisation meets the requirements of completing each of the paths. The first level, *ai1,* is achieved after meeting the requirements included in four competence questions, the second, *ai2*, – with the help of 10 competence questions, the third, *ai3*, and fourth, *ai4*, are determined on the basis of 4 questions, the fifth, *ai5*, of two, the sixth, *ai6*, of three, but the seventh, *ai7*, on the basis of nine competence questions.

Table 3.10 Number of control objectives for the acquisition and implementation domain

Level of *ai* variable	Number of control objectives
ai1	4
ai2	10
ai3	4
ai4	4
ai5	2
ai6	3
ai7	9

On a measurement scale for client organisation maturity in the third domain (*ds*, 'delivery and support'), 13 values – levels (impact factors) were determined:

1. service definition and management – *ds1*
2. third persons' service management – *ds2*
3. realisation and capabilities management (potential) – *ds3*
4. ensuring continuity of service – *ds4*
5. ensuring system security – *ds5*
6. identification and allocation of costs – *ds6*
7. user education and training – *ds7*
8. service desk and incident management – *ds8*
9. configuration management – *ds9*
10. problem management – *ds10*
11. data management – *ds11*
12. physical environment management – *ds12*
13. operations management – *ds13*.

The indication of each of these values is carried out in a manner similar to the case discussed above (for the 'acquisition and implementation' variable), based on a fixed set of competence questions, the number of which is given in Table 3.11.

Table 3.11 Number of control objectives for the *delivery and support* domain

Level of *ds* variable	Number of control objectives
ds1	5
ds2	4
ds3	5
ds4	20
ds5	23
ds6	7
ds7	3
ds8	7
ds9	3
ds10	4
ds11	4
ds12	5
ds13	5

For the quantitative evaluation of client organisation maturity within the fourth domain of 'monitoring and evaluation' (*m*), the impact factor was associated with the following levels:

1. IT efficiency monitoring and evaluation – *m1*
2. internal control monitoring and evaluation – *m2*
3. ensuring compliance with external requirements – *m3*
4. ensuring IT management – *m4*.

The possible values of the *m* variable are defined in the same way as was shown above, with the use of competence questions, the number of which is indicated in Table 3.12.

Table 3.12 Number of control objectives in the *monitoring and evaluation* domain

Level of *m* variable	Number of control objectives
m1	14
m2	8
m3	8
m4	6

Given the above specification, the processes proposed in the IPP model for large organisations (small client organisations are evaluated directly on the basis of suitability and matching processes) should be extended with new IPP procedures (expressed in Figures 3.19-21 and Tables 3.9-12) compliant with COBIT (Fig. 3.20). Taking into account the procedures, processes and variables, client maturity c_t can be specified by equation (2.5), if we assume that the variable hcm_t (of personal matching of the client) aggregates the variables po_t and ai_t and the hcc_t variable (of competence suitability of the client organisation) includes variables ds_t and m_t. A scalarisation method for such vector variables is described in Appendix 4.

According to the assumptions discussed above (regarding the division of processes and the selection of control objectives), the vector of client organisation maturity evolves partially among the fixed values of individual variables, arranged in a sequence, expressed in Fig. 3.21. In the evaluation of client organisation maturity, progress in the realisation of this process is considered from the viewpoint of four defined domains (Fig. 3.20-21). The progress is evaluated directly through the identification of competence and the acceptance of the transition (as structurally characterised in Tables 3.9-12).

In each domain, there is an identical sequential-iterative method of linguistic evaluation of the progress. Fig. 3.22 shows the geometric interpretation of possible increments representing the progress in the realisation of each of the four-dimensional coordinates of client organisation maturity from Fig. 3.21. Such progress (similar to project negentropy and provider organisation maturity) is evaluated linguistically according to the set {N, VS, S, M, B, VB} or {0, 1, 2, 3, 4, 5}, where N or 0 describe 'null', VS or 1 mean a level of ‘very small’ or ‘very low’, S or 2 – “small’ or ‘low’, M or 3 – ‘medium’, B or 4 – ‘big’ or ‘high’, and VB or 5 – ‘very big or ‘very high’. The values of this increment come from the evaluation of responses to competence questions posed to IT project realisation teams (in a fixed cycle of the project). As long as the progress in project realisation has a low evaluation (such as N/0, VS/1, S/2, and M/3), there is no ‘transition’. Only an increment evaluation of B/4 or VB/5 (Fig. 3.22) leads to a change involving a jump transition to the next, higher level in the domain.

The above method to evaluate the maturity of an organisation will serve as the basis for the verification of the MITM (the IPP and the maturity capsule) as described in Section 4.3.1.

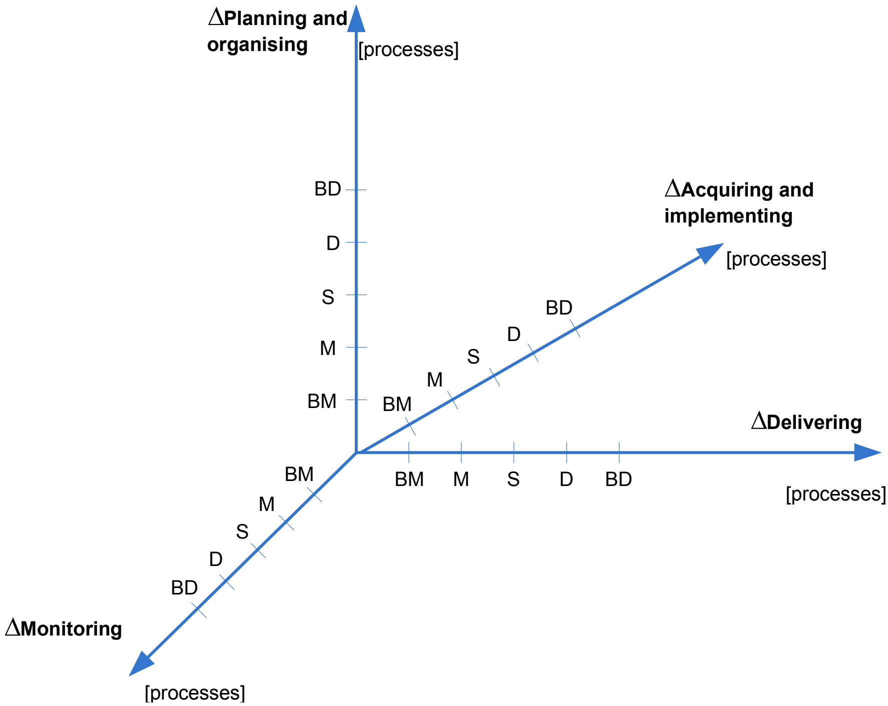


Fig. 3.22 Linguistic specification of the progress/increment in the coordinates of client organisation maturity in the MITM model of information technology management

3.6 Summary and Conclusions

Chapter 2 presented the specification of the IPP processes and of the maturity capsule, as well as their variables, on the basis of the concept of a vector description and the quantitative evaluation of the variables of the introduced vector coordinates (in terms of numbers, percentages, degrees and linguistic levels). Those specifications determined the three parts of this Chapter. Their application in the information technology management model, MITM, facilitates a uniform description. The specification of the 'project negentropy' variable required a more detailed explanation than the description of the other two variables (for example, due to the innovative concept of project negentropy).

With regard to the 'project negentropy' variable, the specification presented in Section 3.2 helped to structure its description, enabling the determination of its basic levels. Measures of scalar negentropy for the development of enterprise architecture were introduced, based on two cases: the Continuum and the ADM processes, as well as the so-called main cycle of development. The consideration of the two cases (with different negentropies) enriched the environment and conditions for the selection of negentropy measures. The construction of the Continuum repository formed the basis for the classification of project negentropy in terms of rough categories (1 to 5) or linguistic ones on a pentavalent scale (from very small to very big negentropy).

The analysis of both cases led to the specification of negentropy measures, which is based on the chart for determining the variables of the width of the architecture development repository, the length of the documentation catalogue and the height of the *ITM* section, (a_t , d_t , pr_t). These variables are determined with the use of appropriate levels of process realisation. Their monitoring is based on the result of competence questions, which allows for a linguistic evaluation of the increase in the value of negentropy.

For the specification of the variable of 'provider organisation maturity', the purposefulness of using the CMMI model for evaluating this variable was analysed. It has been proved that a better solution for service organisations is to use the ITIL standard. The concept is related to the changes that are taking place on the IT market, where most organisations act as support organisations evaluated and developed on the basis of the specifications of this standard. It is natural, therefore, to use the operation areas of the organisation instead of the key areas, while keeping the (three-level) structure of the description of organisation maturity. Thus they were treated as variables for the evaluation of an organisation. The areas were matched with key processes for their operation, and the analysis of the realisation degree of these processes (as in the case of project negentropy) was evaluated on the basis of sociological research, involving the analysis of answers to control questions concerning the quality (level) of the processes of an organisation.

The key organisation processes underwent a detailed specification. Their method of evaluation was also changed (in comparison with the traditional approach) to offer a linguistic interpretation on a four-level scale, since the five-point scale caused great difficulty to members of the client organisation. However, the evaluations obtained from the control questions were transposed to the more convenient pentavalent organisation maturity evaluation scale. The introduced dichotomy of evaluations (the four-level scale for responses to questions and the five-level evaluation of organisation maturity) reflects a natural divergence between the precision of expert opinion and the required accuracy of the evaluation of an organisation.

The third specified variable was 'client maturity'. It turned out that – as in the case of provider maturity – the adopted variables, suitability and matching, describe small client organisations or their representatives rather well. For larger organisations, the use of these variables was not sufficient to determine how the organisational culture of large client organisations reflects the quality of cooperation between the provider and the client. Therefore, client organisation maturity is expressed with a variable which undergoes the evaluation process with the use of the COBIT standard, introducing a linguistic description for the specification of all the variables of the MITM.

We consequently define the relevant variables relating to COBIT domains (Fig. 3.20), adopting the three-level procedure. The variable values describe the realisation degree of the control objectives of the client organisation, which are tested (as is the case with project negentropy and provider maturity) using control questions concerning maturity progress on a four-level scale of linguistic evaluation. The description of domains and processes with the use of variables and a linguistic evaluation of the control objectives formalise and simplify the task of client organisation maturity evaluation and its application in the model of information technology management (MITM), which was characterised in the early part of this work.

The analysis carried out in this book and the proposed MITM model suggest the need for changes in the organisation evaluation method using IT standards. In place of the initially adopted CMMI standard, for provider organisations we suggest using the ITIL standard. In the case of client maturity evaluation, in place of the initially accepted criteria of matching and suitability, we introduce the COBIT standard. In the case of project negentropy we base our discussion on TOGAF.

It should also be noted that the direct result of applying the standards to the developed MITM model is the demand for their adaptation to conduct research of the maturity capsule.

Since the primary mechanism for measuring the level of organisation maturity is derived from a subjective evaluation of its members, in order to improve it, the linguistic concept has been used enriched by weighting (relevance and reliability). Using a fundamental philosophy, the description expressed in the IT standards has been organised, offering a linguistic model both to evaluate organisation maturity, as well as for the assessment of responses to questionnaires supporting the process of this classification.

It should be noted in conclusion that the developed model and diagram of information technology management, MITM, as shown in Figures 2.8 and 2.9, is of a conceptual and fundamental nature. It constitutes a generalisation and a formal description of one of the proposed visions described previously (Figures 2.1, 2.2, 2.3) of the multi-level MSM model of information technology selection. The basis for the static processing of the IPP (for the selection of information technologies based on the MSM model) is presented in Figure 2.7, which, in an aggregated way, expresses the parallel (triple-channel) structure of the incremental determination of values of the individual components of the maturity capsule. In Figure 2.8, these channels have been integrated into a static sub-system in the MITM, constituting an entrance to the dynamic management sub-system, *the maturity capsule*, where the state variables are generated. The state variables of the capsule are transmitted to the decision-making sub-system (the implementation of the MSM model) in order to convert them into categories of information technology services.

The combination of the IPP model with the maturity capsule creates conditions for any implementation of the MITM model, since the applied interpretation of the described MSM model (MSMm and MSMt), in the form of a decision-making sub-system, is only one of the possible choices.

In the implementation applied and described in this Chapter, all the coordinates of the maturity capsule increase their absolute values according to increments estimated appropriately, in a questionnaire-linguistic manner. Both in the case of project negentropy, and in the maturity of the client organisation and of the provider, the system of a linear increase is observed (with the respective values of Δp_t, Δc_t and Δs_t, which are limited to represent the transitions between the successive levels). These increases, representing the dynamics of the capsule, influence (through the conversion function P_ϕ) the level of management, determining the final progress (through the function P_ϕ) in project management functionality (in terms of the team, technology and the processes of manufacturing and management).

The introduction of the maturity capsule (C-S-P), in addition to the comprehensive coverage of maturity in an IT project, is also a new approach to evaluating the increases of this maturity. Normally, it is assumed that with the progress of the project, the maturity of both partners increases (client and provider) and so does project negentropy (the organisation of the project increases). In this case, all the increases are positive and reflect (on the basis of the competence questions posed to the organisation) the increase in the absolute measure of maturity.

A doubt may arise at this point as to how this approach will reflect situations in which, along with the progress of the project, the maturity of the client and the provider decreases (this can happen in the case of failed projects). Since competence questions reflect relative increments – they can have the value of zero (very, very small on the linguistic scale), which, as such, does not translate into an increase in maturity. Thus, the capsule can maintain constant maturity. A lack of progress in maturity obviously means unfavourable stagnation at the level of management, which should be an important signal for the managers of the project.

The conclusion is, therefore, that monitoring of the maturity capsule should greatly facilitate the management process and lead to a significant reduction in the number of failed projects.

4

Verification of the Model of Information Technology Management (MITM)

It is a good idea to verify the model of information technology management (MITM) described in Chapters Two and Three of this book and demonstrate how useful it can be to those in charge of projects, organisations and provider and client teams in supporting the processes inside the project (*e.g.* monitoring and forecasting its progress). Due to the multiple aspects of the model, its verification will be presented on four levels, with the use of the classic *top–down* approach.

Level one – the implementation of the MSM model, refers to the evaluation of the MITM model's applicability for the purposes of IT project managers. To this end, we will perform an evaluation of the existing IT tools and then provide the description of the MSM model's implementation in a chosen verification environment. Next, we will demonstrate how the MSM model, implemented in a tool as a plug-in, supports project management processes.

Level two – the implementation of the IPP model and C-S-P maturity capsule in the client organisation. The project negentropy concept prepared for the MITM model will be applied in the Governance process in an organisation which provides enterprise architecture design services and in a company implementing the CRM system (*Customer Relationship Management*). This kind of application will be described in Section 4.3.

Level three – the implementation of the IPP model in supporting the provider organisation's evolution by applying linguistic evaluations of the degree of organisation maturity in the prepared model (of change forecasting). The measures will be applied in six experiments presenting various development scenarios of such organisations.

Level four – the implementation of the IPP model for the control of the client organisation's maturity level and change processes by applying the 'client maturity level' variable proposed in the study in the process of obtaining partial evaluations and a general estimation of the client maturity level.

Z. Kowalczuk and C. Orłowski, *Advanced Modeling of Management Processes in Information Technology*, Studies in Computational Intelligence 518,
DOI: 10.1007/978-3-642-40877-9_4, © Springer-Verlag Berlin Heidelberg 2014

4.1 Implementation of the MITM Model

The designed MITM model has been implemented and verified in practice in the *Jazz* environment from IBM (in accordance with the data support model *Blue Cloud*). All IT environments (related to the implementation of IT project management models) included in the analysis provided similar conditions for the experiment. Those platforms will be described with consideration to the functions required for full implementation of the MITM model. In addition, we will describe the implementation of the model in the form of a component integrated with the implementation environment (Jazz). The Chapter ends with a presentation of plug-ins developed on the basis of the prepared components.

4.1.1 MITM Model Implementation Environments

The first platform presented is Microsoft *Visual Studio Team System* 2008 (VSTS), which constitutes an integrated software environment for application manufacturing and manufacturing process management [60]. It involves using tools for software architects, software developers, testers and project managers in accordance with their relevant functions within a team. The environment includes management and manufacturing processes (easily adapted to projects) which provide teams with assistance in implementing changes in the process of software manufacturing, supporting the management of software changes and reporting in real time (on a current basis), and applying key quality and capacity standards. Project management based on a software environment also enables shareholders and IT management to monitor the project status, and helps software teams to improve the manufacturing process. It also increases the team's capacity owing to advanced product life cycle tools and intuitive operation adapted to individual functions in the manufacturing team. Moreover, it provides a number of compatibility functions which facilitate communication between manufacturing, operation and business teams. Together with the Visual Studio Team System environment, the MSF packet (*Microsoft Solutions Framework*) is provided, which is a set of best practices used in project management [50, 52].

Compared to the Jazz platform offered by IBM, VSTS provides a wider variety of specialised tools adjusted to different functions within a team (architect, tester or database specialist). VSTS also offers a number of method templates: MSF for AGILE, MSF for CMMI and many more, developed by Microsoft partners. In addition, it is possible to adapt those templates to a given project. Application testing is also more important in VSTS. Load testing and web application tests can be performed. The environment offers creator tools assisting in the design of distributed applications and their validation with the use of typical project templates. It also enables the verification of products prior to entering them in the repository (usually with a comment, code description).

Borland Gauntlet is a tool used for the automation of the manufacturing and testing process in accordance with the AGILE method. It cooperates with version and configuration management systems and supports the control of code correctness before being entered in the repository. With this tool, project teams can monitor the quality of manufactured software and project managers can assess work progress and time required for completion on a current basis. In a software environment it is possible to integrate software, test and manage products. The manufacturing process can be evaluated by collecting intelligent metrics related to the activity of individual software developers and the results of their work. The system enables the use of the main panel for improving project development based on the updated (in real time) switchboard, which includes both current and historical data. It also provides for the use of an open plug-ins system, as well as their integration with other tools and systems used for software manufacturing through the API.

Detailed evaluation of both Gauntlet and VSTS tools showed their full fitness for supporting project management. Although each of the tools satisfied the functional criteria, the Jazz environment was chosen for the implementation of the model. It is presented below.

Jazz Environment Characteristics

Constructed with the use of the Eclipse platform, the Jazz environment includes the Rational Team Concert application (RTC), in which software management and manufacturing processes developed by distributed production teams are formalised.

The term: Eclipse *is a software development platform used for building applications in client-server architecture.*

IBM provides an integrated manufacturing environment for software written in the Java language, which is distributed together with the Eclipse platform. The environment is based on client-server architecture with the RTC application (IBM) as a commercial access client and the RTC server (based on the Apache Tomcat or Websphere architecture) [33].

The term: Apache Tomcat architecture *is a Web application server architecture created as part of the Apache project. It enables application start-up by using J2EE application server technology* (*Java Servlets and Java Server Pages*).

The term: Websphere architecture *is a software development platform provided by IBM, based on J2EE platform multi-layer architecture, where the presentation layer, the database engine and the resources may adjust the application scale to current business requirements.*

For the Jazz environment, a Jazz.net collaboration portal was created for the exchange of knowledge and information through a communicator and discussion groups. In this portal the documentation of the entire Jazz environment manufacturing process has been made available by IBM, presenting the history of past iterations and project stages. With this transparent approach to software manufacturing, knowledge and experience can be made accessible to both the creators of a system and their clients implementing the system. It is also possible to consider the installation of own plug-ins supporting the functionality of the environment for IT management. The idea called *Open Commercial Software Development* is, according to Danny Sabbah, General Manager at *IBM Rational Software*, "the next major innovation in collaborative engineering" [66].

The Jazz environment supports not only manufacturers but also manufacturing managers with regard to work organisation, role assignment and supervision of all project participants. It helps to collect data on manufacturing and project management. By identifying users who take part in the project we also define the teams which they belong to (*e.g.* user interface team, requirements team). An important advantage of the system is the support in work-related communication within teams (through *Jabber* or *IBM Lotus Sametime* communicators). Developers should be able to implement the *client* and the *RTC server* in teams working in Jazz without any major problems. RTC is an extension for the Eclipse platform which provides the functions of team work and project documentation. To make the usage more convenient, access to the Jazz software development environment is also possible from the web browser level via an Ajax-based interface. Methods of communication with the Jazz environment and of supporting teamwork are shown in Fig. 4.1.

The key feature of the presented method of collective software manufacturing is the variety of possible approaches to the management of manufacturing processes. The templates designed for supporting the AGILE and SCRUM lean methods provide the convenient selection of project templates for maturity expressed with the C-S-P capsule.

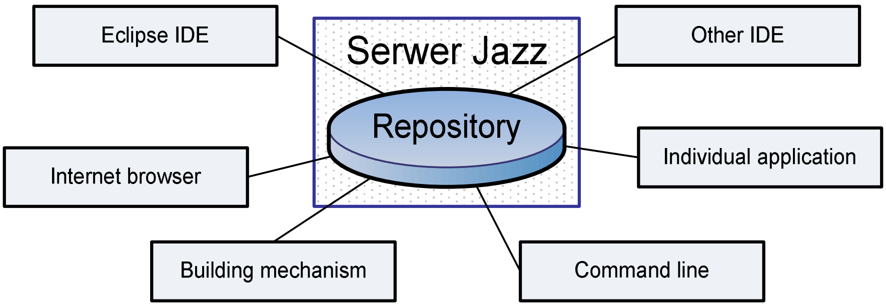


Fig. 4.1 Jazz environment infrastructure: methods of access to RTC Jazz server [54]

The environment in the standard version provides the following templates: Eclipse Way, OpenUP, SCRUM, Agile Process and Simple Team Process (editable to enable adaptation to own needs). So using a template does not mean that a certain way of acting is forced, but rather a set of rules that ensure compliance for that method. An approach to the use of templates within the Jazz environment is illustrated in Fig. 4.2. The diagram shows relationships between them (this is not the only possible set-up: depending on individual needs, the structure can be adapted to the project by adding new components). For the Jazz environment, components which support decision-making with respect to the selection of project management templates are available.

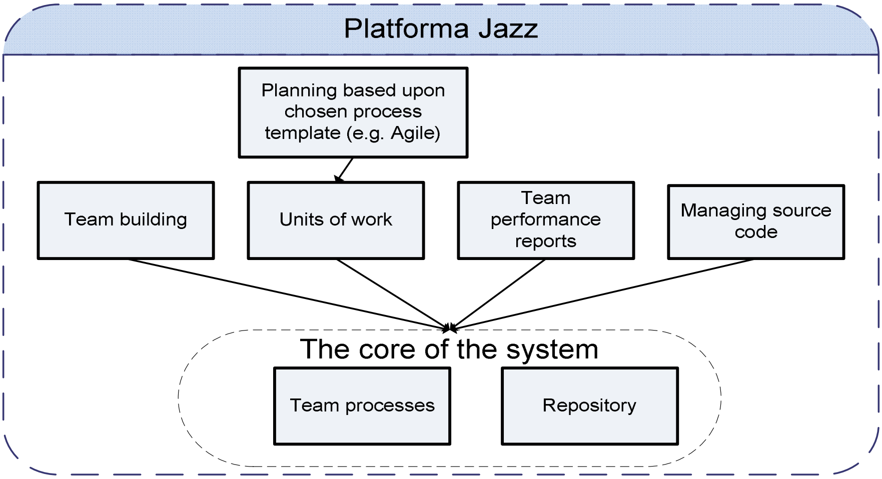


Fig. 4.2 Jazz platform components and relationships between them [54]

4.1.2 Design of Components for IT Selection for the Jazz Environment

Developing a design for Jazz environment components began with the specification of the MSM model described in Chapter 2. After that, description rules were developed including input and output variables. Below is an example of a set of model rules:

IF defined AND organic AND not suitable AND matching THEN lean AND Agile (4.1)

IF defined AND organic AND not suitable AND not matching THEN hard AND OpenUp (4.2)

IF defined AND organic AND suitable AND matching THEN lean AND Agile (4.3)

The description of the conditions applied clearly refers to the C-S-P maturity capsule.

The hypotheses included in rules (4.1), (4.2), (4.3) refer to the selection of a method and its specification. The inference described above was conducted on the basis of expert knowledge of IT project management.

After defining the rule structure, the implementation possibilities of the rule-based description for the selection of an IT project management template were assessed on the basis of the developed model. The RTC environment analysis indicates two possible methods for this:

- building an own plug-in
- adding a code ensuring the desired function to the program source code.

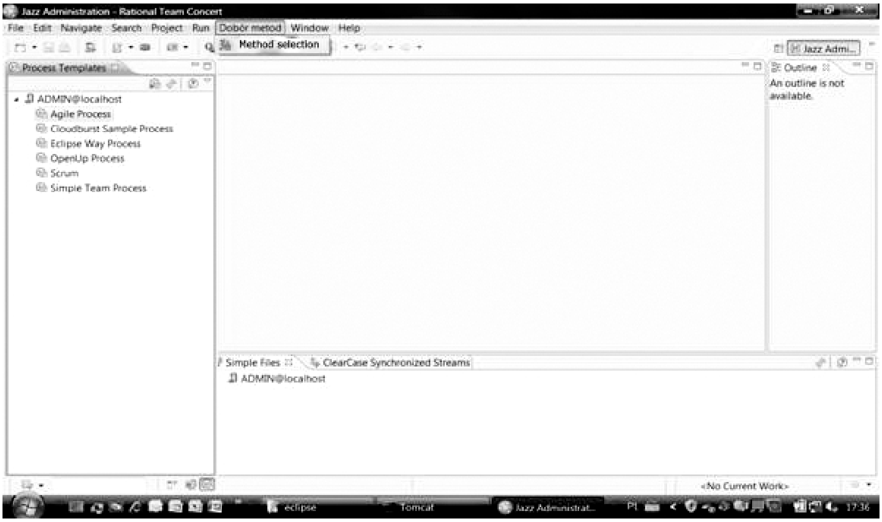


Fig. 4.3 Plug-in developed for the RTC environment

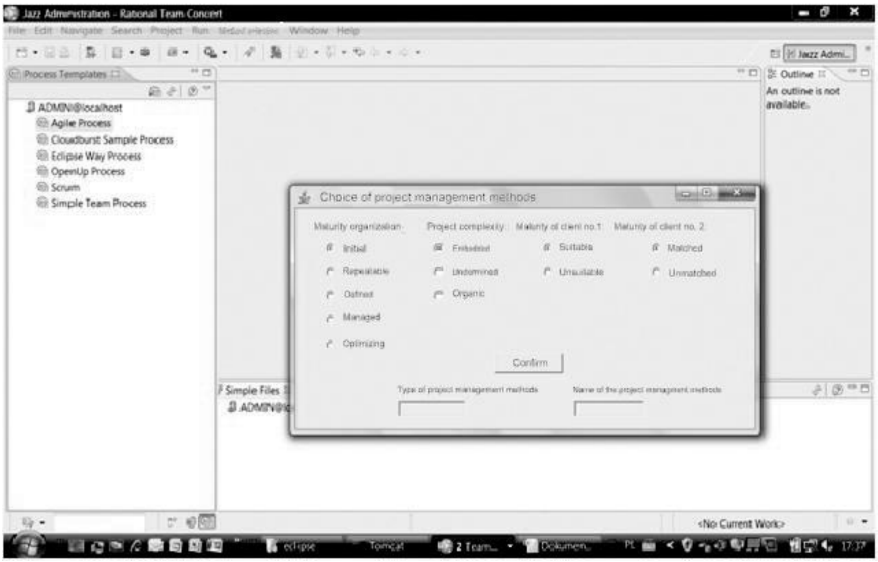


Fig. 4.4 Structure of an RTC plug-in based on the MSM model

First, an own plug-in was built with the use of the function made available on the level of the RTC user interface (Fig. 4.3).

The plug-in was built using access to the Eclipse platform and providing a source code based on the developed rule structure (see Chapter 2 for details) and their verification (examples of rules (4.1) – (4.3)). A drawback of this solution is access to the developed component from the main menu level (Fig. 4.3 and 4.4). The source code is presented in Fig. 4.5.

```
<?xml version="1.0" encoding="UTF-8" ?>
<?eclipse version="3.2"?>
<plugin>
<extension point="org.eclipse.ui.actionSets">
        <actionSet label="Select method" visible="true" id="selectmethod.actionSet">
                <menu label="Select method" id="MenuSelectMethod">
                                        <separator name="SelectMethodGroup" />
                        </menu>
  <action label="Select method" icon="icons/sample.gif"
    class="selectmethod.actions.SampleAction" tooltip="Select method."
                menubarPath=" MenuSelectMethod / SelectMethodGroup"
     toolbarPath=" SelectMethodGroup"
                id=" selectmethod.actions.SampleAction"/>
                   </actionSet>
         </extension>
</plugin>
```

Fig. 4.5 Source code for the plug-in for the selection of methods and IT management tools

In the plug-in, the Eclipse software development platform was used with PDE (*Plugin Development Environment*) version 3.3 installed. Extension points were also used with which the plug-ins enhance the functionality of the existing plug-ins (*e.g.* core plug-ins at the Eclipse level) and make the use of other plug-ins for functionality extension easier. The defined environment implements the concept of the ITM management model (MITM), where the user adds new functions (with the code generator) to the used tool as the project develops. Fig. 4.6 shows an example of opening class **window1** (line 28), which performs one of the functions in the model:

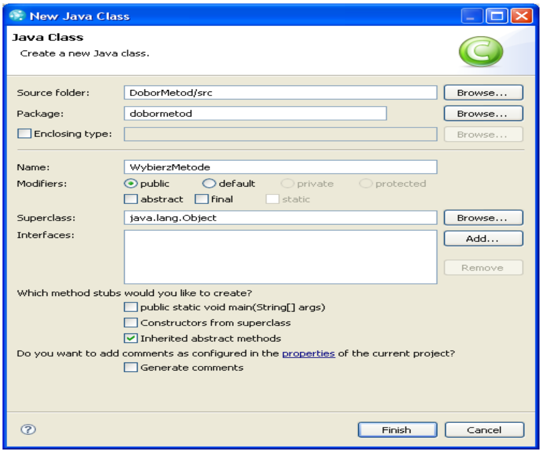


Fig. 4.6 Code skeleton generator

```
package selectmethod.actions;

import org.eclipse.jface.action.IAction;
import org.eclipse.jface.viewers.ISelection;
import org.eclipse.ui.IWorkbenchWindow;
import org.eclipse.ui.IWorkbenchWindowActionDelegate;
import javax.swing.*;

public class SampleAction implements IWorkbenchWindowActionDelegate
{

  public SampleAction() {
  }

  public void run(IAction action) {
  JFrame.setDefaultLookAndFeelDecorated(true);
  JDialog.setDefaultLookAndFeelDecorated(true);
    try
    {
    UIManager.setLookAndFeel("com.sun.java.swing.plaf.
    windows.WindowsLookAndFeel");
    }
    catch (Exception ex)
    {
    System.out.println("Failed loading L&F: ");
    System.out.println(ex);
    }
  new selectmethod.window1 ();
  }

  public void selectionChanged(IAction action, ISelection selection) {
  }

  public void dispose() {
  }

  public void init(IWorkbenchWindow window) {
  }
}
```

Due to limited possibilities for the implementation of a rule-based description through the standard functionality provided by plug-ins, the procedure of viewing the RTC source code was used as well as contact with users through a discussion forum and the team's own experience of the implementation of Java-based components. The RTC source code, however, proved to be complex and difficult to identify, and the search time required for the identification of the code-placing location was extensive. The search procedures applied were not effective and that is why – despite identifying the suitable location for placing the component – the plug-in was not added to the RTC code. Instead, the developed code was added to RTC in a standard way, *i.e.* through the main menu. The advisory component thus placed (based on the above-mentioned rule-based description) is shown in Fig. 4.7.

The solution, although very useful for an RTC user, has a disadvantage which consists in the need for control over the source code structure. The presented model assumes that (as the project develops) the project manager selects functions suitable for the project needs on a current basis. The approach is correct on condition that the user knows the environment applied and the source code. At the current stage in the development of IT environments and tools, especially OpenSource, this is possible. It is good to remember to use source-code search procedures with relevant (for the code complexity) comment-markers, which should be described and made available to users. Such practices are conducive to transparent software manufacturing.

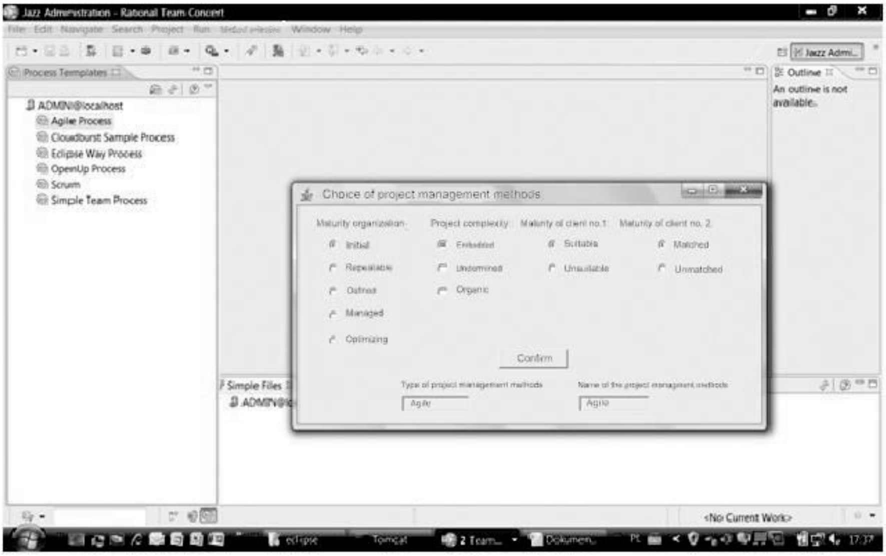


Fig. 4.7 Session with a plug-in developed for RTC needs

The implementation of components shows that the available environments for manufacturing and manufacturing management are equipped with the relevant functionality enabling the implementation of requirements defined by the user. Therefore, the proposed MSM model for the selection of methods and then tools and their functionalities proves to be effective in practical use. The innovative formal description constitutes a source of valuable knowledge for IT professionals who are serious about the evolution of IT tools during project development. It is evident that the analytical selection of a tool before it is implemented is essential. It is also recommended to use OpenSource environments, which can be freely extended. This policy is common practice in large IT companies. A good example is the parallel development of the Eclipse platform. On the one hand, the OpenSource environment is developed, and on the other hand, supporting plug-ins are manufactured. Supporting the development of such environments by using transparent manufacturing processes provides for the inclusion of user requirements both at the manufacturing stage and in the course of further development. That is why the proposed concept of the functional development of IT tools in the course of project development (*e.g.* in the form of plug-ins) conforms to the general strategy for IT product development.

4.2 MITM Verification – Support for Processes Ensuring Corporate Governance

A basic problem for many IT project managers is the right usage of Governance processes. In the case of enterprise architecture development, a clue can be found in the TOGAF standard. It includes an extensive description of architecture development processes and compliance analysis. For an experienced team leader, this can form a set of good practices.

Nevertheless, questions arise: apart from standard good practices for management, should Governance processes also be used as good practices in the organisation? Would it be a good idea, in this case, to use formal models? Can Governance processes be applied in implementing other systems in a company (not necessarily related to enterprise architecture, *e.g.* CRM)?

IT projects during their development are treated as ventures of growing negentropy (management and manufacturing processes are organised, also due to regular contact with the client). By assuming certain defined quantity measures for negentropy, the project manager gains the possibility of constantly monitoring the project status/negentropy and observing its changes. In this way, on the basis of project negentropy (maturity), it is easier to conduct Governance processes. But how should these processes be developed in order to effectively increase the negentropy value? The answer can be found in the translation of Governance processes into negentropy parameters: the width of the architecture repository, the length of the documentation catalogue and the height of the ITM section. To achieve this, Governance processes need to be presented and referred to the analysis of project negentropy coordinates (see Appendix 4).

Below, two main Governance processes are presented with reference to an IT project's negentropy, whose purpose is the development of EA in accordance with the ADM:

- adopting compliance measures for the development and implementation of architectures on the basis of (a_t, d_t, pr_t)
- monitoring the compliance with the use of project negentropy.

Architecture development processes, apart from the ones formally defined in the ADM, should not include other processes, *i.e.* those that extend beyond the definition of negentropy (Fig. 4.8). The material and drawings used in the verification process [63], presented in this sub-section, were created on the basis of the TOGAF framework.

After Governance processes are equipped with the innovative tool of project negentropy, the main task is to adopt the terminology and conditions for conformance. This should occur between the project negentropy level and the actual degree of its development. It needs to be kept in mind that the notion of 'conformance' can be understood differently in diverse organisations. For the purposes of the verification of the usefulness of negentropy in IT project management, the concept of conformance, which includes six levels referring to linguistic measures (null, very low, low, medium, high, very high), will be used in the analysis of Governance processes for the purposes of the ADM (Fig. 4.8).

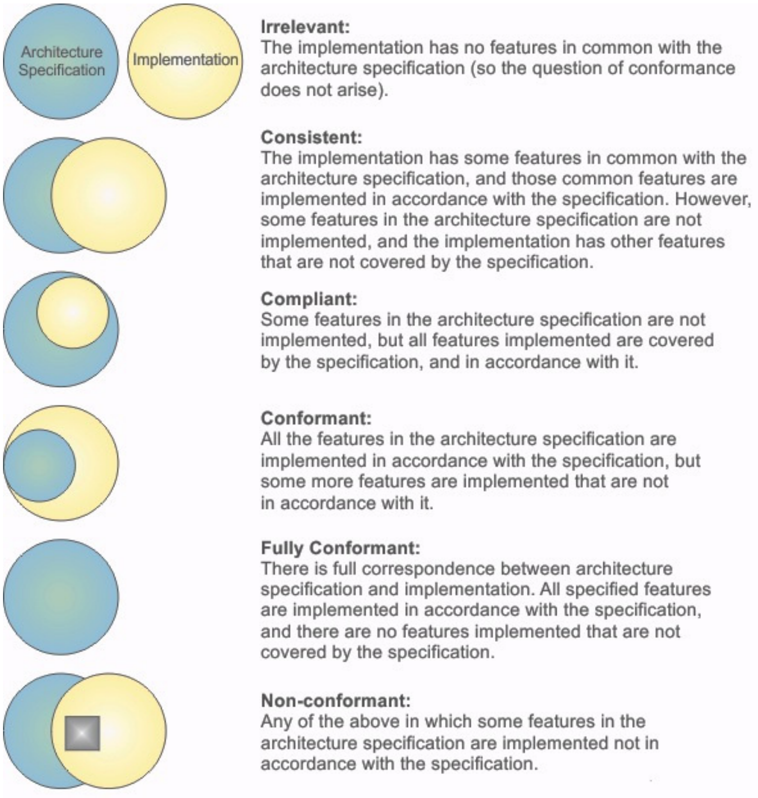


Fig. 4.8 Conformance levels

Hereafter we will focus on the verification of the applicability of the developed model for the purposes of Governance processes. The verification process will be divided into two stages.

Stage one will include an introduction of the specification of IT project negentropy measures for Governance processes. In stage two, we will verify the applicability of those measures with respect to two different uses of Governance processes in:

- the design of enterprise architecture compliant with the ADM
- implementation of the CRM system for the purposes of an IT organisation.

Both cases require that the project environment and the method of the implementation of negentropy measures be defined for the purposes of Governance processes, and that the applicability of our solution for ensuring corporate governance of a company be specified.

4.2.1 *Verification of Governance Processes in Developing Enterprise Architecture*

Governance processes will be verified for the development of enterprise architecture on the basis of TOGAF. In accordance with the specification (point 3.3.2), the description of project negentropy includes the width, length and height variables (a_t, d_t, pr_t).

The Width of the Project Repository

Table 4.1 Evaluation of the 'width' of the primary negentropy of ADM enterprise architecture development for the purposes of Governance processes

Governance process	Negentropy measure for architecture development
Analysis of the conformity of the business objectives and application deployment	Conformity of the business objectives of the project and application deployment. Conformity of services required for the implementation of the project.
Mapping for target architecture purposes	Conformity of the target architecture domain functionality and project scope. Conformity of the system scheme and its division into logical partitions, physical topologies.
Review of project management processes	Conformity of milestones and the time schedule with the progress of the project. Conformity of assigned roles and tasks with the progress of the project.
Future implementation	Conformity of future directions in project implementation with good practices.
References	Conformity of documents describing implementation details with the progress of the project.

Tables 4.1 and 4.2 present a method for the estimation of the primary negentropy of development with respect to the width of the enterprise architecture repository for the ADM and Continuum evaluated on a six-level scale (null, very low, low, medium, high, very high).

Table 4.2 Evaluation of the 'width' of the primary negentropy of Continuum enterprise architecture development for the purposes of *Governance* processes

Governance process	Negentropy measure for architecture development
Analysis of the conformity of the business objectives and application deployment	Conformity of the plaster project with its use in an EA project. A plaster project is a relevant subset of EA projects saved in the Continuum.
Mapping for architecture purposes	Conformity of the target architecture domain functionality and project scope on the basis of the Continuum resources. Conformity of the system scheme and its division into logical partitions, physical topologies on the basis of the Continuum resources.
Review of project management processes	Conformity of milestones and the time schedule with the progress of the project on the basis of the Continuum resources. Conformity of assigned roles and tasks with the progress of the project on the basis of the Continuum resources.
Future implementation	Level of the conformity of future directions in project implementation with good practices with this respect on the basis of the Continuum resources.
References	Level of the conformity of project documentation describing implementation details with the progress of the project on the basis of the Continuum resources.

The Length of the Project Documents Catalogue

Estimation of the degree of documentation development is conducted in a similar way as the evaluation of the architecture development level. The first evaluation is related to the need for using the project Continuum and, on that basis, the examination of the repository of compliance processes. Opinions on compliance are required here, prepared on the basis of the control of the compliance of development processes in a given architecture project with the processes described in the Continuum, with respect to the established criteria, adopted strategy and business objectives. The formal preparation of such an opinion is usually the main element in the strategy of constructing a company's architecture. Therefore, for the evaluation of manufacturing documents and project negentropy, typical measures were adopted: null, very low, low, medium, high, very high. The measures are assigned in accordance with the following procedure:

- Preparation of a strategy document for the cooperation of architecture development teams and resource sharing: two measures were applied here: very high in the case where the document is available, and very low in the case of a lack of the document.
- Opinions related to the conformity of functional architecture: five measures were adopted here: very high and high in the case where the conformity document is available, very low and low in partial cases (functional conformity for one or several architectures) and null if there is no such document.

- Opinions on the compliance of the developed architecture with business needs: here, four measures were applied: very high and high if the opinion is available, and very low and low if there is no opinion or it has been partly prepared, considering:
  - ➢ conformity of the objectives of the developed IT project with the company's architecture (in order to guarantee that IT projects are adapted to business objectives)
  - ➢ deep investigation of the technical infrastructure (problem with the identification of the company's main activity)
  - ➢ conformity of company operation with the developed architecture (aiming for changes in key areas and indicating major hazards).

The Height of the ITM Section

Similarly to the evaluation of the length of the documents catalogue, the height of the ITM section was established in consideration of the progress of management processes with regard to the following elements:

(a) high level:

- ensuring the use of best practices in architecture development
- performing a review of compliance with the company's architecture (based on standards)
- indicating the location for the modification of standards
- identifying services or specific applications used in the company
- applying new technology solutions available owing to technology development
- communication management for obtaining the readiness of technology for the project

(b) medium level:

- identifying key criteria for orders for components and other products (*e.g.* incorporating COTS components and product documentation)

(c) low level:

- identifying problems of architecture development and communication between providers and clients in the case of considerable differences between the assumed and actual level of the developed architectures

(d) very low level:

- review of the compliance of architecture development processes with the ADM and selection of alternative solutions for the architecture in the case of no compliance
- CIO failing to use the results of a compliance analysis of architecture development with the ADM while making decisions related to further development processes.

Despite continuous attention to the compliance of architecture development processes with ADM processes, it is a good idea to prepare and implement a procedure for detecting nonconformities in this regard, particularly inconsistent areas in the company. Such a procedure, supported with an apt time schedule, is important when considering constant changes in architecture development. Its implementation increases the negentropy value of a project.

Scenarios for Governance Processes Controlled by Project Negentropy

During the implementation of Governance processes, the possible scenarios should be considered, which will affect the method of conformance testing, and thus also the identification of negentropy factors. It is considered that the following scenarios are possible:

- For smaller-scale projects, the review could take the form of a series of questions posed by the project architect or manager (using checklists) and include a digest of responses. This is usually taken into account in the overall IT management policy for the entire company. If the questions refer to the level of architecture development, document creation and the implementation of manufacturing and management, the conformance testing can be very formal and can use the measures adopted for project negentropy.
- In the case of larger-scale projects, the aim of the review is usually to increase knowledge of enterprise architecture functions. Such actions are intended to organise, manage and also conduct a review with the participation of business experts. For this purposes the concept of project negentropy can also be applied.
- During a review, in most cases, especially in larger-scale projects, the functionality of EA is analysed in detail. Usually, the lead architect coordinates meetings of business teams and technical domain experts in order to conduct the review using questions. These can relate to the evaluation of domains by business and technical experts. On a similar basis, the review can be formally performed in relation to the measures of project negentropy.

In all cases, the process of an architecture compliance review requires backing by senior management, with authorised personnel assigned. For large-scale projects developed in a company, the CIO or another architecture development unit is usually authorised to assess the architecture level within subsequent annual reviews. The assessment is constructive if it includes the level of architecture development, document preparation and the implementation of manufacturing and management processes (as per the project negentropy measures). An example of a standard architecture compliance review in accordance with the ADM and TOGAF is presented below (see also Fig. 4.9). It does not include our negentropy concept (the width of the architecture repository, length of the documents catalogue and height of the ITM section: a_t , d_t , pr_t). Naturally, consideration of the category of project negentropy (with the observation of management processes, documents and architectures) would make the demonstrated compliance procedures easier to analyse by the CIO.

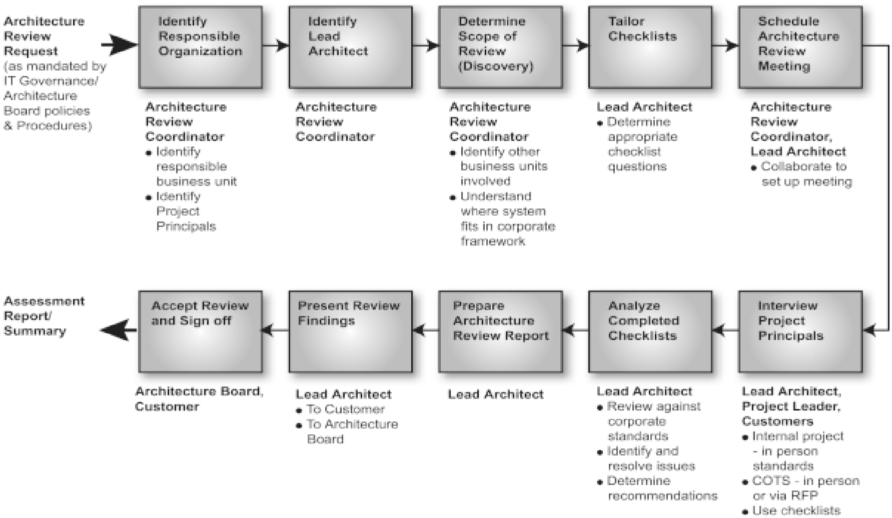


Fig. 4.9 Procedures for a compliance review of the architecture development process (ADM)

Competence Questions for the Evaluation of Project Negentropy

For the evaluation of the project negentropy value, a set of competence questions is suggested (as in the COBIT standard) which are used to prepare an opinion on architecture compliance. The questions refer to the level of knowledge (of project negentropy) in the provider organisation with respect to software engineering, information management, security and management systems. The checklist is based on materials provided by members of The Open Group. Organisations can also use their own questions. They should include a brief description of architecture building principles (the reason for the questions) and a description of what is expected from the answers. The checklist presented below has been compiled for various projects and adapted to the individual needs of an examined company related to the project negentropy testing discussed in the following Section (4.3.2).

Competence questions – Control of the width of the architecture repository

- What is the approach to the project life cycle?
- At what stage is the project in its life cycle?
- Which crucial project issues have been identified or analysed (evaluation of hardware, operating systems for networks, servers and end-user devices)?
- Which functions of the developed system require high-volume and/or high-frequency data transfer?
- How does the device design system work and does it allow end-users to be involved in the process?
- How many data processing users use the software locally and globally?
- What are the conclusions for the project based on the affinity of data, applications, *etc*.? To what extent is the project data similar?

- Was the hardware and operating system chosen before the development of key elements of the system?
- If hardware and operating system decisions were made outside the project control:
    - Does project awareness justify those decisions?
    - How do those decisions affect the framework of the project?
- If certain architecture development standards have not been applied:
    - What are the basic technical and business requirements in the case of applying those standards (with the use of corporate standards)?
    - Are they supported by the company?
    - Have business activity assumptions been subject to control?
- What is the process for the full evaluation of life cycle costs of hardware and operating systems?
- How has corporate financial management been involved in the evaluation of life cycle costs?
- Has the provider's financial analysis been performed?
- Has a commitment been made to select a particular provider?
- Can the requirements be fulfilled by only one provider?

Competence questions – Control of the length of the documents catalogue

- Describe the error which results from the incorrect operation of the application module.
- Describe the overall application structure and methods of defining and placing modules.
- Give a general description of the defined parameters for patterns and methods.
- Describe the approach used for minimising the number of iterations between the client and server, particularly for complex data structures.
- Describe the main data structures and communication protocols used in operation between major system components.
- Describe the extent to which the system was developed and saved on the basis of the component level and status.
- Describe the extent to which objects are created, used and removed.
- Describe the extent to which the system is based on threads.
- Describe the approach used for preparing internal documentation.
- Describe the review process of the code used to build the system.
- Describe unit testing used for testing individual system components.
- Describe the pre- and post-condition testing in various system modules.
- Demonstrate the support for all types of software connection interface or for other components (language bindings and other forms). Have those actions been organised?
- Describe the extent to which data formats need to be handled in various software development environments.

- Describe what tools and processes have been used to test the system memory, computing power and operation security.
- Describe the layering of system services. Describe the general number of links between major system components. Does the main system consist of many point-to-point applications or interfaces, or is it a distributed system?
- Describe to what extent system components are connected.
- Which system requirements do not need to be considered with respect to library building, system monitoring, load balancing, communication protocol support, transaction processing, naming services (infrastructure)?
- Describe how the system and its components are designed for *refactoring*.

Competence questions – ITM section control

- Are there possibilities for using standard products (for application sharing, videoconferences, email, work flow management, document publishing, software management, project management)?
- Describe the possibilities for using business requirements for company infrastructure which are not fulfilled by standard products.
- Do any of the requirements met by standard products support one or more line-of-business applications? For example:
  - business applications related to purchase or sales and marketing
  - application manufacturing engineering
  - provider applications management
  - production planning applications
  - client support applications
  - financial applications, HR management applications and service applications
  - other IT systems applications in companies
  - business processes engineering.
- Are the requirement engineering processes for business application building supported by standard products?
- Is the integration of points (business/activity processes, applications, data, computer environment) based on previously built architecture?
- Will integration techniques be used (shared use of objects, standard data definitions for the manufacturing of a shared user interface for the presentation)?

4.2.2 Verification of Governance Processes for the Implementation of an IT System

Below is an example of the verification of project negentropy specification processes for the purposes of a client organisation implementing the CRM system. The organisation is focused on providing complex IT solutions including hardware and software technologies, and has a well developed service department. With approximately 100 employees, it belongs to the *Small and Medium-sized Enterprises* sector (SME). It is a mature organisation, with good knowledge of the IT market, and is implementing the CRM system to ensure permanent contact with clients.

The analysis of project negentropy changes, including: architecture development, documents and production (manufacturing and management) processes for architecture and documentation development, throughout the duration of the project is good for the IT system implementation.

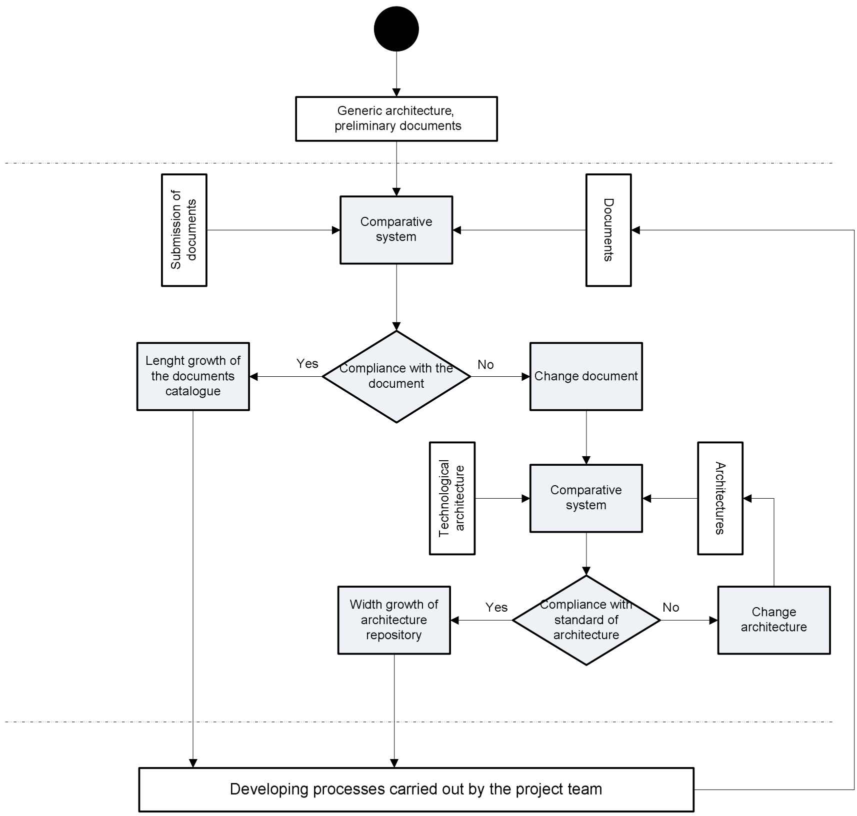


Fig. 4.10 Governance processes controlled by project negentropy

After establishing the project requirements, the identification and evaluation of Governance processes was performed for the negentropy component categories. Fig. 4.10 shows the characteristics of Governance processes controlled by project negentropy. This scheme was applied in the examined company to recognise, organise and limit the implementation risk for design processes. This was possible since the project was considered in terms of project negentropy, to which Governance processes were referred, conducted on two levels of comparative systems: compliance of the prepared documents with transfer documents (as per the project negentropy) and compliance with the technology architecture (also in accordance with the project negentropy) – for details see Tables 4.3, 4.4 and 4.5.

The Width of the Project Architecture Repository

Bearing in mind that in our approach the ADM development process is treated as a process of low negentropy (at level 1, 2) and the Continuum-based processes are considered as no higher than medium (at level 1, 2 or 3), it was proposed that the development of enterprise architecture should be based on the Continuum, which includes a collection of architectures, documents and processes which may be used in development. To this end, the Continuum was built and architectures (based on standard functions) of eight system types were created and incorporated in its resources: CRM, SRM, SCM, ERP, Human resources and payroll, Finance and accounting, BI, and a stock module. Fig. 4.11 presents the specification of those architectures grouped in a spreadsheet. The selection of the specification environment was dictated by the need to share the architecture resources on a platform that is easy to use for the client and the provider. The results shown here are based on studies [1, 3, 9, 19] which present the team's own work.

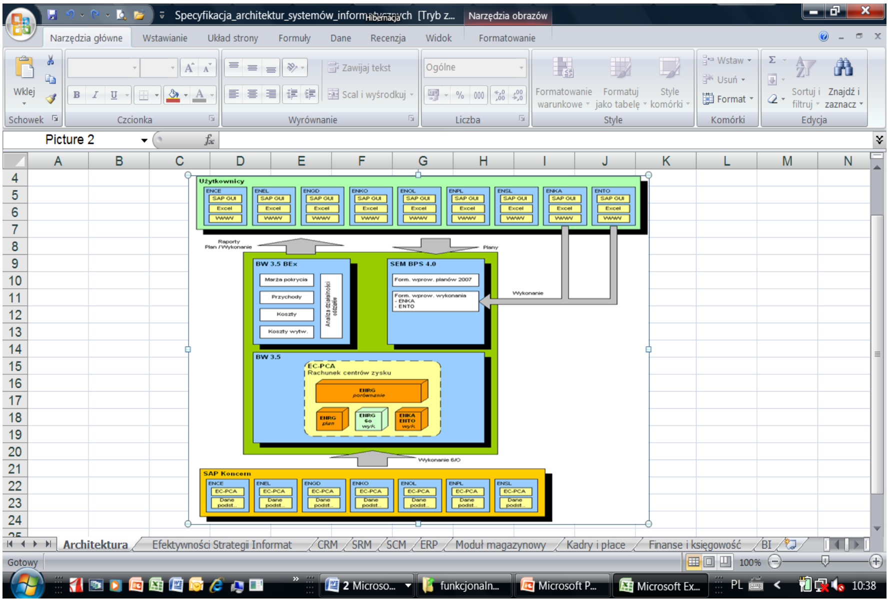


Fig. 4.11 Illustration of a system architecture (implemented in Polish)

The Length of the Catalogue of Project Documents

The next step in the architecture development process (in accordance with the project negentropy principle) was the preparation of three model documents (initial document, project document and transfer document), corresponding to the areas of changes in the company (Table 4.3, 4.4, 4.5), where the concept of pentavalent (1, 2, 3, 4, 5) negentropy was used.

Table 4.3 Initial documents

Area for proposed changes in enterprise architecture	**Documents for architecture changes: their measures and negentropy**
Enterprise structure, within the architecture vision	Support document for all units in the organisation engaged in the process of the preparation and analysis of budget implementation.
Business processes within the business architecture	Support document for the budgeting process with respect to the collection and analysis of data of plan execution.
Information scope within the information systems architecture	Cost form including analysis. Cost form for product manufacturing for own needs including analysis. Income form including analysis. Contribution margins including activity analysis.
Application functions within data systems architecture	Document for the data load procedure. Document for the procedures of automatic data acquisition of plan execution for company branches. Document for the procedures of manual data entry of plan execution for company branches.
Technology used within technology architecture	BW 3.5 – Business Warehouse. SEM-BPS 4.0 - Business Planning and Simulation.

Table 4.4 Project documents

Area for proposed changes in enterprise architecture	**Documents for architecture changes: their measures and negentropy**
Enterprise structure, within the architecture vision	Support document for all units in the organisation. Support of the budgeting process with respect to plan preparation. Coordination document for controlling services.
Business processes within the business architecture	Support document for the budgeting process with respect to collecting and analysing data of plan execution.
Information scope within the information systems architecture	Cost form (year-over-year comparison). Cost form for product manufacturing for own needs (year-over-year comparison). Income form (year-over-year). Contribution margins (year-over-year).
Application functions within data systems architecture	Document for the set of planning functions for plan preparation. Document for the control and monitoring of the budget preparation process. Document for the set of coordination functions for controlling services activity.
Technology used within technology architecture	Growth / migration. BI 7.0 – Business Intelligence. SEM-BPS 6.0- Business Planning and Simulation. EP 7.0 – Enterprise Portal. XI 7.0 – Exchange Infrastructure.

Table 4.5 Transfer documents

Area for proposed changes in enterprise architecture	Documents for architecture changes: their measures and negentropy
Enterprise structure, within the architecture vision	Support document for all units in the organisation. Support of the budgeting process with respect to plan preparation. Coordination document for controlling services.
Business processes within the business architecture	Support document for the budgeting process with respect to collecting and analysing data of plan execution.
Information scope within the information systems architecture	Technical department document: statistical indicators analysis. Sales department document: statistical indicators analysis. Accounting department document: statistical indicators analysis. Executive manager department document: statistical indicators analysis. Documents for the evaluation of own transport costs.
Application functions within data systems architecture	Document for the set of functions: automatic data acquisition of plan execution for company branches.
Technology used within technology architecture	Documentation of usage: BI 7.0 – Business Intelligence. SEM-BPS 6.0 – Business Planning and Simulation. EP 7.0 – Enterprise Portal. XI 7.0 – Exchange Infrastructure.

The Height of the ITM Section

The next step was to evaluate the height of the ITM section, which was based on the competence questions referring to document preparation and the implementation of management and manufacturing processes. Examples of questions together with a linguistic evaluation (in accordance with the proposed negentropy concept) are presented in Figs. 4.12, 4.13, 4.14, 4.15. The assessment was conducted for the implemented CRM system undergoing integration with other systems in the enterprise. The prepared application included 54 questions divided into four areas of IT system functionality: sales, customer service, marketing and services.

For each area, an investigation questionnaire was prepared with a list of competence questions. A four-degree scale was adopted for the quality evaluation of architecture development, document creation and the level of implementation of manufacturing and management processes. The evaluation of each function was consulted with enterprise representatives. Table 4.6 presents the rough and linguistic evaluations of system functionality.

Each of the specified system functionalities was linked to its significance to the organisation (Table 4.7), and levels of certainty were assigned to the evaluations (Table 4.8).

Table 4.6 System functionality evaluation

Linguistic value	Rough value
FAILS TO FULFIL	**1**
UNSATISFACTORY	**2**
SATISFACTORY	**3**
COMPLETELY FULFILS	**4**

Table 4.7 Weights used for functionality analysis

Weight	Weight significance
1	LOW SIGNIFICANCE
2	MEDIUM SIGNIFICANCE
3	HIGH SIGNIFICANCE

Table 4.8 Degrees of evaluation certainty

Degree of evaluation certainty	Meaning
1	CERTAIN
0,5	MEDIUM CERTAINTY
0,1	NOT CERTAIN

No.	Question					Certainty	Result
	Functionality: Sales						
1	Are mail, calendar, tasks and contacts from f. ex. Outlook synchronized with the CRM database?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
2	Is the client data complete?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
3	Is the data of potential clients managed and forwarded to adequate persons?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
4	Is sales probability managed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
5	Is the sales process managed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
6	Is the product catalogue existent and in use?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
7	Are offers and orders managed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
8	Are financial applications integrated and can receipt data be automatically entered into the system?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
9	Are sales norms existent?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
10	Is there a possibility of creating areas of action and service for certain salespersons, which would allow management and review of sales?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
11	Is event history in the sales cycle being registered?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
12	Are break even analyses being conducted and used?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
13	Is receipt history being kept for the client?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
14	Are the competition's actions being watched?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
15	Is there a registry of reasons of an eventual loss?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1

Fig. 4.12 Example questionnaire with competence questions for the evaluation of project negentropy while implementing the CRM system (functionality – sales)

	Functionality: Customer Service					Certainty	Result
16	Have client groups been defined?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
17	Are clients being segregated in-group?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
18	Are sought clients being assigned to groups?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
19	Is information regarding brands and patrons edited en masse?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
20	Are services available on site?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
21	Is online self-service available?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
22	Are phone sales conducted?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
23	Are mail, calendar, tasks and contacts automatically synchronized with the CRM database?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
24	Are reports created and assigned in the customer service process, as well as managed until completion?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
25	Is there a base of pre-existent queries available in customer service, online or through mail?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
26	Are reports being queued?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
27	Are certain reports automatically forwarded to the right worker in order to resolve them easily?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
28	Is a knowledge base with a search option existent and in use?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
29	Is there a functional product catalogue that includes complex price levels, metric units, price drops and payment options?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
30	Is contract management existent?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
31	Is client communication via e-mail being automatically monitored?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
32	Are there report tools that allow to point out typical problems, rate client needs, monitor them and measure efficiency?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
33	Is there an option of generating offers in different file formats?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2

Fig. 4.13 Competence questions for the evaluation of CRM negentropy (customer service)

	Functionality: Customer Service					Certainty	Result
16	Have client groups been defined?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
17	Are clients being segregated in-group?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
18	Are sought clients being assigned to groups?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
19	Is information regarding brands and patrons edited en masse?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
20	Are services available on site?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
21	Is online self-service available?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
22	Are phone sales conducted?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
23	Are mail, calendar, tasks and contacts automatically synchronized with the CRM database?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
24	Are reports created and assigned in the customer service process, as well as managed until completion?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
25	Is there a base of pre-existent queries available in customer service, online or through mail?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
26	Are reports being queued?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
27	Are certain reports automatically forwarded to the right worker in order to resolve them easily?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
28	Is a knowledge base with a search option existent and in use?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
29	Is there a functional product catalogue that includes complex price levels, metric units, price drops and payment options?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2
30	Is contract management existent?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	1
31	Is client communication via e-mail being automatically monitored?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	3
32	Are there report tools that allow to point out typical problems, rate client needs, monitor them and measure efficiency?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	0
33	Is there an option of generating offers in different file formats?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	2

Fig. 4.14 Questionnaire for finding the negentropy while implementing the CRM (marketing)

	Functionality: Service					Certainty	Result
55	Can service orders regarding warranty be managed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
56	Can service orders linked to previously defined contract clauses be managed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
57	Can service actions be planned?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
58	Can service department costs be reported and controlled?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
59	Can worker efficiency and machine sales be rated?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
60	Is there an integrated client relations platform?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
61	Are client interactions multi-channel?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
62	Does the system provide management, tracking and solution of issues?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
63	Does the system provide a knowledge base?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
64	Can contracts be managed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
65	Can the service team be managed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
66	Does the system allow quality analysis?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
67	Does the system support quality management?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	
68	Does the system support the acquiring of orders and leads?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1	

Fig. 4.15 Competence questions for the evaluation of negentropy for the CRM (services)

Linguistic Analysis of Responses to Competence Questions

By the conducted functionality evaluation, responses were obtained to competence questions related to all functional areas of the IT system. Above, we present a method for the evaluation of the 'sales' module, which consists of fifteen questions related to specific CRM system functions, whose weights were established by the sales department. Answers to five questions were considered the most important (Tables 4.6-7: 'completely fulfils' weighted 'high significance'). According to a sales department employee in the client organisation, the following CRM functionalities are also indispensable: 'email synchronisation', 'calendar', 'tasks and links with CRM base' (in Tables 4.6-7: graded as 'satisfactory' weighted 'medium significance').

For the purposes of the CRM system, an application needs to be created (in the course of the project) in the Lotus Notes platform which will include email and calendar functions as well as client information. In the analysed company, the functioning system fulfils the client's requirements concerning the CRM only to a satisfactory level. In the company system there is no information related to payments (Tables 4.6-7: 'satisfactory' weighted 'medium significance'). Thus an additional system, CDM Optima, is used. The potential recipients database is fully usable, and the collected information is passed on to the persons responsible for initiating contact (Tables 4.6-7: 'satisfactory' – 'medium significance'). From the company's point of view, sales management is an essential functionality. In the CRM system it is used to a satisfactory level – stages in the process are automated (Tables 4.6 and 4.7: 'satisfactory' – 'medium significance').

4.3 MITM Proof – Support for the Evolution of the Provider Organisation

The third level of model verification was its application in the processes of the provider organisation's evolution. The verification processes presented below were limited to organisations offering IT support (representative for large organisations), rendering services for an internal unit (within the structures of an organisation which has such a supporting unit).

It must be noted that cooperation between the IT support organisation and the parent organisation occurs according to the conditions for providing such services. The conditions are specified in an SLA agreement, which is subject to negotiation. They are defined in a catalogue of services, according to the description in the Service Level Requirements document.

In view of the above, it was suggested that the evaluation method of organisation maturity and the developed model should be verified. The internal IT support organisation in a bank was considered. Verification tests were conducted in three stages [4]. The first stage was related to adopting the evolution concept based on the ITIL standard ver. 2. The aim of stage two was to organise the development processes of two departments in the supporting organisation: Service Support and Service Delivery. Stage three was intended to apply the developed model in the organisation's development forecast. Six experiments were presented for the design of possible development scenarios for the purposes of the CIO.

4.3.1 Concept of Evolution Processes

Examination of the organisation (by its employees and experts) and the analysis of the ITIL standard showed [4] that for the quantitative (in line with ITIL) evaluation of maturity the 'most crucial' processes and evolution factors must be chosen, using the knowledge of the organisation and the fact that the aim of the assessment is evolution. Although the ITIL standard shows the frame for the evaluation of the state of the organisation (the maturity of its processes), it does not allow for the classification of factors which cause changes in the state of the organisation.

While selecting the processes to qualify the provider organisation, the evaluators paid attention to those which enable moving from one maturity level to another, and classified them as *transition processes*. Thus a list of processes called 'components of the effective organisation evolution' [4] was defined. It was found that, while planning changes of a global nature, for each organisation it is important that manufacturing and management processes which refer to the following components are implemented:

- service design and implementation processes
- projects whose aim is service design and implementation
- manufactured and maintained services
- knowledge acquisition and processing
- technologies used
- the culture of the organisation manufacturing and providing services.

The transition processes, though extracted for the purposes of the evaluation, cannot be considered separately. Yet the subject of this work is not the evaluation of an organisation's evolution, but rather a demonstration of the applicability of the developed model in the evolution process. The idea of the evolution of a supporting organisation as presented in this Section (although restricted to one component of the evolution, manufacturing and maintaining services, it can be easily applied to other components) is based on the following assumptions:

- the basis for evolution is the running of key processes as per the ITIL standard ver. 2
- the processes have a hierarchy of maturity compliant with the ITIL ver. 2 (Table 4.9)
- the evaluation of the processes is linguistic (Fig. 4.16)
- the forecast model is based on linguistic rules of experts in the area.

In Table 4.9 we present the hierarchy of process maturity for the Service Support and Service Delivery processes in the examined supporting organisation.

Table 4.9 Hierarchy of Service Support and Service Delivery processes

Process name		**ID**
IT-Client obligations	-	P1
User support	-	P2
SLAs for systems	-	P3
Client needs management	-	P4
SLM processes	-	P5
AvM processes	-	P6
CAPM processes	-	P7
IT section organisation	-	P8
Service Desk	-	P9
Service Support processes	-	P10
Persons of relevant education	-	P11
Closing out SLAs	-	P12
Closing out OLAs	-	P13
Cost of service	-	P14
Service implemented and modified through projects	-	P15
Support processes	-	P16

Linguistic values were assigned to the processes. Fig. 4.16 presents an example diagram (based on the maturity of manufacturing organisations), which describes the relationship between the organisation providing IT support (IT) and the client organisation (Client) with the use of the values: ‘none’, ‘part’, ‘full’.

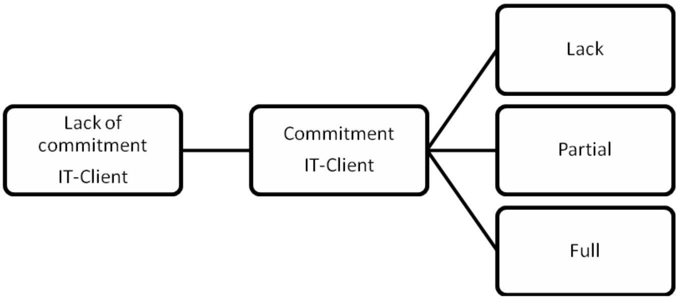


Fig. 4.16 Example of the linguistic description for the process 'no IT-Client obligations'

Based on the linguistic description of the process 'no IT-Client obligations' (Fig. 4.16), rules for the model of the forecast of organisational evolution were developed [4], which take into account the maturity level of the provider (compliant with the maturity presented in the book) by analysing progress related to the possibility for a move to a higher maturity level (Section 3.5).

IF <initial level> AND <change mechanisms> THEN <transition processes> (4.4)

An example of changes of obligations between the IT support organisation and the client over time is presented in Fig. 4.17. At the start of the cooperation between IT and Client, there are no mutual obligations in the process, which need to be established as the cooperation develops. It is indispensable that the obligations are approved by the management, which leads to the conclusion of a pilot SLA agreement (for a part of the organisation). The process of drafting agreements is spread over time, which in Fig. 4.17 is shown with arrows (indicating an increase in the documentation, with the drafted agreement as the final result).

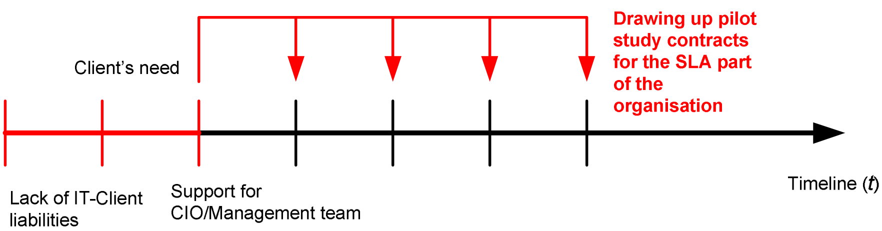


Fig. 4.17 Examplary evolution of obligations between IT support and the client organisations

4.3.2 *Diagnosing the Level of Processes: Service Support and Service Delivery*

The rule-based description presented above was used for diagnosing the quality/level of 'Service Support' and 'Service Delivery' processes (Table 4.10) with the use of a linguistic evaluation.

Very Low Management Level. At this level there are no agreements between the IT organisation and the business client. In the analysed case, SLAs were not defined. Therefore there was no settlement for manufacturing processes or service maintenance. There was also no support system for the manufacturing and maintenance of services by the user.

Low Management Level. A low management level means the signing of initial SLAs and their settlement and the use of measures to take into account the client's needs and service availability as a percentage and per hour. A reporting system was created and a method was developed for measuring the client's needs, service availability and communication level and its evaluation. The use of measures and reporting has increased the quality of customer service and provided the data necessary for the management of service quality. In the organisation, a process has been defined within the ITIL framework and a decision has been made regarding its implementation. Incident management has been based on ITIL ver. 2.

Medium Management Level. A medium management level is based on the establishment of an organisation (instead of an IT section). In the transition process (change of the IT section organisation) an ICT branch (*Information and Communication Technologies)* was also established, as well as a Service Provision Department (SP). After the reorganisation of 8 teams, 17 teams were created and the personnel increased from 44 to 140. For each team:

- the mission and strategic aims were specified
- hunting ground boundaries were defined for individual functions in the organisation
- scopes of responsibility were defined for employees as well as areas outside those scopes
- segregation of duties was established (as per RACI)
- roles and position profiles in the organisation were defined.

For three areas of service life: planning, development and manufacturing, a team was established which helped to systematise the process of IT infrastructure development. One of the objectives was also to formally define the links between the infrastructure and its environment and the incorporation of ICT into the implementation process of new services. In the operational processes of service manufacturing, a quantification of development, planning and manufacturing processes was used as well as IT standards and optimisation measures.

Table 4.10 Example of the linguistic evaluation of the level of Service Support and Service Delivery processes for an organisation providing IT support for a bank

Level	Level	Substantiation
1	Very low	Backing of CIO/management for the approach based on services
1	Very low	Defining client needs
1	Very low	Management of client needs based on system application
2	Low	Availability reports for systems in the IT support organisation
2	Low	Preparing SLAs for systems
2	Low	ITIL training for employees
2	Low	Establishing the configuration management process
2	Low	Establishing the Service Desk and Service Support processes
2	Low	Organisational changes
2	Low	Establishing the Service Delivery
3	Medium	Application of management methods for an IT project
3	Medium	Approval of the service-based model by a business partner
3	Medium	Creating a support tool for management processes
4	High	Service support processes provide for a service-based approach
4	High	SLA pilot for service design and implementation
4	High	Preparation of a management report for service quality
4	High	Implementation of operational rhythm for tactical processes
4	High	Signing an SLA with clients
4	High	Periodic service reviews for client purposes
4	High	Establishing the service owner role
4	High	Formal announcement of the process for service manufacturing
4	High	Payment of amounts due under agreements with clients
5	Very high	Estimation of recurring service costs (TCO)

High Management Level. A high management level is characterised by the use of tools supporting the settlement of services. A service monitoring system available in the organisation was created in 2008 as part of an IT project. Based on the data collected in the system, services were decomposed, which allowed for the ongoing evaluation of the management level.

Within the organisation the following tools supporting service management were created:

- virtual Service Desk
- tools provided by HP
- model of service decomposition.

Very High Management Level. It is characterised by the implementation of service management processes and the capital asset pricing model (CAPM) for an IT organisation. By implementing the model of service decomposition for each process, detailed process maps were implemented, approved and formally defined. Operational instructions were used for service manufacturing and *dashboard*-type software for data and service level control. Periodic payment of amounts due under SLAs and OLAs was used. A pilot version of the system/SLA was introduced. This enabled the evaluation of service costs at any level.

4.3.3 Forecasting the Development of a Support Organisation on the Basis of the IPP

Based on the assumptions of Chapter 3 (the hierarchical structure of processes in the support organisation, their linguistic description and rule-based processing), below is an example of the application of the model developed in the work. It served the purposes of building development scenarios for a support organisation. Six experiments were run which show its applicability for the improvement and development of maturity evaluation processes in an organisation.

The first experiment was intended for an organisation of very low maturity which wants to develop, but the direction of the development has not been defined. As a result of using a linguistic description, we can analyse many rules, and linguistic measures can be assigned specific values [1]. This enables forecasting and the suggestion of directions for development.

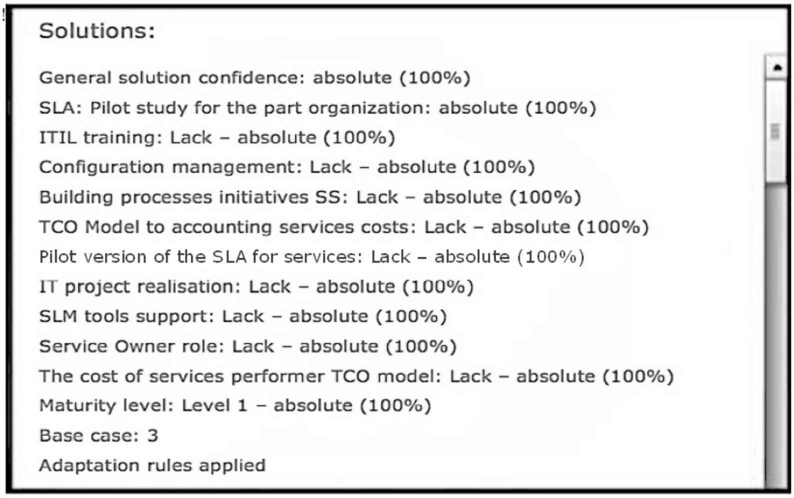
Experiment 1: Forecast for an organisation of a very low maturity level (in chaos)

Inquiry for the System. An organisation in chaos, where there are no structures or obligations. The following variable values were assumed:

- IT-Client obligations = None
- User support = None
- Client needs = Trigger
- Backing by CIO/management = Trigger
- Remaining variables in a non-conclusive area = *.

The term: *A trigger is a fact or artifact of a one-time or permanent nature, which is required for the start and/or duration of a transition process.*

System Response. The system suggests that pilot SLAs should be prepared for the organisation and used for a part of the organisation (Fig. 4.18). In this way, conditions are provided for the verification of the agreements. Their pilot application only for a part of the organisation is an indicator of the correctness of this approach. The system also suggests that the examined organisation is at maturity level one. The description of the remaining variables, described as “None” should be interpreted as of no use in terms of advisory support.


Solutions:

General solution confidence: absolute (100%)
SLA: Pilot study for the part organization: absolute (100%)
ITIL training: Lack – absolute (100%)
Configuration management: Lack – absolute (100%)
Building processes initiatives SS: Lack – absolute (100%)
TCO Model to accounting services costs: Lack – absolute (100%)
Pilot version of the SLA for services: Lack – absolute (100%)
IT project realisation: Lack – absolute (100%)
SLM tools support: Lack – absolute (100%)
Service Owner role: Lack – absolute (100%)
The cost of services performer TCO model: Lack – absolute (100%)
Maturity level: Level 1 – absolute (100%)
Base case: 3
Adaptation rules applied

Fig. 4.18 Development forecast for an organisation prepared on the basis of the MITM model

Experiment 2: Forecast for an organisation building its structures and obligations

Inquiry for the System. An organisation which builds its internal structures and obligations. The following variable values were assumed:

- IT-Client obligations = Partially
- User support = Partially
- SLAs for systems = Prepared for pilot use
- Client needs management = Based on systems
- Client needs = Trigger
- Backing by CIO/management = Trigger
- Decision on the implementation of SS (Service Support) processes = Trigger
- Remaining variables described as None.

System Response. Based on the system suggestions, it can be concluded that the prepared pilot SLAs should be signed (Fig. 4.19). Service Support processes should also be supported, which should result in the creation of relevant units in the Service Support organisation (Service Provision, Service Delivery and ICT). In addition, an organisation responsible for configuration management should be established. It is also suggested that pilot ITIL training be conducted. The effect of the suggested actions is to attain a higher maturity level (level 2).

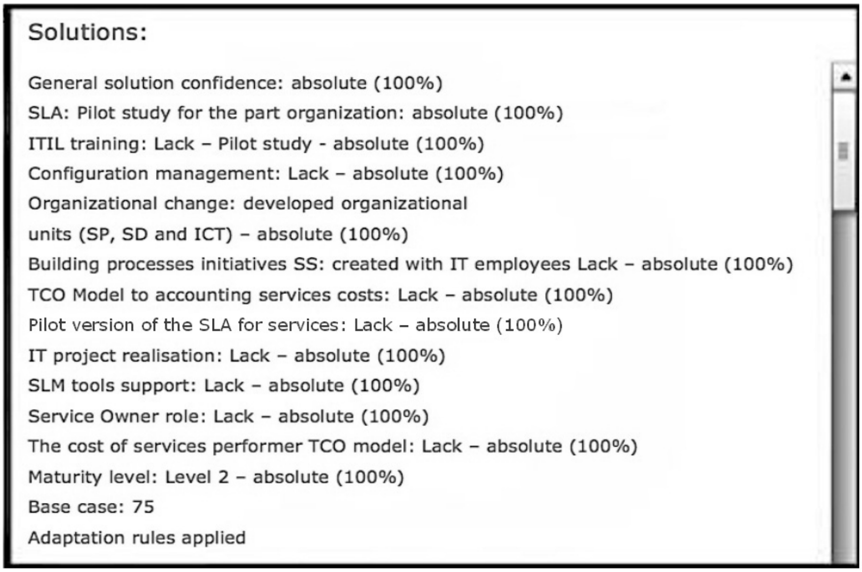


Fig. 4.19 Development forecast for an organisation prepared in line with the MITM model

Experiment 3: Forecast for an organisation of a medium maturity level

Inquiry for the System. An organisation of medium maturity, currently at the next stage in its development. The following variable values were assumed:

- IT-Client obligations = Partially
- User support = Partially
- SLAs for systems = Prepared for pilot use
- Client needs management = Based on systems
- SLM processes = Established
- CAPM processes = Established
- AvM processes = Established
- IT section organisation = Organised
- Service Desk = Established
- Service Support processes = Partly established
- Personnel of suitable education = Partially
- Client needs = Trigger
- Backing by CIO/management = Trigger
- Decision on the implementation of SS processes = Trigger
- Remaining variables described as None.

System Response. The system suggests creating a service decomposition model, which may be the first stage in an IT project (Fig. 4.20), and implementing the model, which will provide terms for raising the organisation to maturity level 3.5 (forecast with a certainty level of 0.95).

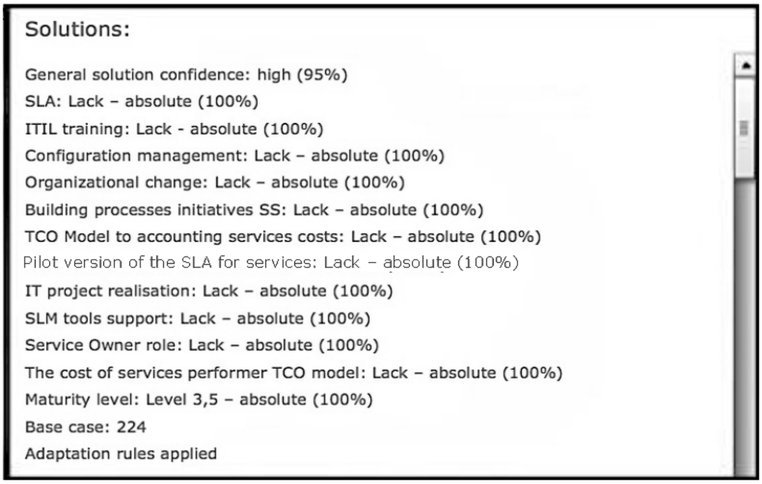


Fig. 4.20 Development forecast for an organisation prepared on the basis of the MITM model

Experiment 4: Forecast for an effective but imperfect organisation

Inquiry for the System. The organisation effectively functions. Although its structures and obligations are operating, the organisation structure is not mature. The parameters assumed:

- IT-Client obligations = Partially
- User support = Partially
- SLAs for systems = In the entire organisation
- Client needs management = Based on systems
- SLM processes = Established
- AvM processes = Established
- CAPM processes = Established
- IT section organisation = Organised
- Service Desk = Established
- Service Support processes = Established in the entire organisation
- Personnel of suitable education = In the entire organisation
- SLAs settlement = Chaotic
- OLAs settlement = Chaotic
- Service costs = Calculated at any level
- Services modified and implemented by projects = Partially
- Client needs = Trigger
- Backing by CIO/management = Trigger
- Decision on the implementation of SS processes = Trigger
- Approval of the service model by the business = Trigger
- Remaining variables described as None.

System Response. The system suggests the implementation of the TCO model for the settlement of service costs, which will improve the service management process (Fig. 4.21). It is suggested that the final project stage is conducted, which consists in taking stock of and decomposing the services. Assuming that SLM support tools have already been developed, the system suggests their implementation in the entire organisation. At the same time, the Service Owner role should be established for the entire organisation; for a part of the organisation the service cost settlement, based on the TCO model, should be activated. After these modifications the organisation should reach a four maturity level (clues were evaluated at a certainty of 85%).

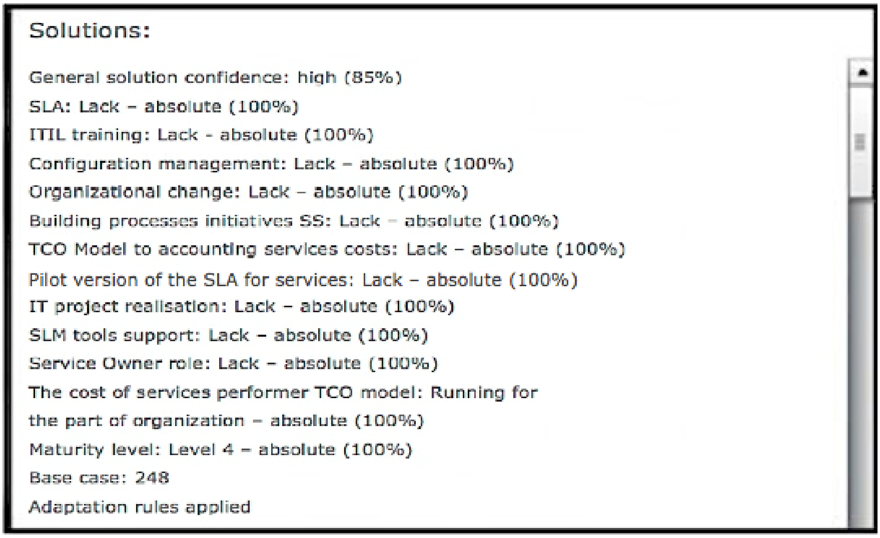


Fig. 4.21 Development forecast for an organisation prepared on the basis of the MITM model

Experiment 5: Forecast for an organisation which only specifies some variables

Inquiry for the System. A case in which the system operator incorrectly enters data. His/her knowledge of the activity of the examined organisation may be very limited, and the data entered uncertain. The following variable values were assumed:

- IT-Client obligations = Partially
- User support = None
- Personnel of suitable education = In the entire organisation
- SLAs settlement = Periodic
- Client needs = None

- Backing by CIO/management = None
- Remaining variables were defined as insignificant for the case and marked with *.

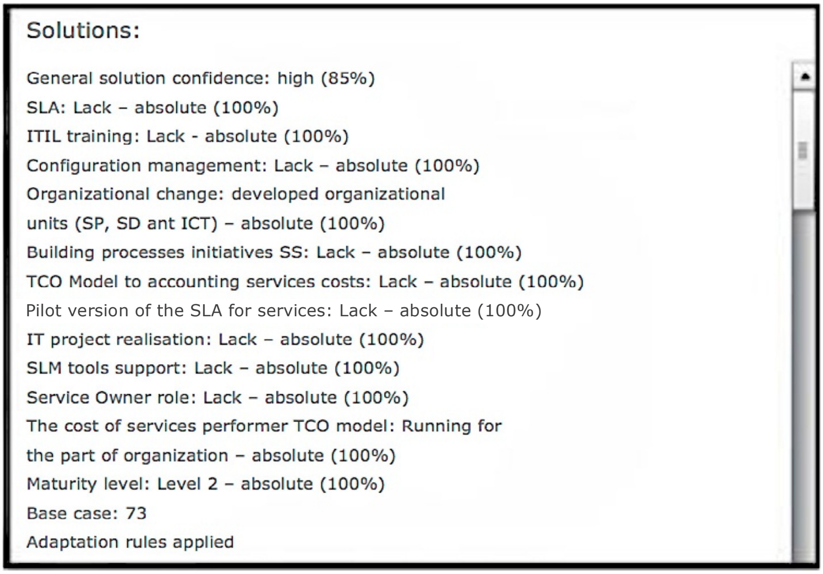


Fig. 4.22 Development forecast for an organisation prepared on the basis of the MITM model

System response. The system suggests that the user establishes relevant units in the organisation as part of the Service Support processes. Despite the modifications, the organisation will still be at a low maturity level. As the solution of Fig. 4.22 is one of many possible solutions. A high certainty level means that the suggestions are worth considering in the decision-making process.

Experiment 6: Organisation of a high maturity level and with poor information flow

Inquiry for the System. An organisation at a high maturity level. The system operator, having limited knowledge of the processes and organisational structure, assumes a low certainty level of the variables entered. For the purpose of the system, the following variable values were assumed:

- IT-Client obligations = Partially
- User support = Partially
- SLAs for the systems = In the entire organisation
- Client needs management = Based on systems
- SLM processes = Defined role
- AvM processes = Defined role

- CAPM processes = Defined role
- IT section organisation = Organised
- Service Desk = Established
- Client needs = Trigger
- Backing by CIO/management = Trigger
- Decision on the implementation of SS processes = Trigger
- Remaining variables described as None.

System Response. The system suggests a low certainty level for the solution (Fig. 4.23). Nevertheless, it is logical. For maturity level three the system suggests implementing periodic ITIL training sessions in the entire organisation. The decision is justified with reference to the organisation development. The system may also suggest other solutions for detailed input data.

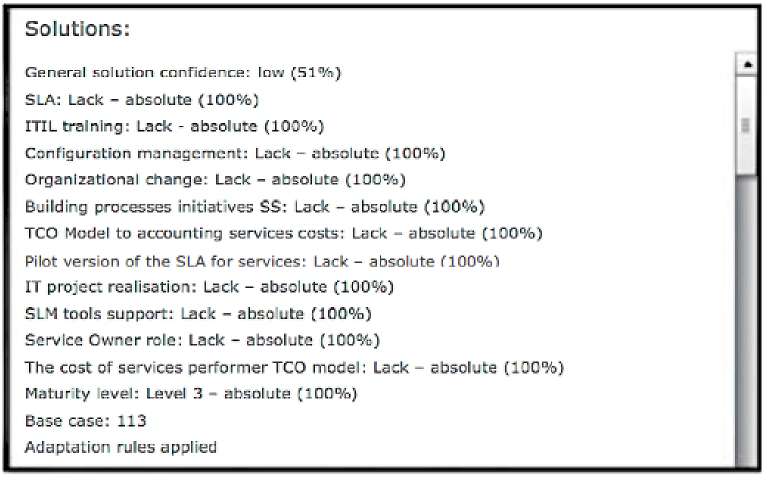


Fig. 4.23 Development forecast for an organisation prepared on the basis of the MITM model

4.4 MITM Verification – Controlling the Evolution of the Client Organisation

The verification of the MITM model with respect to the control of the linguistic degree of maturity of the client organisation is considerd here for an exemplary organisation.

4.4.1 Use of Investigation Questionnaires

For the evaluation of the organisation, a diagnostic sheet was created on the basis of the COBIT 4.1 standard. It includes a checklist of questions related to the evaluation of the organisation level in accordance with the domains indicated in the COBIT description and the corresponding model variables. By means of this, it is possible to identify the weak and strong points of the analysed organisation and indicate possibilities for the implementation of IT systems. In the sheet, the linguistic evaluations as suggested for the discussed models were used.

While evaluating the organisation, priorities were used (of the validity of processes affecting its maturity level) with reference to individual questions. In accordance with the model of the evaluation of client maturity analysed in the work (Section 3.4), each process defined in the sheet (as per COBIT) has a general control objective measured with specific control objectives. The degree of the fulfilment of the general objective is the basis for defining the organisation's maturity level divided into four basic domains. In Figs. 4.24 – 4.28 (showing an example of using COBIT for the evaluation of an organisation in the PO domain 'planning and organisation', in parts I-V) questions are specified which relate to the general objective and specific objectives (the remaining results of our investigation are presented in Appendix 2).

No.	PLANNING AND ORGANIZATION					Grade
	DEFINING THE STRATEGY					
1	Are Internal Clients aware of the values created by IT?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
2	Are IT actions efficient and fully supportive of business actions?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
3	Do decision makers have access to actual, coherent and adequate information that let them make sounder, faster and important decision?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	2
4	Does the IT strategy stem from the company's main business strategy?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	3
5	Is there a plan of IT investment based on defined objectives?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
6	Is there a defined template of IT upholding?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
2	DEFINING INFORMATION ARCHITECTURE					
PO 2.1	Can applications be customized to support decision making?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
PO 2.2	Is there a coherent company-wide data dictionary allowing identical interpretation by all users?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
PO 2.3	Is company data protected?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
PO 2.4	Are procedures assuring coherence and correctitude of electronically stored data defined?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
3	DETERMINING TECHNOLOGICAL DIRECTIONS					
PO 3.1	Are potential phone information needs defined?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	2
PO 3.2	Are there plans for increasing investments into company telecommunication?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	2
PO 3.3	Does the company, in an eventual crisis, have a trusted supplier who could save the business?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	3
PO 3.4	Does the company have standards it bases its actions upon?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	2

Fig. 4.24 COBIT application: evaluation in the PO domain 'planning and organisation' (I)

PO 3.5	Are there formal mechanisms of architecture implementation control?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
4	DEFINING IT PROCESSES - ORGANIZATION AND RELATIONS					
PO 4.1	Is there a Process Framework including specific data regarding process structure.	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 4.2	Is there a commitee to support the IT strategy?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 4.3	Is there a commitee of IT management to define priorities in investment programs?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 4.4	Is there an internal and external organization structure?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	1
PO 4.5	Are there quality assurance groups? Is there a quality management system?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 4.6	Are critical roles for IT risk management defined?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 4.7	Are decisions regarding information systems made by the owners of data and information systems?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	1
PO 4.8	Is there a role division? Are workers only fulfilling only the responsibilities for which they are cleared?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
5	MANAGING IT INVESTMENTS					
PO 5.1	Is there a financial schedule for IT investments and costs?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	1
PO 5.2	Is there an operational decision process for IT resource allocation?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 5.3	Is budget preparation practice set?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 5.4	Are costs monitored and reported? Is there an implemented process of cost management comparing them to the budget?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	1
6	MANAGEMENT THROUGH COMMUNICATION					
PO 6.1	Are IT control environment elements defined?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0

Fig. 4.25 COBIT application: evaluation in the PO domain ‘planning and organisation’ (II)

PO 6.2	What does the IT strategy support policy?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 6.3	Are workers treated as an integral part of the whole company due to implemented IT technology?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	1
PO 6.4	Are all IT users aware of the direction of IT growth?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	2
7	IT HUMAN RESOURCES MANAGEMENT					
PO 7.1	Are worker positions assigned with regard to the workers' abilities?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	3
	Are there positions staffed by people with inadequate qualifications?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
	Does the IT system support correct work assignment?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 7.2	Are worker competences regularly verified?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
	Is there a defined template for IT worker competence?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 7.3	Is there a system for responsibility and role control working with regard to the management policy?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 7.4	Does the system provide IT workers with self-growth opportunities?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
	Do IT workers have the opportunity of expanding their ability?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
	Is IT worker growth coherent with the organization's goals?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 7.5	Are the procedures clear and understandable to users?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 7.6	Are workers regularly rated?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
	Are worker grades compliant with procedures and standards?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
8	QUALITY MANAGEMENT					

Fig. 4.26 COBIT application: evaluation in the PO domain ‘planning and organisation’ (III)

PO 8.1	Are QMS standards followed?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	1
PO 8.2	Does QMS identify quality criteria; key IT processes and their interactions?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 8.3	Does QMS define organization structure for quality management?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 8.4	Are quality standards and practices kept for key IT processes?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 8.5	Is there a general quality plan for corrections?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	1
9	IT RISK ASSESSMENT AND MANAGEMENT					
PO 9.1	Are there IT risk management borders?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	1
PO 9.2	Are these borders compliant with the company risk management borders?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	2
PO 9.3	Are sources of project risk being diagnosed?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 9.4	Are events that negatively affect operative goals being identified and analyzed?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0
PO 9.5	Are risk types defined?	□ Doesn't comply	□ Insufficiently	□ Sufficiently	□ Completed	0

Fig. 4. 27 Example of using COBIT for the evaluation of an organisation in the PO domain ‘planning and organisation’ (part IV)

PO 9.6	Has risk probability been approximated?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
10	PROJECT MANAGEMENT					
PO 10.1	Is there a rating system for project changes?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
	Are all changes analyzed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	2
PO 10.2	Is there a structure for project management?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
PO 10.3	Are particular roles and responsibilities of workers assigned to projects?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	1
PO 10.4	Is there a defined and documented nature in order for stakeholders to be confirmed and expanded?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
	Is the definition of project range formal and confirmed by project sponsors before its initiation?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
PO 10.5	Is announcing to each worker every next step of the project encouraged?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
	Is the confirmation of each next phase of the project based on acceptance of the last phase?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
	Are approval points assumed by the program and sponsors to authorized progress?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
PO 10.6	Are workers rated for work completed?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
PO 10.7	Is there a system for HR assurance?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
PO 10.8	Is there a system for providing the project with the necessary services and resources?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0
PO 10.9	Is analysis and risk assessment conducted?	☐ Doesn't comply	☐ Insufficiently	☐ Sufficiently	☐ Completed	0

Fig. 4.28 Example of using COBIT for the evaluation of an organisation in the PO domain 'planning and organisation' (part V)

4.4.2 Linguistic Evaluation of Control Objectives for Planning and Organisation

Below we present the result of the COBIT analysis (obtained from the questionnaires in Figs. 4.24–4.28) for the 'planning and organisation' variable. The description for the remaining variables is included in Appendix 2, where each control process was assigned a general objective and the degree of its fulfilment evaluated. In addition, the maturity of COBIT processes was also diagnosed (in accordance with the questionnaire). The diagrams presented in Figs. 4.29–4.38 show the evaluation of specific objectives of each basic process. The analysis is complemented by a tabular description of each control process (with areas which require improvement or function to their optimum with respect to maturity).

Based on the results obtained from the diagnostic investigation sheet, a specification of enterprise processes was prepared which classified them as: good, requiring partial improvement, and not functional. The classification is derived from the evaluation of organisation processes in accordance with control objectives, and is linguistic (with four degrees):

- 1 – control objective is not fulfilled (processes do not function properly)
- 2 – control objective is not sufficiently fulfilled (processes require some improvement)
- 3 – control objective is sufficiently fulfilled (processes require large improvement)
- 4 – control objective is completely fulfilled (processes do not require improvement).

Figs. 4.29–4.38 present (in accordance with COBIT) the method for reaching the maturity of processes by fulfilling individual control objectives. The maturity of a process (reaching its high level) may be accomplished in one step (a high level of control objectives immediately translates into a high level of maturity of a process) or in stages (the maturity of a process is reached by raising the level of successive control objectives).

In the second case, illustrated in Figs. 4.35, 4.36, 4.37 and 4.38, the brackets include columns which represent stages in fulfilling the control objectives of individual processes (P07÷P010). The height of the columns indicates the fulfilment level of a given stage (or control objective), which is obtained through control tests. The degrees comprise the maturity level of control processes and the maturity of the examined organisation.

Descriptions of control objectives are included in Tables 4.13-4.22. In the tables, control objectives (entitled PROCESS EVALUATION) have linguistic descriptions assigned to them (*e.g.* "Processes which do not require improvement"). In many organisations, the number of stages of fulfilling the control objectives is limited to one or two (such incompleteness has been demonstrated in the tables and diagrams). The stages of fulfilling the control objectives are identified as PROCESS_ID (second column in the tables), whereas the column CONCLUSIONS RELATED TO PROCESS EVALUATION includes comments which are common for a given control objective.

In addition, each control objective has been assigned weights which represent the significance of a given objective (group of processes) in an organisation:

- 1: control objective is of little significance for the purposes of the organisation, failure to fulfil the objective will not cause major problems in the operation of the organisation in the long term
- 2: control objective is significant, failure to fulfil it in the organisation will cause serious problems in the long term
- 3: control objective is indispensable, failure to implement the processes related to the objective will cause serious problems in the current operation of the organisation.

Taking into consideration the evaluation of control objectives with assigned weights, the degree of fulfilment of control objectives for the organisation processes is estimated (Table 4.11). The final evaluation of the organisation is based on the weighted sum of the fulfilment of control objectives with reference to the maximum result that can be obtained by an "ideal organisation" (an example evaluation is presented in Table 4.12).

Based on the evaluation of control objectives, a complex evaluation of the organisation level was obtained. Table 4.12 presents the results of the evaluation with reference to an organisation at the highest maturity level (maximum result).

Table 4.11 Examination and evaluation of organisation maturity for the ‘planning and organisation’ domain

No.	PO1 – defining strategy	Evaluation	Weight	Result
1	Do internal clients know what values are created by IT?	2	2	4
2	Are IT activities effective and fully supportive for the business activity?	4	3	12
3	Is it possible for decision makers (at all levels) to access updated, coherent and adequate information which helps to make better, quicker and important decisions?	4	3	12
4	Is IT strategy derived from the principal business strategy of the company?	3	2	6
5	Is an IT investment plan established on the basis of defined specific objectives?	3	2	6
6	Is an IT maintenance scheme defined?	3	3	9
	POINTS TOTAL			49

Table 4.12 Results of the evaluation of an organisation’s maturity

Process	**Result obtained**	**Maximum result**	**% of fulfilled control objectives**
PO1 – defining the strategy	49 pts	60 pts	81.67%
PO2 – defining the information architecture	34 pts	36 pts	94.44%
PO3 – determining technological directions	33 pts	40 pts	82.50%
PO4 – defining IT processes	47 pts	64 pts	73.44%
PO5 – management of IT investment	19 pts	24 pts	79.17%
PO6 – management by communicating objectives and directions	22 pts	40 pts	55.00%
PO8 – quality management	66 pts	76 pts	86.84%
PO9 – evaluation and management of IT risk	51 pts	64 pts	79.69%
TOTAL	**321 pts**	**404 pts**	**79.46%**

As a result of the conducted examination, a partial image of organisation maturity in the ‘planning and organisation’ domain was obtained, which should be compared to an organisation of the highest maturity level that, with assigned weights, is awarded maximum grades for each control objective. Against this background, the examined organisation reached the level of 80% fulfilment of control objectives.

Below we present a specification which enables the classification of planning and organisation processes evaluated with the use of the above-mentioned procedures on the basis of the degree of the fulfilment of control objectives. The description is based on a division into domains in accordance with COBIT procedures.

Domain – Planning and organisation

P01 . Defining the strategy

Principal Objective. Control of an IT process – defining the IT strategy compliant with the business objective which is to reach a balance between IT potential and business requirements with respect to IT (also in the future).

Level: Defined

Evaluation of the Possibility of Achieving the Assumed Objective. The principal objective in the organisation has been partially fulfilled. In accordance with the analysis conducted, it can be stated that the organisation classifies the process in this regard at level 3. This means that the process of strategic planning has been defined and communicated to all employees; it also ensures that the relevant planning is practicable. However, there are no procedures for the regular control of planning processes. Details of the evaluation are presented in Fig. 4.29 and in Table 4.13.

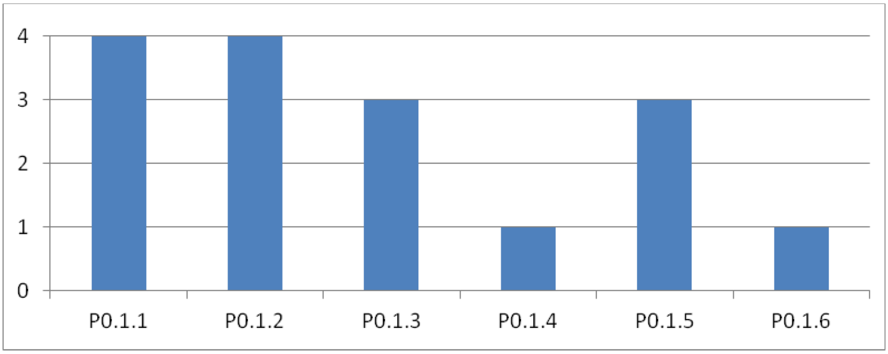


Fig. 4.29 Fulfilment level of control objectives for process P01: defining an organisation’s strategy

Table 4.13 Analysis of process P01: defining an organisation's strategy

EVALUATION	PROCESS_ID	CONCLUSIONS RELATED TO PROCESS EVALUATION
Processes which do not require improvement	P01.1	The process of defining the strategy is implemented in the organisation. The information about the process has been passed on to employees. They also received information related to the general values created through the use of IT in the organisation. It is assumed that all activities concerned with the functioning of information technology fully support the organisation's business activity.
	P01.2	
Processes which require slight improvement	P01.3	The president of the organisation has full access to information which supports making the right decisions. Senior management has access to selected areas of the information.
	P01.5	There is an IT investment plan defined on the basis of the organisation's objectives. The objectives are identified *ad hoc* – when the need arises to expand the system or to improve it. They are not planned in advance, and the investments are adjusted to the current needs of the organisation and are not of strategic importance.
Processes do not function	P01.4	There is no formalised IT strategy, no information safety principles are observed. The principles are informally agreed.
	P01.6	

P02. Defining information architecture

Principal Objective. Control of IT processes in the organisation. Defining the conformance of the architecture with the organisation's business objectives.

Level: Defined

Evaluation of the Fulfilment of the Objective. The importance of enterprise and IT systems architecture is understood by everybody. The process of building the architecture is most often a result of changes in the functioning of an organisation and its business needs. The level of IT safety has been informally agreed by the employees responsible for the IT and is enforced in all areas of IT activity. The organisation's IT systems are supplied with available data, and the need for the supply of the IT systems arises in an informal and intuitive way. The process shows a repeatable quality. An example of such a process is system administration; automatic tools for the general organisation of data administration also appear. Details of the evaluation are presented in Table 4.14 and in Fig. 4.30.

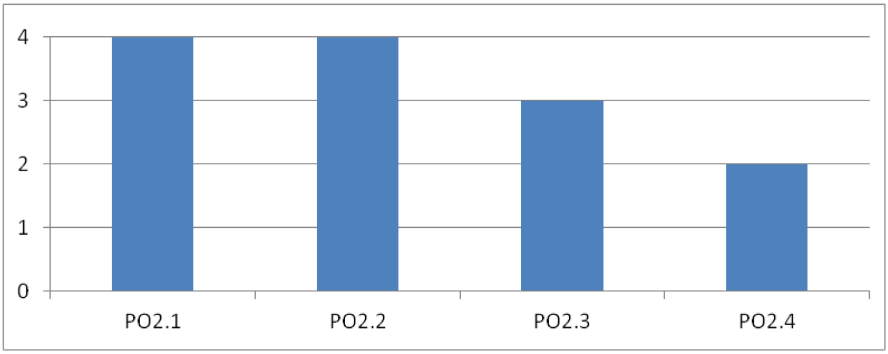


Fig. 4.30 Fulfilment of control objectives for process P02: defining information architecture

Table 4.14 Analysis of process P02: defining information architecture [9]

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS RELATED TO PROCESS EVALUATION
Processes which do not require improvement	P02.1	Information architecture supports the decision making process in the organisation. Technical documentation has been prepared where specialised terminology has been implemented for use in the organisation. The documentation serves as a data dictionary for the organisation, which ensures the use of uniform vocabulary (to identify phenomena/actions) in the organisation.
	P02.2	
Processes which require slight improvement	P02.3	The data processed in the organisation is protected despite the lack of a security policy. The CRM system used by the organisation does not allow the leaking of data outside the organisation's activity area. The principles and procedures of work with confidential documents are clearly defined.
Processes which require considerable improvement	P02.4	The organisation does not have defined procedures to ensure the coherence and correctness of data. Its coherence is determined by the procedures provided by the used IT systems (which do not ensure data coherence).

P03. Determining the directions for information technology development

Principal Objective. Control of the process of information technology application and demonstrating the need for integrating information technologies with business objectives for the implementation of business strategy.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. The organisation should put the integration process to analysis, since it is not fully

Table 4.15 Analysis of process P03: determining the direction for IT development

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS RELATED TO PROCESS EVALUATION
Processes which do not require improvement	P03.2	In the organisation there are plans for the implementation of new telecommunications investments. Their purpose is to improve the functioning of the organisation, *i.e.* expand the CRM system by compatibility with MS Outlook.
Processes which require considerable improvement	P03.3	The organisation has decided to choose OpenSource software, which results in a lack of direct access to the assistance of the software manufacturer. The decision was made in relation to the fact that the SugarCRM system is used by many companies. In critical situations it is thus possible to seek help from other users. An example of such behaviour might be using online platforms. This extends the time of obtaining the information about the system and does not guarantee that the problem will be solved.
Processes do not function in the organisation	P03.1	The organisation has not defined the telecommunications needs related to its strategy. Neither does it make plans for development – in the aspect of strategy preparation.
	P03.4	In the organisation no standards are observed with regard to examination and evaluation, such as ISO or ITIL. As a result of not using such standards, the processes lack maturity and there is no policy for improving their quality.
	P03.5	Formal means for control of the efficiency of the implementation processes of enterprise architecture and IT are not applied. The company is not ready to assess whether its processes are optimal, which of them require changing, in which areas and in what way.

implemented. The planning of IT investments is conducted solely on the level of tactics and is the effect of technical problems which arise and the response to those problems. The evaluation of changes in the organisation is not carried out formally (the process has not been defined). The evaluation is performed by various people (no independent position) and is intuitive. No formal training is conducted and information of the roles and responsibilities of employees is not communicated. Details of the evaluation are presented in Table 4.15 and in Fig. 4.31.

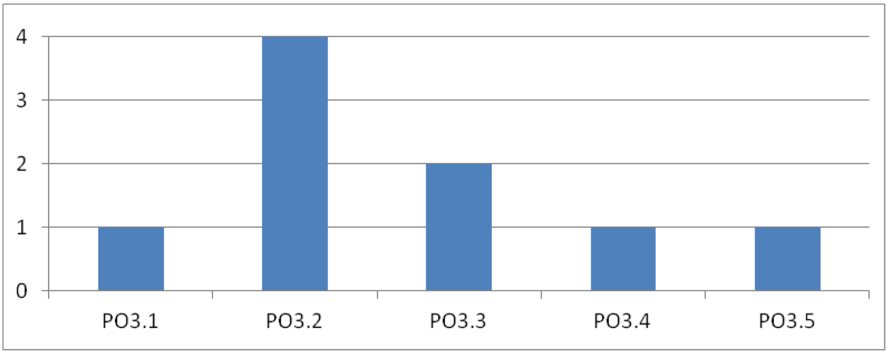


Fig. 4.31 Fulfilment of control objectives for process P03: determining the direction for information technology development

P04. Defining the organisation processes and their relationships with reference to IT

Principal Objective. Control of the IT process – defining the IT organisation processes and their relationships with reference to the information technologies used.

Level: Defined

Evaluation of the Possibility of Achieving the Assumed Objective. In the organisation, the roles and responsibilities of employees and business partners have been defined. For the organisation a certain policy for development and communication between partners has been adopted. The processes are not documented, so their conformance with the IT development strategy cannot be confirmed. The IT organisation (complete in functional terms) is focused on the development of technology rather than business issues (no analysis). IT employees and collaborating business partners have clearly defined tasks. The evaluation data are given in Table 4.16 and in Fig. 4.32.

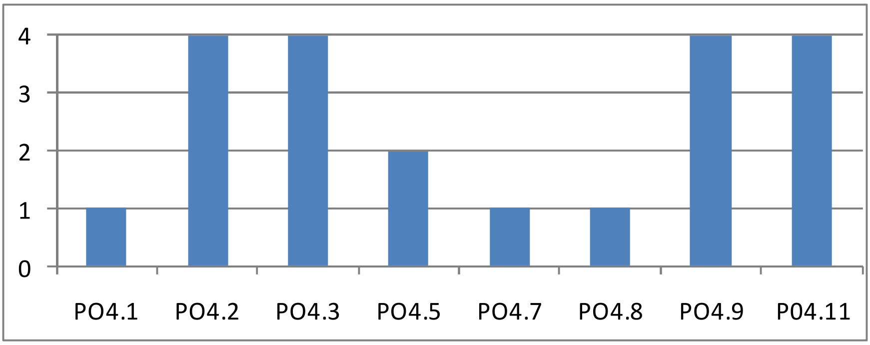


Fig. 4.32 Fulfilment of P04 objectives: defining IT processes: organisation and relationships

Table 4.16 Analysis of P04: defining organisation processes and their relationships with IT

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS RELATED TO PROCESS EVALUATION
Processes which do not require improvement	P04.2	The president is responsible for support of the complex IT strategy, although the strategy itself has not been formally defined in the organisation. A one-person IT control committee has been established to set priorities for the implementation of investment programs. The task of the committee is also to control the project level, monitor the level of service and improve it. A clear division of roles and responsibilities for work performed has been adopted in the organisation. There are no cases of employees neglecting their duties.
	P04.3	
	P04.9	
	P04.11	
Processes which require considerable improvement	P04.5	In the organisation an internal and external organisation scheme has been adopted. It is impossible to establish whether and how business objectives are fulfilled. They have not been defined.
Processes which do not function	P04.1	In the organisation there are no process frameworks which include details of their structure and relationships between them. There is no model of business processes. In the organisation there are no formalised units responsible for quality management. There is no quality assurance unit or a quality management system.
	P04.7	
	P04.8	No IT risk management. Critical roles for IT risk management are not defined.

P05. IT investment management

Principal Objective. Control of the IT process – IT investment management whose business objective is to ensure control of the disbursement of financial resources.

Level: Managed

Evaluation of the Possibility of Achieving the Assumed Objective. IT investment management processes are not very advanced. However, good practices are used with respect to cost and process identification to boost the effectiveness of the investment. The processes are constantly improved. The need for the support of investment initiatives of an extended time schedule has been recognised and new business processes are created, which are enabled by the use of suitable technologies. Details of the evaluation are presented in Table 4.17 and in Fig. 4.33.

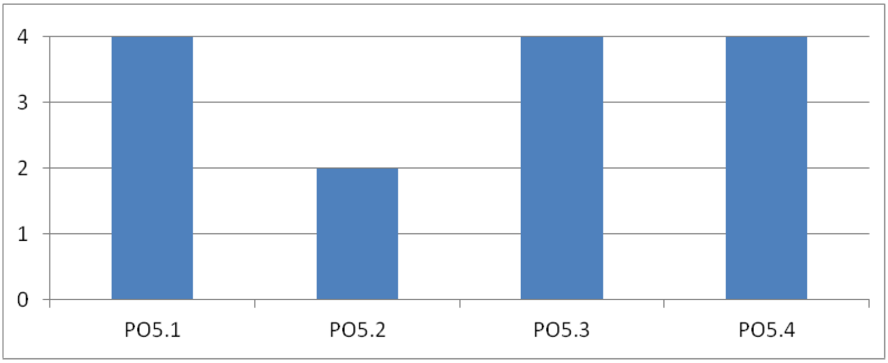


Fig. 4.33 Fulfilment of objectives for process P05: IT investment management

Table 4.17 Analysis of process P05: IT investment management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS RELATED TO PROCESS EVALUATION
Processes which do not require improvement	P05.1	In the organisation, financial frameworks for IT investment and cost management have been established. They provide for the planned budget for investments in progress. The organisation has established and implemented practices for the preparation of such budgets. In the organisation, the processes of cost monitoring and reporting on costs are operating. A cost management process which includes the budgeting system has been implemented (with costs derived from the prepared budget).
	P05.3	
	P05.4	
Processes which require considerable improvement	P05.2	In the organisation, no decision-making process has been implemented which relates to assigning a priority status to IT resource allocation. This is a consequence of decisions being made single-handedly by the President of the Board.

P06. Management of objectives and directions for organisation development

Principal Aim. Control of the IT process – management of objectives and directions for organisation development and compliance of the process with the business objective. The control is intended to ensure awareness and understanding of the objectives and directions for enterprise development among employees.

Level: Initial

Evaluation of the Possibility of Achieving the Assumed Objective. The process should be considerably improved. In the organisation, no relevant control policies, procedures or standards have been implemented. They are created *ad hoc*, as required. Documents which include procedures and standards are created according to individual needs. Management can be described as improvised, referring to the requirements of an information control environment. The processes of organisation development and communication for ensuring this development, and the compliance of the enterprise strategy and IT operate informally and inconsistently. Training is provided for interested individuals as needed. Details of the evaluation are presented in Table 4.18 and in Fig. 4.34.

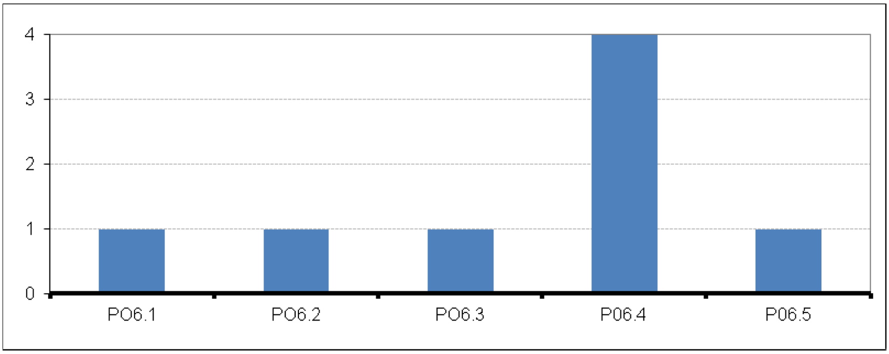


Fig. 4.34 Fulfilment of objectives for process P06: management of objectives and directions for organisation development

Table 4.18 Analysis of P06: management of objectives and directions for development [9]

<table>
<tr><th>PROCESS EVALUATION</th><th>PROCESS_ID</th><th>CONCLUSIONS RELATED TO PROCESS EVALUATION</th></tr>
<tr><td>Processes which do not require improvement</td><td>P06.4</td><td>All members of the organisation are treated as an integral part of the entire organisation. This is enabled by the use of suitable IT technologies. They are users of systems indispensable for the organisation's operation.</td></tr>
<tr><td rowspan="4">Processes which do not function</td><td>P06.1</td><td rowspan="4">In the organisation there are no defined elements of an IT control environment.
In the organisation there are no frameworks which would define the holistic approach to IT risk.
This prevents the development of the organisation.
There is no policy of supporting the IT development strategy.
IT system users are not informed of either the objectives nor the directions for IT development.</td></tr>
<tr><td>P06.2</td></tr>
<tr><td>P06.3</td></tr>
<tr><td>P06.5</td></tr>
</table>

P07. Human resource management in IT

Principal Objective. Control of the IT process – human resource (HR) management in IT. The objective for the process is to have and maintain a highly motivated and competent team, and to ensure the strongest possible participation of the team in IT processes.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. The process needs to be improved. In the company there is a need for IT human resource management. The activities are carried out on a tactical rather than strategic level. No regular analysis of the pay scale in the IT sector is conducted to ensure competitive salaries. In the organisation, career paths are not taken into consideration. Informal training is provided for new employees (as required). Details of the evaluation are presented in Table 4.19 and in Fig. 4.35.

Process P07.1 refers to assigning positions and should be automatic (a position is assigned on the basis of skill and suitability). It does not exist in the organisation.

The diversity of control objectives may raise reservations as to the reliability of such an evaluation (which is very difficult anyway). Experience suggests that it is ambiguous and that formal evaluation procedures (COBIT) need to be implemented in order to standardise its results. Due to the lack of such procedures, the evaluation of the organisation's maturity, performed by estimating the fulfilment level of control objectives, may be subject to significant error.

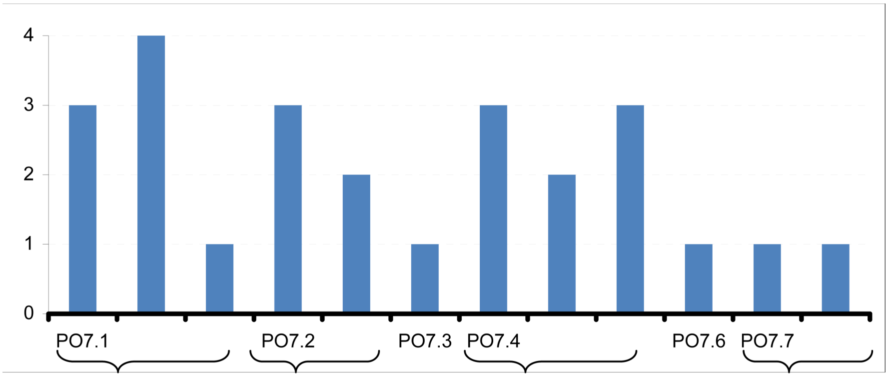


Fig. 4.35 Fulfilment level of control objectives for process P07: human resource management

In Fig. 4.35 the maturity of process P07 is evaluated for seven control objectives P07.1÷P07.7. Within each control objective we can distinguish stages of fulfilment. For example: control objective P07.1 includes three stages (the heights of three columns which represent them indicate the degree of fulfilment of the requirements set for the stage), and objective PO7.7 – three stages (three columns). The levels of stage realisation are defined on the basis of responses to control questions related to the control objectives in the organisation on a scale of 0-4. The descriptions of control objectives are included in Tables 4.13 – 4.22.

Table 4.19 Analysis of process P07: human resource management in IT

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS RELATED TO PROCESS EVALUATION
Processes which do not require improvement	P07.1	Process P07.1 (refers to filling positions) has not demonstrated the unsuitability of staffing in the organisation.
Processes which require slight improvement	P07.2	Process P07.2 referring to competences of employees has demonstrated that they are verified on a regular basis. The person responding to questions could not specify if they are valid.
	P07.4	Process P07.4, referring to skill development in the IT team, has demonstrated that the organisation supports employees in the selection and conducting of training which raises their qualifications. These are not training programmes. There are no defined career paths for employees. The training of IT personnel supported by the organisation is compliant with the set objectives.
	P07.1	Process P07.1, referring to the division of positions, demonstrates that in most cases the positions are assigned in accordance to the skills and qualifications of employees.
Processes which require considerable improvement	P07.2	Process P07.2, referring to the competence model, demonstrates that there is no formalised model for the competences. There is also no possibility for the verification of those competences on the basis of certificates owned by employees (although there are exceptions to this rule in the organisation).
	P07.4	Process P07.4, referring to the self-development orientation of employees, does not demonstrate the direct stimulation of employees with respect to raising qualifications. There is also no system solution to deliver information on the career paths of employees on the basis of certification paths. Such information may be obtained through relevant CRM system reports, which show the employees' development paths and provide information and motivation for raising the effectiveness of own certification.
Processes which do not function in the organisation	P07.1	Process P07.1 refers to assigning positions. It should be automatic (a position assigned on the basis of skill and suitability). In the organisation examined this is not the case.
	P07.3	In the organisation there is no control system or supervision of roles and responsibilities. There is also no remuneration system compliant with the adopted management policy, procedures, code of conduct, *etc*.
	P07.6	In the organisation there are no procedures which determine actions to be taken in such cases.
	P07.7	The area does not apply to the examined organisation. The evaluation of employees is not conducted on a regular basis. No standards or specification of the number of employees have been adopted. The organisation employs few people and does not see the need to implement the process.

P08. Quality management

Principal Objective. Control of the IT process – quality management enables the evaluation of the compliance of IT processes with business requirements, and ensures the continuity of the evaluation and measurable improvement of IT quality.

Level: Initial

Evaluation of the Possibility of Achieving the Assumed Objective. The process needs to be improved. In the company no actions are taken to improve the quality of IT processes. There is also no defined system of quality management. Details are given in Table 4.20 and in Fig. 4.36.

Table 4.20 Analysis of process P08: quality management

<table>
<tr><th>PROCESS EVALUATION</th><th>PROCESS_ID</th><th>CONCLUSIONS RELATED TO PROCESS EVALUATION</th></tr>
<tr><td rowspan="5">Processes which do not function in the organisation</td><td>P08.1</td><td rowspan="5">In the organisation there is no quality management system. No standards for action have been established; quality requirements, key IT processes, their consequences or interactions cannot be specified. In the organisation structure there is no unit for quality management which, with regard to the quality system, precisely defines the roles and responsibilities.
Lack of a quality system in the organisation prevents the maintenance of standards for key IT processes.
This situation can be reflected in the ineffective functioning of the organisation. Due to the lack of a general quality programme it cannot be ascertain ed that the firm's objectives will be fulfilled. A quality programme should provide support for those objectives.
In the organisation there is no system for the compliance measurement of the implemented processes with the stipulations of the quality system. Processes cannot be corrected or optimised, as there is no basis for their improvement.</td></tr>
<tr><td>P08.2</td></tr>
<tr><td>P08.3</td></tr>
<tr><td>P08.5</td></tr>
<tr><td>P08.6</td></tr>
</table>

P09. Management and evaluation of IT risk

Principal Objective. Control of the IT process – management and evaluation of IT risk. The aim of the control is to verify the compliance of IT processes with business objectives and indicate hazards for their compliance. To this end, relevant actions are taken, such as limiting the complexity of projects and increasing the objectivity of evaluation in important decision areas.

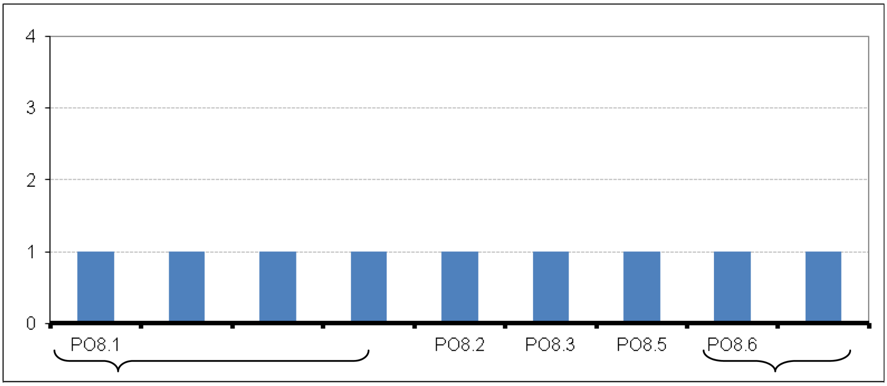


Fig. 4.36 Fulfilment of control objectives for process P08: quality management

Level: Initial

Evaluation of the Possibility of Achieving the Assumed Objective. The process requires considerable improvement. In the organisation, the evaluation and management of IT risk are not conducted. The company is aware of its legal, contractual obligations, but it considers the risks for IT processes at an *ad hoc* basis. No defined procedures/policies are applied. In the informal risk evaluation processes exist for each project. Roles and responsibility for project development risk, however, are assigned. The evaluation of risk related to current IT processes is rarely discussed in management meetings. If a risk appears, no procedures for limiting the risk are consistently applied. Details of the evaluation are presented in Table 4.21 and in Fig. 4.37.

P10. Project management

Principal Objective. Control of the project management process. The aim of this process is to control the priorities adopted for project development. These include: time of development, scope and project budget.

Level: Defined

Evaluation of the Possibility of Achieving the Assumed Objective. In the organisation, the method for project management has been adopted, formally established and presented to the team members. After adopting the method, the relevant business, technical and technological objectives of the IT projects developed are defined. Clients are involved in the implementation of project management processes. For each project, milestones are set and roles and responsibilities are selected. Details of the evaluation are presented in Table 4.22 and in Fig. 4.38.

Table 4.21 Analysis of process P09: management and evaluation of IT risk

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS RELATED TO PROCESS EVALUATION
Processes which do not function in the organisation	P09.1	In the organisation no development risk management frameworks for IT processes have been defined (risk is specified solely at the level of projects developed for external clients). In the organisation the compliance of IT process risk analysis with the frameworks defined for the purposes of the organisation cannot be determined.
	P09.2	
Processes which require considerable improvement	P09.3	In the organisation no diagnosis of risk sources is performed, so it is not possible to improve preventive processes. There is no distinction between the kinds and types of risk. Therefore, the risk cannot be estimated or identified. Neither can its influence on the functioning of the organisation be examined. There are also no procedures for assuming responsibility for IT process risk. Control processes for the analysis and evaluation of IT process risk are not planned.
	P09.4	
	P09.5	
	P09.6	

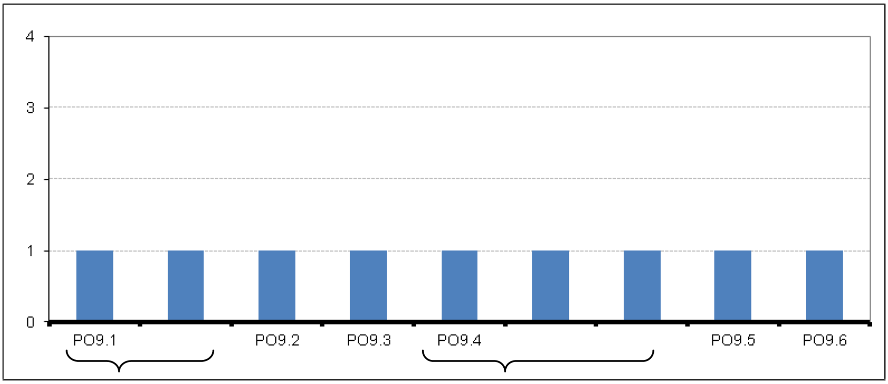


Fig. 4.37 Fulfilment of control objectives for P09: management and evaluation of IT risk

Maturity Model of an Organisation. The maturity model of an organisation, which includes 34 control processes within 4 main domains, enables the evaluation of the organisation's level in accordance with the structure presented above. For the evaluation of the maturity level, borderline values have been adopted as required for reaching a given level in accordance with the breakdown presented in Table 4.23.

Table 4.22 Analysis of process P010: project management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS RELATED TO PROCESS EVALUATION
Processes which do not require improvement	P010.2	In the organisation, the structure for project management has been defined. Each project is run by a Project Manager, who builds a suitable team. Each project team member performs precisely defined tasks and is responsible for them. For each project, a scope is defined which is approved by a group of clients. Each stage is approved by the client. Contractors and client representatives decide on the realisation of ensuing stages. In the case of process P010.6, the initiation of each consecutive stage is clearly presented to all project participants. As part of the process, the analysis of the project development risk is discussed in a meeting of the project team. Each team member is subject to evaluation based on the scope of work performed in the course of the project.
	P010.3	
	P010.5	
	P010.6	
	P010.9	
	P010.7	
	P010.13	In the organisation there is a system for result measurement, which collects reports regarding costs and the degree of fulfilment of the project objective.
Processes which require slight improvement	P010.5	Analysis of process P010.5 (with regard to the need for defining the project scope) demonstrates that defining the project scope in the organisation examined is usually of a formal nature and is approved by project sponsors before the start of a project. Analysis of process P010.6 (with regard to the approval and acceptance of the products of a project stage) demonstrates that the approval of each successive project stage in the organisation examined is based on the realisation and approval of the previous stage. It is also recognised that it is possible to approve of the possibility of the realisation of an updated business case for the following period on condition that a method for project management is used in the organisation. Analysis of the process also demonstrates that in the organisation there is a system for controlling changes in the project, whose operation basically consists in the control of the software version. Analysis of the closing process demonstrates that it is implemented in a semi-formal way. Knowledge of the project's successes and problems is not collected.
	P010.6	
	P010.12	
	P010.14	

Table 4.22 (*continued*)

Processes which require considerable improvement	P010.1	Analysis of process P010.1 (changes in the projects) demonstrates that not all changes are analysed in the organisation.
	P010.8	The analysis of the process of assuring resource and service availability shows that their evaluation is conducted manually. In the organisation there are no tools to support the process. There is no analysis of the workload level for individual team members.
Processes which do not function	P010.1	No evaluation system of the influence of changes on the development of the project.
	P010.10	No quality management plan for projects.
	P010.11	No quality assurance plan.

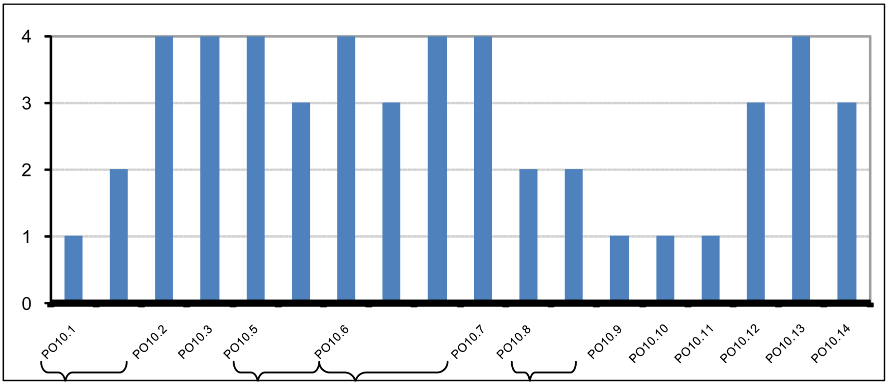


Fig. 4.38 Fulfilment of control objectives for process P010: project management

Table 4.23 Borderline values for the degree of process qualification for the maturity level

No. (level)	Fulfilment degree of specific objectives (borderline values)	Level
0	0,00%	non-existent
1	20,75%	initial
2	41,50%	repeatable
3	62,25%	defined
4	83,00%	managed
5	100,00%	optimised

The evaluations of control processes in the planning and organisation domain as expressed in Fig. 4.39 show an even distribution of the maturity level of each process, *i.e.* there are processes which require considerable improvement and also those that function nearly optimally. The organisation, if it intends to grow and make progress, should improve the processes of the lowest evaluation (quality management and IT risk evaluation and management). In Fig. 4.39 it is demonstrated which processes require considerable improvement (initial or repeatable level) and should be developed (defined, managed or optimised level).

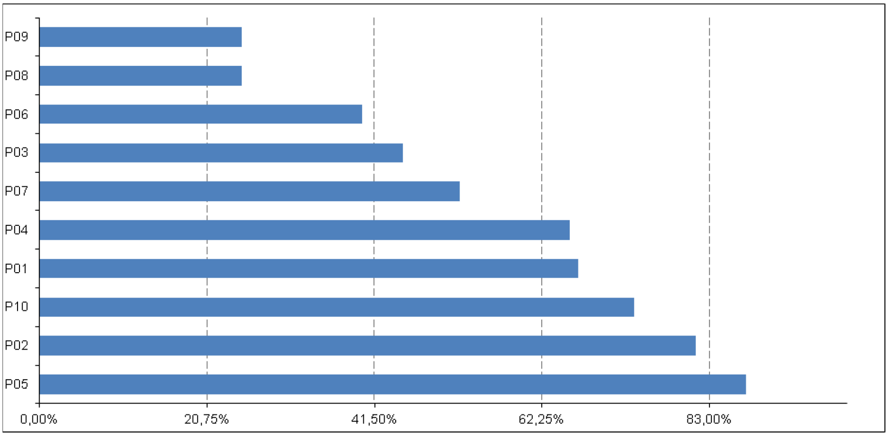


Fig. 4.39 Degrees of maturity of control processes in the PO domain

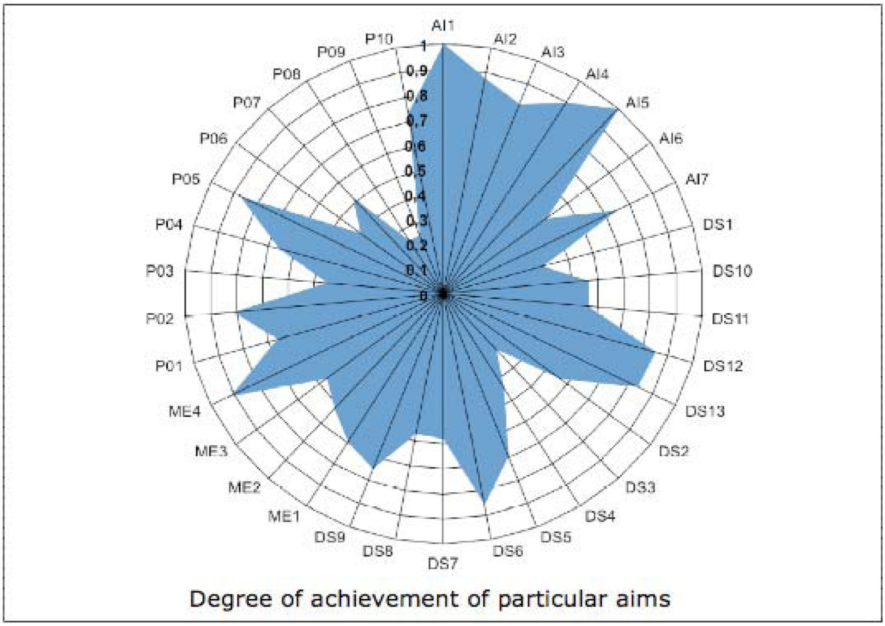


Fig. 4.40 Degree of specific objective fulfilment for 34 processes

In Fig. 4.40 the degree of the fulfilment of specific objectives in the organisation is presented. Based on those results, it can be stated that the organisation, despite numerous faults and shortcomings, has mature processes in areas which need to be a little improved in order to move to a higher level. Lack of procedures in the organisation significantly decreased the evaluation of many processes. Therefore, it would be beneficial to apply good practices resulting from the long-term implementation of repeatable processes in the organisation. That is why the evaluation of the organisation in accordance with COBIT does not fully reflect its actual maturity. Due to its nature and small scale, the organisation examined does not require the use of strictly formalised mechanisms or procedures, without which it functions well in the market and ensures the security of material, immaterial and information resources.

4.5 Summary and Conclusions

In this Chapter, we presented the environment and the results of quality verification and specification which document the usefulness of the prepared model of information technology management.

Since IT management is an aspect of IT project management (except for organisation and process management), we tried to demonstrate that the solution proposed here is useful for project managers. The structure of the verification environment, with consideration to the specific nature of IT projects, has proven the significance of both technical and technological aspects, as well as social ones. For these to be taken into account, hierarchical-staged verification procedures have been proposed.

The applicability of the prepared MITM model was evaluated. The multi-level sequential model MSM for the selection of IT technologies presented in Chapter 2 as well as the model for initial processing procedures (IPP) and the maturity capsule were verified. The applicability of the capsule was examined for Governance processes in an organisation which designs enterprise architecture (with the use of project negentropy). The functionality of the IPP model was verified on two levels of detail. On the first level, the possibility of adaptation to the proposed approach for forecasting the development of a supporting organisation was presented. On the second (lower) level, a classic reference was made, whereby the IPP model was used for the evaluation of the organisation's maturity. The environment of multi-aspect verification clearly indicated the usefulness of the prepared MITM model.

The verification of the MITM model was performed on the IBM RTC platform. The tool selection lies within the competences of the provider organisation's manager. As the manager runs the project, he/she chooses the management method and the platform. The implementation of the model prepared on the platform demonstrated its usefulness in the selection of methods and tools to minimise the risk of project failure. It also enables the manager to monitor the project development status and make adequate decisions related to possible changes to maximise its negentropy.

The study of the prepared solution for the purposes of supporting Governance processes has demonstrated that a sectional consideration of the project in the categories of its negentropy organises the implementation of Governance processes. The proposed monitoring of Governance processes with the use of project negentropy confirms the need for the analysis of conformance processes (for established areas). The two considered cases (Sections 4.3.1 and 4.3.2) of Governance processes for the design of enterprise architecture and its implementation prove that monitoring a project in the categories of its negentropy makes it easier to identify Governance areas, and at the same time reduces the risk of project failure and increases the chances for ensuring corporate governance in a company where an IT system is implemented.

Control of development processes in an organisation is crucial for the development of a supporting organisation. The model developed in the work assists the CIO in making decisions in this respect. The ITIL standard has so far been used only for building the structures of such organisations. The proposed solution, where the evaluation of the provider organisation (its objectives and processes) is performed in linguistic categories, allows for planning a development path and including mechanisms of change. The prepared specification of the evaluation of provider organisation maturity has brought about the classification and identification of potential directions for the development of organisation statuses. In order to authenticate the prepared model, six experiments were used which illustrate scenarios for organisation development. Statuses of IT support organisations were presented as well as the method for creating conditions for presenting directions in the development.

The final verification stage was based on the environment of client organisation maturity evaluation, with the use of investigation questionnaires. This created the need for their modification and consideration in linguistic categories. The typical two-value scales used for the indirect evaluation proved ineffective. Therefore, a four-level scale was used, which better reflects intermediary statuses of an organisation and complies with the philosophy of using information technology standards.

The client organisation analysed was in the process of implementing the CRM system in order to build enterprise architecture. This case qualifies for projects of high negentropy, where the evaluation of the client's level is of considerable importance. The COBIT philosophy was used with modified (for the IPP) control questions. The conducted analysis improved the relationship between the provider and the client and contributed to identifying 'problem' areas.

The verification process showed multiple aspects of the prepared MITM model, both with reference to the organisation and to the projects. The model effectively supports the environment of IT project management, and can form the basis for building systems to support decision-making, *i.e.* it can be used while making decisions with respect to changes in project negentropy as well as changes in client and provider organisations.

5

Summary

The discussion was devoted to the subject of information technology management (ITM) and its model (MITM) built with the use of formal tools, on the basis of the authors' years of experience in information technology project management and based on standards for enterprise architecture development (TOGAF), and organisation evaluation (ITIL and COBIT). This model can be used to support the managers of provider and client organisations in decision-making within project management and in the selection of methods and tools to support management processes. The model helps to evaluate the provider (supplier) and the client organisation and to estimate project negentropy. It is easier to allocate adequate resources and reduce project risk when project negentropy is known. In consequence, the possibility to realise such projects is increased, in relation to projects managed without this model.

A complementary aim, which emerged during the work on this book, was the critical evaluation of the applicability of IT standards, not only in terms of modelling, but also in terms of the evaluation of IT organisations. The conducted analysis has shown how difficult the implementation of information technology standards is in organisations.

The book has been divided into four main parts.

The first part presents the major aims of the book and the methods to achieve them.

The second, main part discussed project manufacturing and management processes taking place in client and provider organisations in order to develop the MITM model. The presentation includes basic statements, terminology useful in the modelling process, ITM management tools, the concept of negentropy and the maturity capsule, as well as IT systems architecture and IT standards useful in evaluating the capsule. Subjects also discussed were the dynamic model of IT project management – MITM, and its components: the static initial processing procedures – IPP, the maturity capsule, and the linguistic model of the fuzzy-rule-based decision-making subsystem which executes the MSM model of information technology selection. The maturity capsule, including project negentropy and the

Z. Kowalczuk and C. Orłowski, *Advanced Modeling of Management Processes in Information Technology*, Studies in Computational Intelligence 518,
DOI: 10.1007/978-3-642-40877-9_5, © Springer-Verlag Berlin Heidelberg 2014

maturity of the provider and the client, was complemented by the methodology of the evaluation of variables.

The developed MITM model is the main model of information technology management – *ITM*, the IPP processes and the maturity capsule are two (internal and contextual) sub-models of project management, while the MSM is a technology selection sub-model realised as a decision-making subsystem (*i.e.* MITM is created by the IPP and MSM subsystems).

The third part was devoted to the specification of variables and a discussion of the IPP processes, as well as the maturity capsule. The applicability of the IPP processes in defining the MITM model was evaluated, especially in the linguistic description of variables. The concept of the maturity capsule was used in determining the maturity of the client and the provider, and the maturity of the project expressed by the concept of negentropy. Extracting the knowledge contained in the TOGAF standard on enterprise architecture development helped to establish the details of project negentropy. For a complete specification of the model, maturity evaluation processes of provider and client organisations were defined, with the use of IT standards.

The fourth part of this work concerns the verification of the developed model. It shows how the solutions proposed here can be used by those who manage projects, manufacturing teams and client teams in order to support the processes within the project in terms of monitoring and predicting its development.

Four levels of verification were suggested. On the first one, the applicability of the MITM model for information technology project managers was evaluated. The second level referred to support for the processes of corporate governance (Governance) and the use of project negentropy in supporting project management processes. The third level was devoted to the linguistic evaluation of provider organisation maturity in predicting its evolution. Whereas on the fourth level, 'of controlling the client organisation and the change processes of this organisation,' a linguistic description was used to support such an evaluation.

The Continuum project repository, in which all the useful experience and artifacts (processes and documents) taken from successful projects are stored, has a special function in the development of architectures. It is an important aid in the implementation of current projects. Thus the Continuum should reflect successful contracts and their preparation processes as well as the governance processes related to them (Figure 1.1 highlights two process areas: the current project and the processes contained in the Continuum). In this way, when describing processes of the current project, the project manager can relate to the processes of governance contained in the Continuum, which relate to the development, monitoring and control of documents (needed for the preparation of the contract) and to ensuring compliance of the implementation of the project with the processes contained in the Continuum.

An important objective of this book was to evaluate the applicability of the TOGAF, COBIT and ITIL standards for those who manage information technology projects. This view is related to the usefulness of the knowledge contained therein in evaluating the values of the model variables. The demand from team managers for a set of documents which would be useful in taking management decisions during a project was also an important factor. Our experience of participation in IT projects confirms this. Therefore, it must be concluded that the carried out evaluation of the potential application of these standards is useful in the implementation of information technology projects.

A deeper analysis of the IT standards shows:

- their significant generality and lack of indication as to their applicability for different projects (no guidelines for project manufacturing and management processes of all sizes)
- content redundancy in many of their parts (*e.g.* in the case of TOGAF, a similar description of ADM processes occurs in many parts of the work)
- conceptual changes, versions 8 and 9 of TOGAF differ in many important areas (see the introduction)
- lack of clarity with respect to several TOGAF standard specifications – the construction of the Continuum and its TRM and SIB models: for example, the standard analysis does not answer the question of whether it is more favourable for the project manager to build a detailed Continuum first (as a supporting process), and then implement the ADM process, or whether the task should be performed in the reverse order.

There are more problems like those mentioned above, and this is not only a feature of the TOGAF standard. A detailed analysis of the COBIT and ITIL standards shows similar defects. Therefore, project managers should be made aware of this situation when they are presented with shortened versions, based on artificial intelligence and models tailored to the wishes of managers. Such descriptions can be used in agent-based systems of information technology evaluation which, being knowledge-based, can be used to formalise the knowledge expressed in the maturity capsule.

The proposed evaluation system also requires that the suggested models of *management and development sequences*, describing management and development processes, are taken into account. This allows for the introduction of strict quantitative measures (Section 2.5.7), and makes those measures more real for the actual development and management processes. It also allows the use of this model to support the architecture development process regardless of the method, *i.e.* the software development cycle (iterative or the waterfall model [43, 53]).

The Appendices attached, divided according to the basic variables of the model, present the collected material needed for the IPP processes and their verification, as well as for constructing models and specifying variables. This can be a complement for the users of IT standards. It can also be used in the further development of the model.

Overall, it appears that the standards presented in the main part of this work and its Appendices may be used satisfyingly for the evaluation of information technologies and the verification of the MITM model.

The modelling of project and information technology management is based on a formal, discrete-time linear dynamic description extended with the essential non-linear mechanisms, realised in the form of the fuzzy-rule-based system (with a linguistic evaluation developed on the basis of experts' responses to sets of questions contained in an interview questionnaire).

The authors' innovative method of modelling the management processes of projects and information technologies has been applied in the work. During the development of the presented MITM model, the authors had a number of dilemmas and doubts. The first refers to the structure and complexity of the model, especially the problem of whether the variables used in the model fully describe the IT project management process. It seems that the answer to this question is positive, as shown by the fact that the initial processing described here proved to be useful for the specification of the MITM model variables.

The analysis of determining the *client maturity* variable highlighted the need for the client to accept the subsequent manufacturing products. This approach (acceptance or rejection) is, however, only possible when we are dealing with a mature client. Any attempts to contact an immature client can directly contribute to the failure of a project. Therefore, taking into account the *client maturity* parameter is very important in the construction of the project management environment.

The *provider organisation maturity* variable performs a similar function. Its application is very important for a project (with any negentropy) when dealing with the client and in the manufacturing processes. It also involves mature managers having a different vision of the project and opportunities resulting from it. For instance, a manager of a mature organisation can use light management methods, whereas their use by a less mature manager may be associated with high risk.

Scalar project negentropy is another important variable affecting the IT project management environment. In complex projects, the application of light methods can be problematic, but in the course of the development of projects with a smaller negentropy, light methods may be reasonable for the project manager.

Taking into account the three variables mentioned above allows for a functionally useful and complete description of project management and for the selection of appropriate solutions. It should be noted that the evaluation methods of the client and provider environments are common, *i.e.* both standards (COBIT and ITIL presented in the Appendices) can be used to evaluate the client and the provider.

The selection of variables closes the structural modelling of the project environment. The model has to be made precise by estimating the values of these variables. Their rough and not very useful determination can be based on the CMMI model to evaluate provider organisation maturity and on COCOMO to evaluate project negentropy. The evaluation of client maturity can be carried out on the basis of a qualification of the psychological profile of the client. This book offers a more effective solution to estimate these values using the appropriate initial processing procedures, IPP. The application of the management model MITM, including the IPP processes, simplifies the implementation of information standards such as ITIL, COBIT and TOGAF for the needs of the client and provider organisations. A proper evaluation of variables expressed in the maturity capsule C-S-P substantially improves the efficiency of the implementation process of these standards.

A detailed material, being the basis for the initial processing of variables, is presented in the Appendices. Appendix 1 discusses material in the TOGAF standard, which is complementary to the specification of information technology project negentropy. Here we focus on the part of the standard concerning the description of ADM processes and the development of the Continuum (in the supporting process).

Within the ADM area, we focus on the relationship between the ADM and the Governance process: suggesting they are formed (in a given company) in a way which provides the maximum value of project negentropy. Therefore, when developing the Continuum, we pay attention to those processes which are in accordance with Governance. Appendix 1 provides the knowledge (to those who develop enterprise architecture and project repositories) on development processes and documents which facilitate the introduction of corporate governance – thus increasing project negentropy. The substance contained in Appendices 2 and 3, which provides knowledge on the evaluation of the maturity of client and provider organisations, is presented in a similar way. The basic goal is for the material to be suitable for those building their own teams (provider and client) and for the analysts who deal with modelling the processes of the client and provider organisations.

The application of initial processing, in combination with the descriptions of standards included in the Appendices, as well as their subsequent fuzzy evaluation (not considered in this study) allow for the effective application of the developed sequential multi-level model of IT selection (MITM) by team leaders of organisations which realise IT projects. In this way, we introduce new prospects of building models useful in the evaluation of client and provider maturity and in the evaluation of information technology.

In terms of the selection of methods and tools, the developed and presented MITM model of information technology management (in terms of its structure consisting of the IPP, the capsule, and the MSM) is a practical compromise between the precision and the applicability of the description. It is built using the *top-down* approach, moving from the general description to the detailed one. Further studies are obviously needed to improve, for example statistically, the determination of the degree of applicability of the management model.

In the opinion of the authors, the intention to obtain a practical model can be based on a sequence of three postulates:

(1) the fuzzy modelling philosophy, based on shaping the membership function, is a sufficient foundation of the universality of the developed model

(2) the number of times the model is used is an appropriate evaluation criterion of the quality of the model

(3) the identification of the model parameters and variables, based on linguistic evaluations resulting from competence questions makes the model more precise.

Gdansk University of Technology is working on such models. The studies mainly focus on developing examples of applying the IPP models and the maturity capsule to evaluate the maturity processes of provider and client organisations and to evaluate project negentropy. We assume that the construction of fuzzy models of multi-dimensional project negentropy, designed for those who manage information technology projects, will continue to be a priority.

In the studies on project management processes carried out so far, the state or the maturity of the project were not analysed in such a complex way. Both elements have been included in the concept presented in this book in order to predict and optimise information technologies in the management of IT projects.

The presented analysis of the maturity capsule, the possible progress in terms of maturity, and the level of management and monitoring allow for predicting technologies to support the desired changes in the maturity capsule. In this sense, the developed solution constitutes an innovative perspective on technology and IT project management processes, as it involves the aggregation of knowledge on the maturity of the capsule entities (client, project and its provider) and the decomposition of information technologies onto services and IT functionalities, and allows for the construction of tools for dynamic changes of information technologies used in the project management environment.

In the context of this work, the developed MITM model has the widest scope (integrating the IPP, the maturity capsule and the technology selection model MSM). So far, the issue of project management was analysed with the use of different models (such as COCOMO and CMMI or IT standards including ITIL, COBIT and TOGAF), which were partially predictive in nature (mainly replicative). Whereas the solution presented in this book is clearly a structural model. Due to the careful analysis of the IPP processes and the introduction of a central dynamic system (described in a state variable space) called the maturity capsule with its particular component, project negentropy, conditions for building different utilitarian (predictive) models were created. Of course, the MSM model, expressed in the form of a decision-making system, is only an example of the realisation of this part of the MITM model (*i.e.* other decision/predictive MSM models can be suggested). Nonetheless, the accepted structure of a decision-making system based on the fuzzy-rule-based method is a very general, universal and adaptable solution.

It should also be noted that a similar solution can be applied for the other two project *management components* related to changes in project teams and in manufacturing and management processes. The method indicated in Section 2.3, alternative to the fuzzy-rule-based approach for dealing with the MSM issue, expressed by the agent-based environment, can also be used to implement other prognostic models of changes in these components of management.

The solution described in this work, referring to a comprehensive project evaluation and involving the application of the maturity capsule, requires that the frequency of this evaluation, as well as the analysis of the possibility of its application in the organisation/team of the client and provider, which dynamically change during the project, should be indicated. In this case, the standard use of evaluation questionnaires may be inadequate. The development and application of a system in which specialist agents carry out evaluations would be a better solution. The purpose of such a system (see page 18) would be to evaluate the information technology project realisation environment.

At this stage, it is difficult to accurately estimate the costs of such a dynamic evaluation. However, the approach to the evaluation of maturity proposed in this work should be treated in terms of minimising the risk of a project. Controlling the current state of maturity of the client and provider organisations gives the project manager the opportunity to immediately respond to a reduction in maturity. Such a proactive support of maturity, based on a current evaluation of the capsule, allows for the practical minimisation of the risk of failure in IT projects.

A

Appendix
Project Negentropy

This Appendix includes mainly materials related to the component-based approach to enterprise architecture development, and presents a project repository significant from the processing point of view. Repository components are discussed in detail – beginning with models and patterns, and ending with their uses for business scenarios. This Chapter, therefore, is the basis for the modelling which is the subject of this work.

This Continuum presentation forms the basis for the IPP processes and is an extensive source of knowledge on repositories which support architecture development. The applied concept of a common language of description increases the effectiveness of communication. In particular this refers to architecture development with the use of COTS components. Examples of such components are architectures which were once used for the purposes of an enterprise, and in IT branch standards such as TOGAF and its own technical reference model (TRM). They could also be specific components for certain IT systems, *e.g.* CRM (*Customer Relationship Management*). The TRM is understood as a reference model included in the TOGAF Continuum, whose task is to support enterprise architecture development processes.

A.1 Specification of the IPP Processes for Evaluating Project Negentropy

In the concept presented in the main part of the book, and based on TOGAF [65], we assume that a company builds a repository of its artifacts (architectures, models, blocks and other components), which are useful during the development and implementation of architectures compliant with the ADM. The Continuum resources not only prove very helpful in architecture development but are also used in communication between the client and provider within the companies which implement the ADM. They make it possible to take into account the specific nature of companies as they provide solutions where the architectures manufactured in the ADM meet the needs of individual clients. As a result, architects participating in the manufacturing of an IT system are provided with an environment offering example solutions, manufactured with the use of a coherent language. The data presented in this Appendix was obtained on the basis of studies of ITIL and TOGAF [41, 63-65].

In the TRM, the architecture development process is described, from the generic architecture (*Foundation Architecture*) to a set of architectures specific for the company. Generic architecture includes definitions of the used approaches and management procedures, software components, common models of technology selection procedures, guidelines, templates, checklists and other detailed materials. The process of enterprise architecture development (which includes generic architecture components) requires that architecture resources and their components included in the Continuum are taken into consideration. Generic architecture (*Foundation Architecture*) is the basic form of architecture manufacturing included in the repository. Architectures dedicated to enterprise needs designed in the proper process are based on its components.

The Continuum may also develop own (generic) architectures. The structure of the Continuum provides for two reference models (which organise the resources). The first of these models is the TRM, which refers to general services and functions which form the basis for the construction of more detailed architectures and their elements. It includes bases of architecture development standards (SIB), *i.e.* basic information about manufacturing processes together with specifications and standards. The other model is either an integrated IT infrastructure model III-RM (based on generic architecture) or a boundaryless information-flow model.

The term: III-RM Model (*Integrated Information Infrastructure Reference Model*) *– an integrated model of IT reference infrastructure, whose main task is to provide support for the construction of environments for boundaryless information-flow.*

The term: Boundaryless information-flow concept – *assumes obtaining, processing and using information for the purposes of architecture development. The components of this concept are a company's employees and its IT resources.*

Detailed Description of the Continuum

The discussed concept for enterprise architecture development (ADM) assumes that the Continuum resources are used. The resources should include enterprise architectures and a coherent description of the general principles of their manufacturing and the management of manufacturing. The principles are mostly related to the method of re-using the repository artifacts. Therefore, the function of the Continuum is to support the relationships (supporting development) between the structures of the available groups of processes (such as TOGAF), shared system resources (*e.g.* III-RM) and architectures for specific industry branches and enterprises. The repository, however, is primarily a useful environment for the development of architectures dedicated to enterprise needs (within the proper design process), which incorporates the manufacturing of SBBs (*Solution Building Blocks*) and the identification of possibilities for their re-use.

The environments are created as a result of an agreement between clients and business partners on the incorporation of principles for building/manufacturing products (enterprise architectures). They support the process of model creation with consideration to differences between models, systems and the services provided by the systems. Examples of models stored in the Continuum include:

- domain-based models of business processes in the following areas: healthcare, transport, finance, *etc*. developed by the OMG (*Object Management Group* [68])
- detailed business models for the telecommunications industry developed by the TMF (*TeleManagement Forum* [69])
- models for supporting government agencies developed by the FEAB (*Federal Enterprise Architecture Business* [70])
- high-level business models for areas such as e-commerce, supply chain management, with examples which include:
  - REA model – agent-based model of business processes [71], mainly accounting systems, used as the main model for manufacturing frameworks for traditional companies, e-commerce and supply chain management [72]
  - STEP standard (ISO 10303) – standard for the exchange of data models (dedicated for the development of products and the supply chain), used by large aircraft manufacturers and in other branches (construction industry) which use complex graphics and data processing in architecture development
  - OAGIS model (*Open Applications Group Integration Specification*) – model for the integration of heterogenic business applications (in particular ERP) and supply chain management and e-commerce, made by the OAG (*Open Applications Group,* [73])
  - model of the RosettaNet consortium [74] – standard e-business processes for SCM systems (*Supply Chain Management*).

Technical Reference Model – TRM

In TOGAF, generic architecture is the basis for building architectures and their elements dedicated for companies. It includes two main models:

- TRM – which is the standard and a taxonomic description of these architectures
- SIB – which creates a database of standards that may be used for the specification of services and other elements of the developed enterprise architectures.

The TRM is commonly used for the manufacturing of IT systems models. It can be used for building IT systems architectures. The SIB, on the other hand, is aimed at organising the process of architecture development by using open manufacturing standards, such as TOGAF. The purpose of the SIB is to make resources (standards and specifications) available for ensuring the quality of the manufactured software.

TRM Model Components

The main components of the TRM model are:

- taxonomic description, which includes a coherent description of model components and the structure of the system of information on enterprise architecture
- graphic TRM, which is a visual representation of the taxonomic description, helping to understand the meaning of the TRM.

The TRM model includes definitions of standard application platforms and related interfaces. By using it, an architecture developer obtains information on the effect of potential application platforms on the development of specific enterprise architectures. The basic function is to ensure a higher level of manufacturing processes. Other components of the model (taxonomies and graphics) can also be used for the development of dedicated enterprise architectures (by adjusting to the TRM model). A solution which improves the development process may also be to re-use architectures or taxonomies. In the development process, various graphic interfaces are used (available within the TRM) which reflect the provider concept and improve the communication (during the development stage) between the client and the provider. The TRM model includes three main platforms (joined by interfaces): applications, infrastructure and communication.

The term: Application platform – *a platform of operating systems for the implementation of enterprise architectures.*

The term: Infrastructure platform – *a hardware platform for the implementation of enterprise architectures.*

The term: Communication platform – *a platform for the exchange of resources between applications functioning on the application platform.*

For connecting these entities two types of interface are used (application platform interface and communication infrastructure interface) presented in Fig. A.1. Through these interfaces it is possible to fulfil the two principal objectives of enterprise architecture development:

- transferability of applications by defining the functionality of their interfaces and specifying the set of services available,
- communicativeness of applications by defining the communication infrastructure services performed through the interfaces.

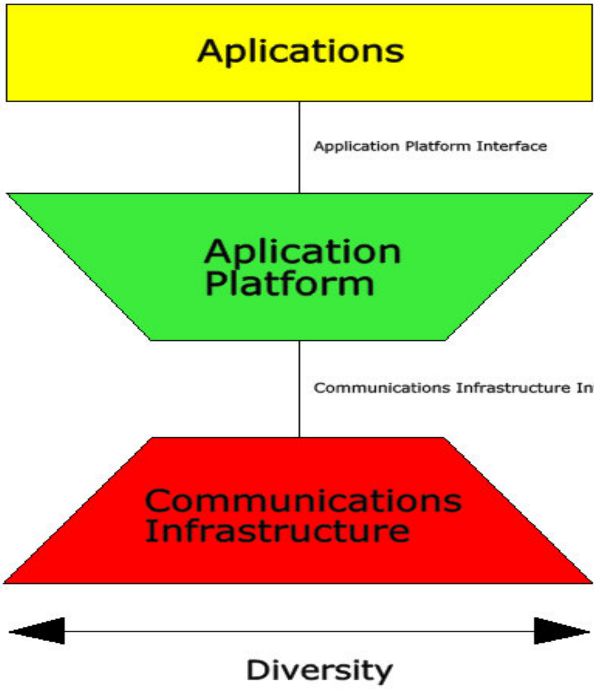


Fig. A.1 TRM model components

The TRM resources (applications, application platforms and communication interfaces) enable the manufacturing of varied applications – a set of services based on broad networks as the environment for building a communicative corporate background. Next we present a detailed report on the TRM model, which includes its main modules and the relationships between them.

Entities and Interfaces of the TRM Model

The TRM model includes three entities and two interfaces (Fig. A.2). Its entities are:

- applications dedicated to the company (or a group) which implement business processes
- application platforms, whose main function is to support the building of applications
- communication infrastructure, whose key function is to ensure the basic services required for the cooperation of systems and supplying the mechanisms for data transfer.

Application Platform

Applications are a company's domain. Communication infrastructure focuses on using mechanisms of 'low-level' cooperation. An application platform requires a broader overview. The TRM treats the application platform as a 'set of high-level functions' which are useful in fulfilling a company's business needs in the form of suitable architectures. It supports only those services which are indispensable for architecture development in accordance with the ADM. For the construction of a group of dedicated target architectures, the application platform structure is not homogenous. It is a combination of different entities intended to perform various functions (of specific architectures), such as providing a file server, printing, applications or internet services. Each of the functions is supported by specific services. A company's IT systems can also perform advanced services. In such a situation, the key task for an IT architect in the course of process analysis (with the objective is to build a coherent application platform) is to create a vision of the architecture supported by application platforms (existing or new). An architect should analyse services which fulfil the company's business requirements and choose a set of SBBs (real platforms) for the development of the technology architecture.

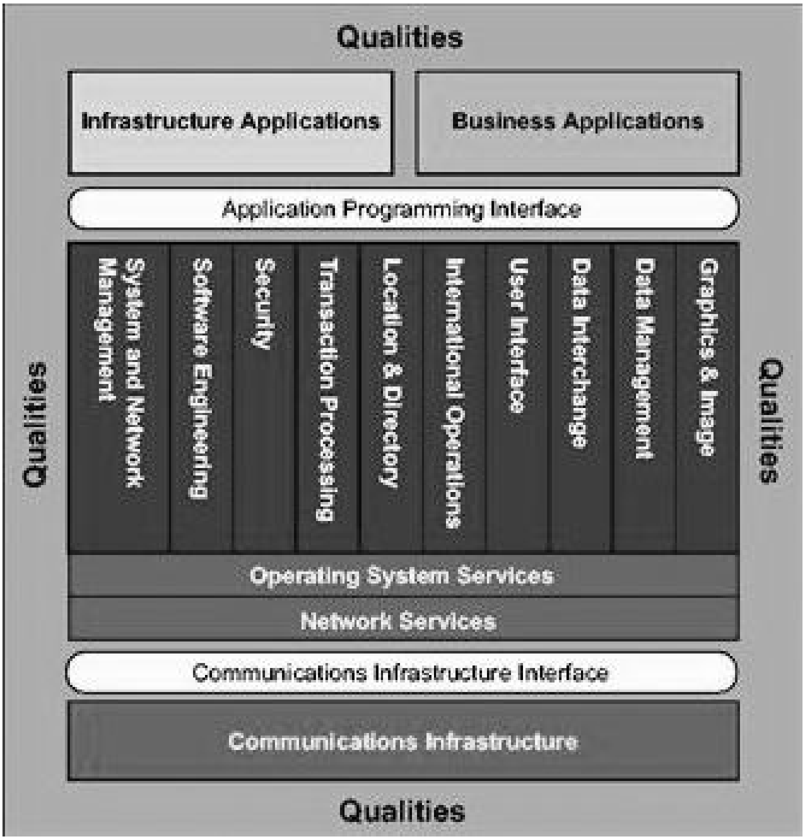


Fig. A.2 Detailed TRM specification

Within the TRM model, a general service platform is defined and required taxonomies are provided (division of services into categories and functions). An organisation can extend the scope of services using the TRM resources. Therefore, the set of those services (identified and specified for an application platform) may vary. New services are added to the TRM if new technologies appear. The API (*Application Protocol Interface*) can also be added, which is meant to create conditions suitable for the cooperation of applications.

The term: API functions (*Application Protocol Interface*) – *a type of IT system functionality whose purpose is to ensure cooperation between applications.*

It is also possible to manufacture dedicated interfaces similar to the API. The main objective is the improvement of the processes of service architecture development in the case of modules which are to be superceded (providing the same services with the help of the same API functionality). Where the API is available specific interfaces should not be used. Examples of application platforms include: spread sheet functions: creating, selecting and presenting information (tables and diagrams) and using VBA software (*Visual Basic for Application*).

The term: VBA (*Visual Basic for Application*) – *programming language in the Windows environment for manufacturing applications which support:*

- *functions and tools for project management environments*
- *routine and complex arithmetic calculations*

with the use of the Windows application platform interface.

Using the application platform interface (API) enables services to be provided between the manufactured software and its platform. A precise definition of the interface is needed to ensure compliance of the platform and the applications. It should include an explanation of the interface syntax and semantics as well as all necessary definitions of protocols and data structures.

The possibility of transferring resources with the help of the API interface depends on the compliance of the platform and applications. The platform serves the specific API interfaces and, in such a way, integrates applications. The API specification defines the communication method between the application and the basic services offered by the application platform. For this communication the application may also use different APIs.

Communication Infrastructure Interface

The interface of the communication infrastructure connects it to the application platform. The purpose of the TRM model is to support the basic set of services. The use of communication via an IP network plays a fundamental part in the development of architectures. It turns out that during the construction stage, interfaces are an important source of requirements related to technology and service architecture as well as to the selection of standards for application platforms (which have an influence on the design of EA).

Apart from the components already discussed, the TRM also includes a set of features or attributes which are significant for the evaluation of those components. For example, rendering management services may be effective on condition that the services offered are of high quality for all platforms, applications and communication infrastructure. These features must be well defined. Some are easy to describe and use. The location of a data set, for instance, may be defined as a specification element and a feature. Some features are described through standards. The standard-based approach is required, for example, to define system capacity.

Application Platform – Taxonomy

Below we present the taxonomy and principles for an application platform. Within the TRM model, two main principles can be distinguished:

- taxonomy defines the terminology and constitutes a sound description of the model
- graphic TRM is a visual representation of taxonomy.

The use of TOGAF, and especially the TOGAF ADM, is not dependent on using taxonomies included in the TRM. Other taxonomies can also be used for architecture development – in particular taxonomies used for previously performed architecture work. Taxonomies specific for an organisation can also be used when the TRM resources are properly adapted. Good results can be obtained by using various TRM graphic forms, especially for internal communication.

Application Platform Service Categories

Below we list the basic categories of services defined for an application platform. It must be noted that object services do not appear in the TRM taxonomy category, as all services of individual objects are included in the main categories of services. In the application platform, the following categories are identified:

- data exchange services (*e.g.* between applications)
- software engineering services (*e.g.* GUI manufacturing, using CASE tools)
- other services.

Application Platform Service Qualities

Apart from services which perform various functionalities, the platform also provides services which ensure the quality of manufactured systems and affect the operation of most functions (also manufacturing). Combining the quality and functional services (of architecture block implementation) ensures the correct progress of architecture development and affects the quality of the future operation of those blocks. Quality assurance services may be supported by software, an external environment and the application platform.

In the course of development work, an architect must be aware of the existence of quality assurance services and their influence on the manufacturing of software blocks used to build the architecture. There are a lot of quality assurance methods. One of them is to create a matrix which describes relationships between the provider's project services and the quality of the designed architectures [35].

SIB Model – Standards Information Base

Below we present a database of information standards SIB (*Standards Information Base*), which includes information on the development methods and applications of typical architectures. With regard to management, it complements the discussed TRM model. The SIB includes a base of facts and guidelines related to the management of manufacturing of IT systems based on architectures. The standards included in the SIB come from many sources: from organisations that provide standards (ISO or IEEE) and authors of authoritative standards: Internet Society, W3C [75], Object Management Group [68]. The SIB standard base has three main uses:

- architecture development – as it is a set of standards which may be used in the development and manufacturing of the architecture of IT systems,
- architecture acquisition – since the standards included in the SIB refer to technical requirements and to assuring conformance with the ADM in orders realisation
- as a source of general information about IT standards used in a company.

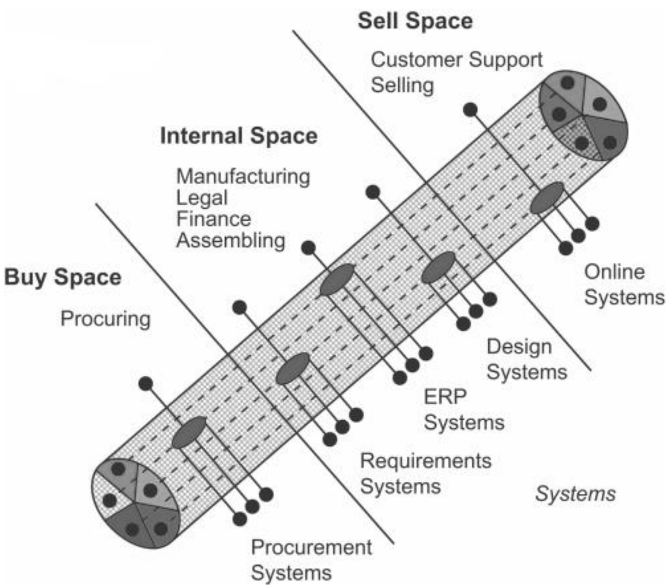


Fig. A.3 Boundaryless information flow: enterprise portals

The standards included in the SIB are classified in the open standards group. They were approved by The Open Group for use in architecture development and order specification. They are connected to the resources of other databases of The Open Group (especially those which refer to standards and registered products) and other organisations. They also offer a set of tools for defining architecture development standards and checking the availability of products which conform to those standards in the market. They can also support the process of manufacturing organised taxonomy lists of standards compliant with the SIB and approved by The Open Group. The set of SIB documents is stored in the database with user access control [76].

Integrated Information Infrastructure Reference Model (III-RM)
The purpose of this extension is to offer assistance in building an integrated information infrastructure for manufacturing a space of boundaryless information flow (Fig. A.3).

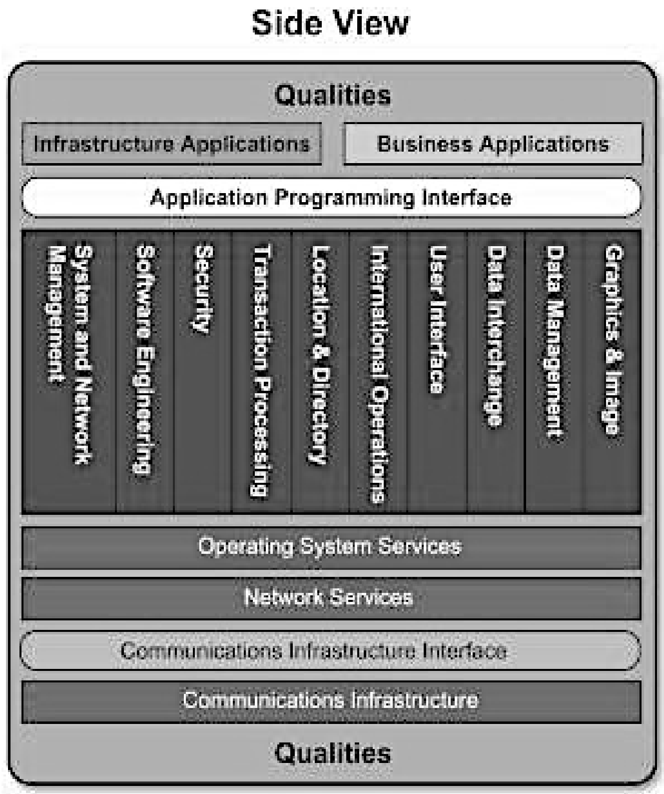


Fig. A.4 View of the TRM model

Below we present the model of information infrastructure (III-RM) which includes the concept, description and taxonomy (Fig. A.4). The III-RM model is a TRM sub-set (Figs. A.5 and A.6) and it focuses on supporting the manufacturing of applications which constitute a shared architecture of Continuum systems. To a certain extent it also expands the TRM model. The expansion particularly refers to business applications and a part of infrastructure applications.

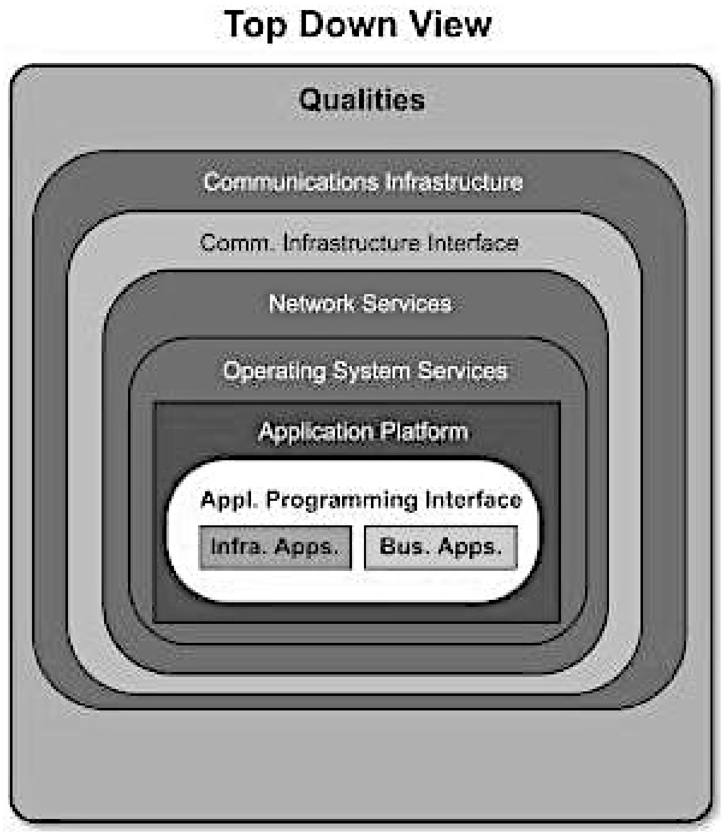


Fig. A.5 View of the TRM model with the III-RM model area marked

The development of such spaces should take place in the processes of both manufacturing and using IT systems. The III-RM model elements are:

- taxonomy – it defines the terminology and elements of the model and IT infrastructure
- graphic model associated with the III-RM, which constitutes a visual representation of the taxonomy and presents relationships between its components.

The III-RM model can be used as a tool for supporting the development of architecture compliant with the ADM. Other taxonomies/reference models can also be used with the ADM as sensible solutions for given organisations.

The components of the III-RM model support the creation of a functional space for boundaryless information flow, which is shared by users, clients and organisations that develop projects together. In this space, it is easier to acquire information from partners which is important in enterprise architecture building, and to manufacture organisation structures which enable the effective operation of individual departments and supporting IT systems. The use of the III-RM stimulates the development of technical qualifications of employees that are required for the performance of specific tasks in the manufacturing process. The III-RM model improves the process of creating cooperative environments in mixed teams (which integrate employees of various specialties, different levels of knowledge, skills and experience). It also poses new challenges for CIOs. They should provide employees with adequate access to information (in accordance with project requirements) from the level of such a mixed team. That is why dedicated IT systems are replaced with universal systems of the *Business Intelligence* type, which are based on dedicated (wholesale) databases.

The term: Business Intelligence systems *are a group of IT systems that collect and process data required by decision-makers for generating reports.*

Organisations can be given support with regard to information flow in the form of the III-RM model concepts:

- by integrating data and company information before using it
- by making the information available via one user-friendly interface.

One implementation of the concept are company portals which offer integrated information from various applications operating in a company with the use of a practical WWW interface and create a so-called 'integrated information infrastructure.' This way, today's architect fulfils the demand for boundaryless information flow.

The uses, views and placing of the III-RM model in the TRM model as well as a detailed description of the III-RM model are presented in Figs. A.3, A.4, A.5 and A.6, respectively. Figure A.3 shows the III-RM model as it fulfils the demand for boundaryless information flow in a company, which is very vital for collecting and processing information on clients and providers.

The views of the TRM model, together with a general division into layers (including 10 vertical III-RM blocks) is presented in Fig. A.4. In order to ensure hierarchical placement of the III-RM model in the TRM, in Fig. A.5 we demonstrate 6 layers of the TRM model including: the main features of the TRM (adaptability to various software development platforms, applications under construction, and ensuring data safety), the communication infrastructure together with its interface and two groups of services (based on the network and on the operating system). Layer six is the III-RM model, on which applications are based (business and infrastructure with relevant interfaces). Fig. A.6 illustrates the III-RM function in the TRM model, showing the structure of the III-RM model (layer six in the TRM).

The term: IPA applications (*Information Provider Applications*) *are information provider applications whose purpose is to ensure that clients have access to data managed at the server level and to provide answers to questions asked by clients.*

The III-RM model consists of the following three elements (Figs. A.4 and A.5):

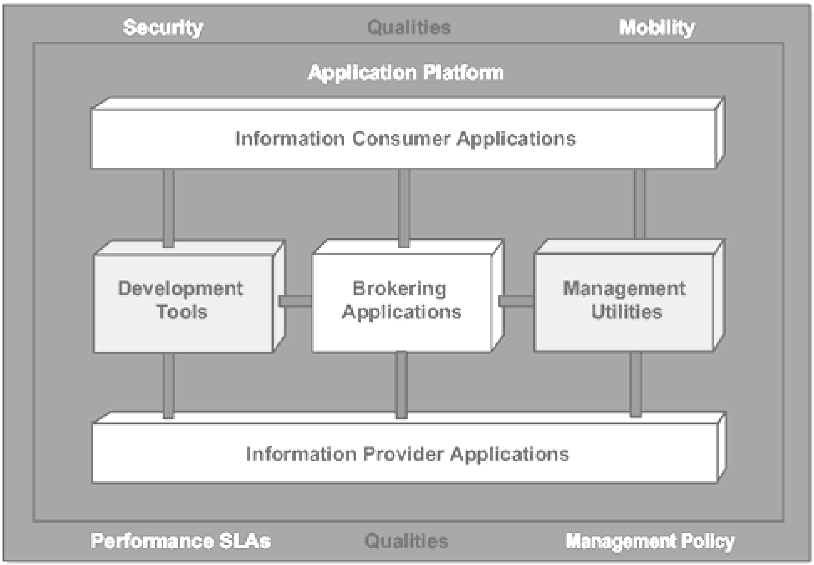


Fig. A.6 Structure of the general III-RM model

- **Business applications** – have two forms of usage:
  - applications of consumer information, which provide the user with the contents and offer access to information on behalf of the user
  - agent applications, which manage the applications of clients and the IPA.
- **Infrastructure applications** with two application types (infrastructure information):
  - manufacturing – *developer infrastructure*, which provides functions indispensable for modelling, developing, building, expanding and implementing
  - management – a manager's application with tools for planning, monitoring and system control (significant in the light of changes in a company and systems).
- **Application platform** (discussed above in the TRM model, Fig. A.4) is a resource of services for all the above-mentioned applications (in areas such as resource location, catalogue, work flow, data management, *etc.*) and it enables locating, accessing and sending information to an integrated information infrastructure.

Interfaces for application programming (the TRM model, Fig. A.5) between the platform and the infrastructure and business applications, including formats and protocols, application programming interfaces (API), data, *etc*.

The model specification includes rules for quality assurance compliant with the basic quality plan (*Backplane*) and procedures included in the SIB.

Taking into account the general III-RM model presented in Fig. A.4, its detailed model is demonstrated in Fig. A.7. It includes three main entities: the application platform, consumer and provider information applications, and agent applications, which will be described via examples:

- **Consumer information applications** are WWW portals, audio/video stream applications, videoconference applications, mail, *etc*. To use them, an environment is required for collecting and storing client information, *e.g.* project repositories, libraries or external system memories. The standard for information transfer is ensured by adopting formats for the communication and integration of applications.
- **Agent applications** and supporting tools appear in the description for the TRM model. They are complemented by information *brokers* (seeking information and making it available) and libraries of tools supporting manufacturing and management.
- **Provider information applications** are used (in addition to the above applications typical for the client) for the verification of electronic signatures, encrypting, access management and keys. They also include applications which support the processes of information mapping and the distribution of inquiries referring to the system.

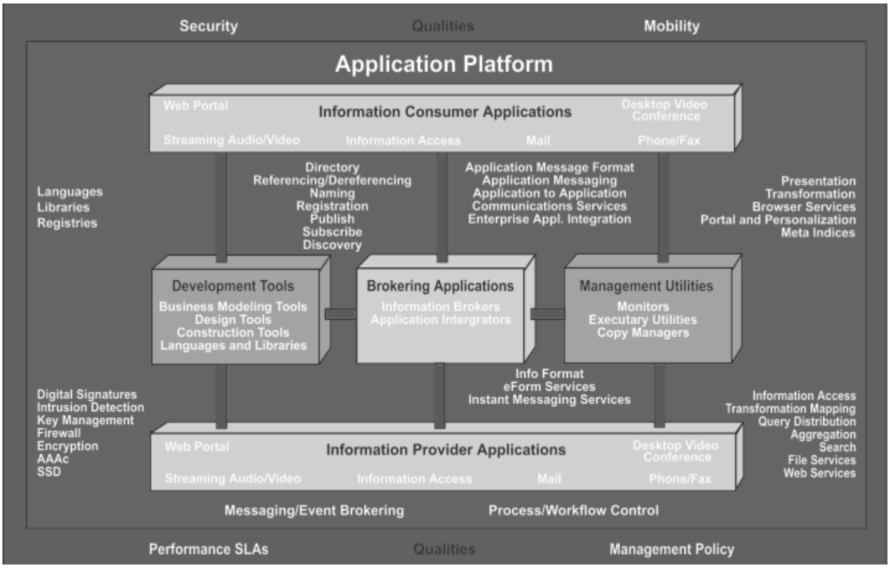


Fig. A.7 Detailed III-RM model

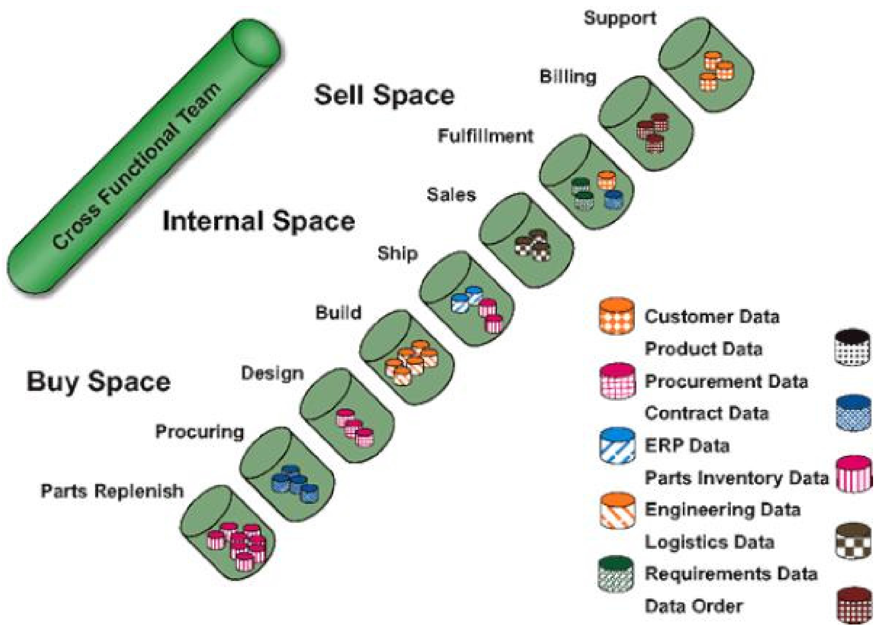


Fig. A.8 Silos providing information for mixed multifunctional teams in a company

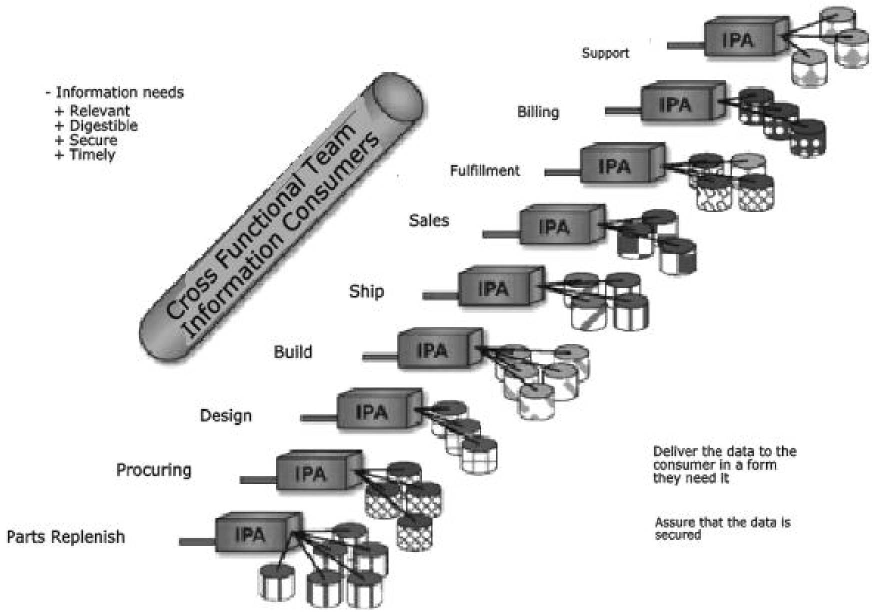


Fig. A.9 Scheme for team access to data through a standard parameterised interface for LAIP application silos

In order to illustrate the possibilities for using the III-RM model resources, let us consider the example of a model enterprise application. The reference model of this application reflects the structure of the shared environment offering a set of services for the purposes of end users and access to heterogeneous database and file systems.

The application concerns providing information to satisfy a client's need for information. It also justifies the establishment of a mixed team (of specialists with expertise in different areas) for acquiring data from the company's information silos. Fig. A.8 presents the structure of such silos, the location of mixed teams and three areas of operation: procurement, internal and sales.

Fig. A.9 presents processes which are supplied with data from information silos through a parameterised LAIP interface for IPA applications.

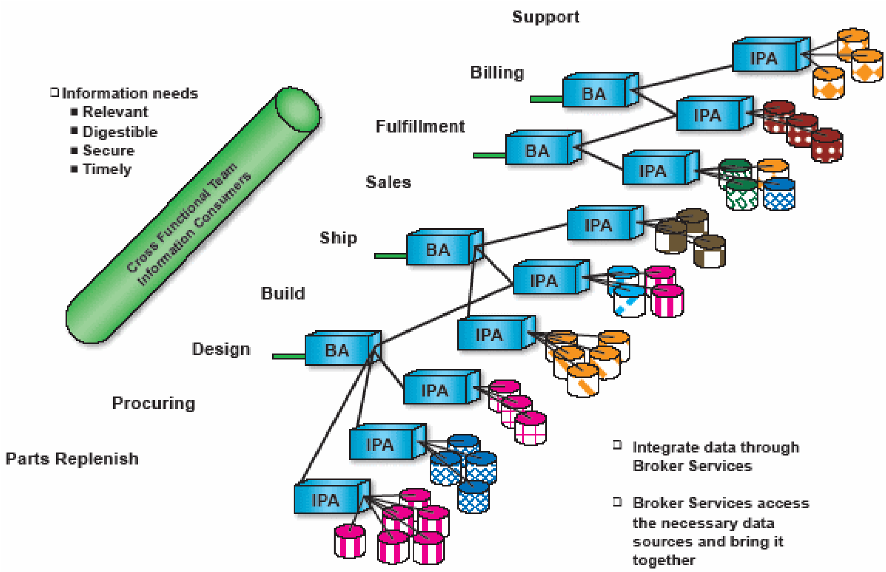


Fig. A.10 Typical enterprise application integrating data through brokering services (example of a consumer information application)

The term: LAIP applications (*Liberate Application of Information Provider*) – *a type of free application whose main task is to provide clients with information (through a parameterised interface) from the resources of heterogeneous information systems.*

Figure A.10 (top left) provides a description of the operation mechanism of a parameterised interface for the purposes of manufactured applications and the concept of open interfaces (Fig. A.11 shows interfaces not included in Fig. 3.10).

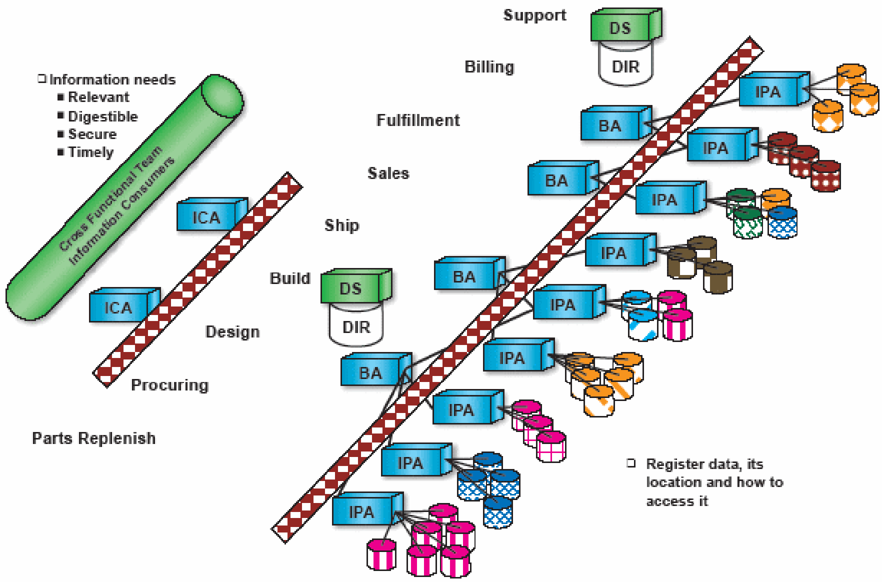


Fig. A.11 Application with communication via open LAIP interfaces and the ICA protocol

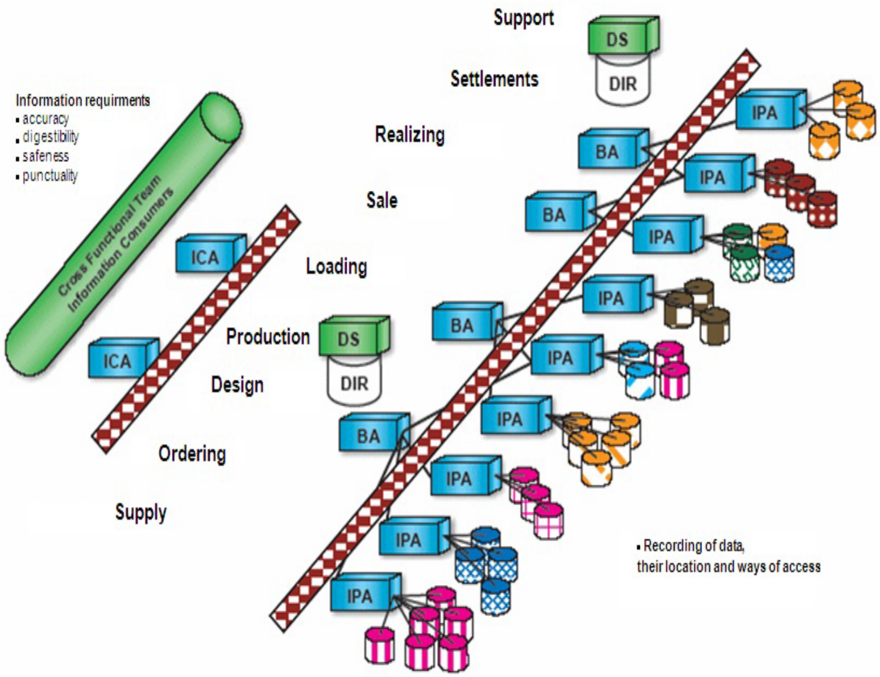


Fig. A.12 Environment for the development and implementation of catalogue services

The Broker House application is used to handle single requests (inquiries) which may be received from different sources. With the help of the application a request is analysed, and responses are collected and sent to the asking client. The task is performed through applications that use open interfaces (provided by the IPA), which also enable a mobile user to access information in a company through a strategic partner.

One example of open interfaces are applets provided with browsers, *i.e.* applications started by any user of an internet browser which offer him/her access to a company's applications through any partner that uses own applications integrated with the company's systems on conditions specified by the IPA.

On the other hand, consumer information applications provide information in a form which is most suitable for users. They communicate via an IPA-type application using open interfaces for brokerage services (BA) which integrate data. In Fig. A.12 we present a typical application with data integration performed with the use of brokerage services.

The term: Brokerage services *is a solution which is used for the management of complex services. Within this solution it is possible to define simple and complex services, and register and perform them. By using a broker it is possible to launch services of adequate quality, in the proper sequence, with the use of a safe communication bus.*

The structure of a company's model application presented in Fig. A.13, when compared with Fig. A.12, has been completed with data processing with the use of the ICA protocol (*Independent Computing Architecture*), which sets the standard in the market for IT systems. The ICA technology usually includes software supporting server functions, the relevant network protocol and client software. The task of the ICA is to separate the logic circuit of an application located on the server from the user interface. This ensures the effective operation of applications, their management and suitable technical support.

Data expected by a client is delivered at the right time and in the right form. The ICA solution ensures a high level of data safety. Suitable tools for the modelling, development and building of an integrated information infrastructure are required for manufacturing such applications. In particular, the tools to be considered are tools for business processes, their modelling, the definition of data, and also traditional manufacturing tools for transforming a business model into applications. The model of integrated information infrastructure (due to its distribution and the need to engage a mixed team for its implementation) requires the use of tools which automate business processes. Below we describe the characteristics of selected tool groups corresponding to the SIB and III-RM.

Tools for the Modelling and Development of Business Processes

This category includes tools intended for modelling rules and business processes which enable the establishment of relationships between the rules for manufacturing process management and the requirements related to information and business environment. Development tools are used for defining, designing and documenting the most important elements in the company's activity, in accordance with the rules and models of business processes. This way libraries containing software components (elements ready to be re-used) are developed.

Tools for System Manufacturing

Using such tools (*Development Tools*) enables the manufacturing or re-using of components of processes, applications and application services. The group of tool includes: intelligent browsers, applications for data processing, compilers and debuggers for programming languages, and also tools for optimising their operation. Also to be listed here are tools for manufacturing distributed heterogeneous client and server applications as well as tools for script generation. An example can be tools of the *Quality of Service Manager Utilities* type, which support the monitoring of services and data flow between any applications working under any operating system.

The term: Quality of Service Manager Utilities – *IT tools whose main function is to support and monitor the quality of services offered by an organisation.*

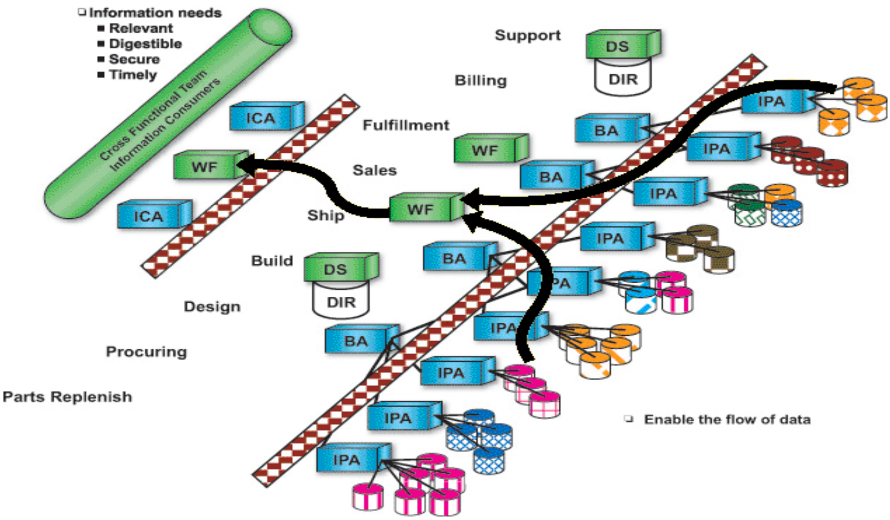


Fig. A.13 Work flows in a model application of a company's additional services

Tool Management

The selection of tools for the modelling and implementation of processes should result from their evaluation. Thus procedures are used which include management (tool selection and planning) with consideration to the criteria of their availability, cost, and suitability for the project.

Application Platform Services

All of the tools described above can support application platform services. The III-RM model, as a component of such a platform, includes a subset of services defined in the TRM. It refers to the client services of the integrated information infrastructure (which includes applications with an open IPA interface). Thus data transfer between applications is supported (with the use of one environment – operating system). In the III-RM model the services are grouped into categories:

- **software engineering services** (manufacturing support) including programming languages, library resources, legal resources and security services (the location of service delivery and access to service objects are described, Figs. A.12-13), and those supporting the manufacturing of catalogue structures and the implementation of applications
- **system services** (tool support and other services): notification register, data publication/subscription, organisation of identifiers of software versions.

Data exchange and Management Services

Data exchange services include: information and data formatting, form manufacturing, using and processing messages. They are used in communication applications and applications for the integration of a company's services. Data management services include: access to data, information and files, transformation methods, inquiry distribution and browsing. They help to integrate data, independently of whether they exist in a traditional (*mainframe*) or a distributed system. On the other hand, services whose purpose is to obtain access to data are performed with the use of open interfaces. Other IT system services are also used which include: agent services for incident identification and work flows (Fig. A.13). An example of this category of services are work flow services within the model company application. Incoming data from information silos is processed by work flow processes and directed to teams. The aggregation of data in information brokers makes it easier to define and implement the work flow services.

Architecture Patterns

The long-known concept of patterns in software engineering is defined as a group of implemented solutions which are used for building architectures or IT systems. They can also be treated as methods of redefining a problem and listing a company's structures (as reference models) and of communication with a group of target clients. The patterns can be defined as elementary blocks or groups of blocks (models) with information on the method of manufacturing and usage. Such patterns can be stored in the Continuum resources and the possibilities for combining them with available architectures and solution blocks (ABBs/SBBs) can be presented. Within the repository, various formats of pattern description are used, but none of them has become a common standard.

A format for pattern description should include the name, context of usage, its manufacturing cost, and the method of solving a problem with the use of the pattern. It should also specify the actions of developers and the possibilities for their collaboration on software manufacturing and hardware configuration. It can include guidelines related to the manufacturing of patterns, their varieties and justifications. In addition, it should include conditions for using the patterns and possibly indicate other, more efficient patterns that can be used for developing new architecture.

The term *project pattern* is often used with reference to the pattern of software, development, manufacturing and implementation architecture. As a rule, three types are identified:

- architecture pattern – expresses the basic organisation structure or application and computer system scheme (a set of ready-made elements, their mutual relationships, and also the rules and guidelines for manufacturing relationships between them)
- project pattern – defines modules of system applications and relationships between them
- *an idiom* – the level of a pattern specific for the programming language – it is described as a method of the implementation of components, their parts and the relationships between them with the use of language functions.

The above division is helpful when using patterns for building application architecture. Taking into consideration the main cycle of ADM architecture development, the use of patterns is necessary at each stage of architecture development. They should be available in the Continuum. The patterns are useful in the development of architecture views, which are chosen components of models representing a complete architecture system but viewed in those aspects which refer to a larger number of project partners. Architecture patterns may also be used while working on business scenarios. For instance, IBM patterns for e-business are a group of modules whose aim is the acceleration of the manufacturing process of e-business applications. The justification for IBM manufacturing these patterns is to:

- provide a simple and coherent method of translating business priorities and requirements into technical solutions
- accelerate the process of the development and integration of architectures, and minimise the use of non-standard solutions (difficult to implement)
- acquire knowledge and best practices from experts and make them available for use by less-experienced employees
- facilitate the re-use of intellectual capital with reference to architectures, manufacturing frameworks and other resources.

IBM defines five types (patterns):

- business patterns (the SIB and TRM included in the Continuum), which support the task of identifying a company's basic activity (*i.e.* business and IT processes) and describe interactions between the business processes and IT technologies
- services offering user access to transactions (on a 24/7 basis)

- cooperation of users with respect to access to data and information
- collecting information from many sources, cataloguing it and presenting it to the client
- integration of data and processes into extended enterprises.

Three types of integration plans are made out, which combine business models into groups:

- access integration, which focuses on access to business functions (typical functions for such integration: single sign-on, personalisation, access coding)
- application integration, which consists in combining databases and systems, based on:
  - complex patterns, which were pre-prepared, identified and collected in the repository (intranet portals of enterprises, e-commerce, WWW sites based on the ASP and the .NET platform)
  - application patterns which contain the main application elements and interactions between those elements, and also data and the method of its integration into application elements
  - application patterns which provide for qualities such as effectiveness, efficiency, scalability and accessibility, as well as the identification of key resources and best practices (the IBM service provides the mapping of applications to application patterns, indicating specific performance technology).

The term: ASP (*Active Server Pages*) *is an application of the .NET platform library for supporting the construction of interactive WWW sites.*

Architecture Views

The architecture views defined above constitute chosen parts of one or more models representing a complete architecture system, but viewed in those aspects which relate to one or more project partners. An architect selects and prepares a set of architecture visions, which should be approved by all project partners, *i.e.* providers, clients, and other participants of the enterprise architecture development process. The aim of preparing such views is also to verify if the system is going to meet their requirements. For architecture is represented by several models, which together ensure a complete and coherent description of system architecture. When developing the architecture of an IT system, a multi-aspect integrated vision is prepared which includes many fragmentary views provided by different partners, so that the created system can be approved by all interested parties.

Views can effectively support the development of various types of architectures (business, data, application and technology architectures), which with the presented approach to development (the ADM + Continuum) increases the significance of views in the categories of the main process. They may be effective in the case of implementing individual projects, such as for instance the development of only business architecture or application architecture. They are also useful in implementing repeatable projects (they improve the process of establishing project requirements).

The implementation of individual EA projects is often performed with no intention of preparing all four architectures or of incorporating them into the general structure of the company. Four architectures are developed only when the long-term strategic aim is known. In the case of no strategic aim, the choice of approach depends largely on the company's outlook on its architecture.

The use of many different views is of significant importance both for the development of chosen architectures (in the case of the company not being convinced as to the need for having an enterprise architecture) and also for the full architecture implementation.

Views are defined in accordance with the ANSI and IEEE Std 1471-2000 standards. They include the chosen elements of one or more models in such a way that the partners' demands can be presented. Views, which can also be treated as perspectives, define: viewpoints (how to prepare and apply them with the use of diagrams and templates), information (which should be displayed in the view, techniques for expressing and analysing information), and a justification of the choices (*e.g.* by describing the purpose and handing it over to the partners). Examples of viewpoints and views (based on the example made available by The Open Group) are presented in Fig. A.14.

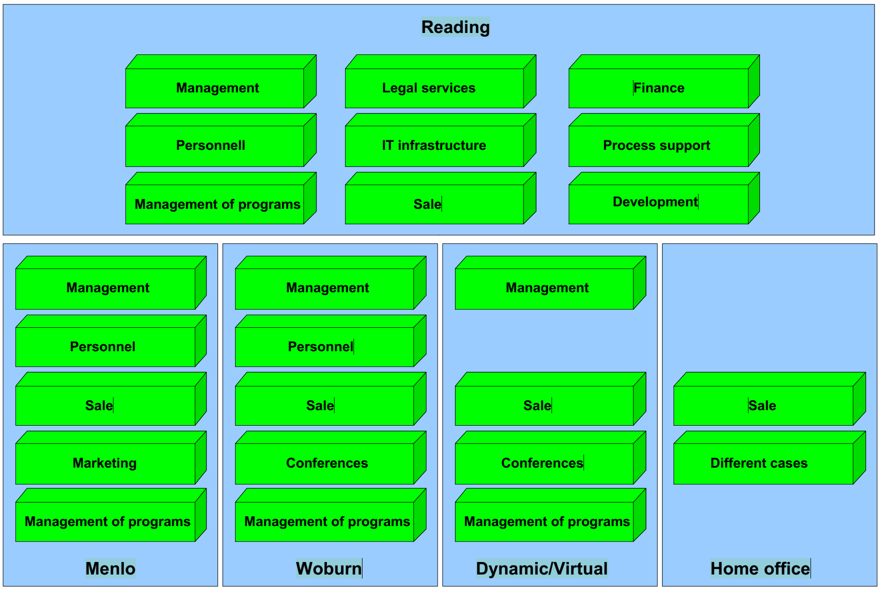


Fig. A.14 Example of views for business domains

Architecture Views in ADM Processes

The choice of architecture views for development purposes is one of the key decisions to be made by an architect. He/she is responsible for ensuring the completeness of architecture (making it suitable for its purposes) with respect to all significant issues related to the system structure and the partners' concerns as to its integrity (the need for the approval of different views of one system). He/she should do it in a way which ensures the safety and effectiveness of view integration (with an iterative manufacturing method).

In the course of architecture development (from technology to business), relevant manufacturing techniques are used, such as business scenarios which support the process of view manufacturing. This is especially important when 'high level' views (of low granularity) implicate a reference to conditions related to processes implemented by different partners. Each of the views should be presented in two environments: base and target environments. Obviously, the architect's opinions are then required about the business and technology architectures for both systems (base and target). This simplifies the gap analysis, that is the evaluation of which views have not yet been analysed, and the indication of current system elements which need to be moved, removed or replaced.

The term: Gap analysis – *the process of analysing company requirements which are not included in the presented views of enterprise architectures.*

View Development Processes and Their Taxonomy

The description below refers to a situation where development processes are supported by the standard ANSI/IEEE Std 1471-2000, which does not specify any particular process for the development of architecture views, and also to an example without such support.

Using the standard in an organisation makes the creation of required architecture views easier through:

- reference to the existing library (repository) containing views
- selection of a suitable view (considering the partners' conditions)
- view development (in the event of their lack in the repository).

This way we gain the following:

- minimisation of architects' work (a quick opinion for the chosen view)
- better understanding of partners (with known views)
- partners more confident with established views.

The taxonomy of architecture development processes defines the minimum set of views in ADM processes. Since each view should provide for different viewpoints, taxonomies can be considered as sets of viewpoints in an organisation. The minimum set of partners (who provide viewpoints and views) includes:

- users
- system and software engineers
- operators, administrators and managers
- buyers.

ADM – Architecture Views

The architecture views discussed above are used for the architecture development process within the ADM framework. The views should be available in the Continuum and support the main manufacturing processes. On the basis of the TOGAF specification, it was established that architecture views can belong to the following categories:

- business architecture views, which refer to conditions of system users and also describe the information flow between partners and business processes
- data architecture views, which refer to the manufacturing environment of the databases of developers and database administrators and engineers who are responsible for the development and integration of databases using different system components
- application architecture views, which refer to the conditions of software engineers, responsible for the manufacturing and integration of various components of system applications
- technology architecture views, which refer to the conditions of buyers responsible for the acquisition of COTS components for software and hardware, which may be included in the system and the activities of employees, system administrators and system managers.

Examples of specific views which may be manufactured in each category are presented in Tables A.1 and A.2 and later explained in detail.

Table A.2 presents an example of mapping those views into the scheme of the Zachman framework. The example is presented in order to demonstrate the usefulness of the views for enterprise architecture development. The Zachman framework constitutes an exemplification of a tool frequently used by enterprise architecture developers. As the tool is supported by system views, the views evidently form an important interface for the collaboration of partners in developing an information technology project.

Table A.1 Example of a taxonomy of architecture views

<table>
<tr><td colspan="4">Interested project parties – project partners</td></tr>
<tr><td>Users, planners, business management</td><td>Developers and database administrators, system engineers</td><td>System and software engineers</td><td>Buyers, operators, administrators, managers</td></tr>
<tr><td colspan="4">VIEWS</td></tr>
<tr><td>Business architecture view</td><td>Data architecture view</td><td>Application architecture view</td><td>Technology architecture view</td></tr>
<tr><td>Business functions view</td><td rowspan="5">Data view</td><td rowspan="5">Software engineering view</td><td rowspan="3">Network processing / hardware view</td></tr>
<tr><td>Business services view</td></tr>
<tr><td>Business processes view</td></tr>
<tr><td>Business information view</td><td rowspan="3">Communication engineering view</td></tr>
<tr><td>Business locations view</td></tr>
<tr><td>Business logistics view</td><td rowspan="4">Data flow view (using organisation data)</td><td rowspan="4">Application interoperability view</td></tr>
<tr><td>Work flow view</td><td rowspan="3">Processing view</td></tr>
<tr><td>People view (organisation scheme)</td></tr>
<tr><td>Usage quality view</td></tr>
<tr><td>Strategy and business objectives view</td><td rowspan="5">Logical data view</td><td rowspan="5">Software distribution view</td><td rowspan="3">Cost view</td></tr>
<tr><td>Business objectives view</td></tr>
<tr><td>Business rules view</td></tr>
<tr><td>Business incidents view</td><td rowspan="2">Standard view</td></tr>
<tr><td>Business achievements view</td></tr>
<tr><td></td><td colspan="3">System engineering view</td></tr>
<tr><td colspan="4">Enterprise safety view</td></tr>
<tr><td colspan="4">Enterprise management process view</td></tr>
<tr><td colspan="4">Enterprise service quality view</td></tr>
<tr><td colspan="4">Enterprise mobility view</td></tr>
</table>

The description provided in Table A.2 explains the usefulness of some views. Their more detailed description and guidelines for using the views in the ADM can be obtained by directly referring to:

- business architecture views created as a result of the collaboration of users, planners, managers and companies focused on the functional aspects of a system from the point of view of its users
- support for the process of defining the efficiency, functionality and applicability of developed architectures (views may be built on the basis of an analysis of the existing information infrastructure and the requirements and limitations related to the developed IT system)
- data and application architecture views with opinions and conditions provided by database and system administrators and developers as well as software engineers (views include the viewpoints of safety engineers, software, data, computer components and communication, and also their influence on system properties, and they are linked to component re-use)
- technology architecture views reflecting the conditions of buyers, operators, communication engineers, administrators and system managers.

Table A.2 Example of a taxonomy of architecture views for the Zachman framework

	Stakeholder	Data	Function	Network	People	Time	Motivation
Scope	*Planner*	Data unit view (class model)	Business functions view Business services view	Business locations view Enterprise mobility view	People view (organisation scheme)	Business incidents view Enterprise service quality view	Enterprise strategy and objectives view
Enterprise model	*Owner*	Data flow view (use of organisation's data)	Business services view Business processes view	Business logistics view (business-to-location mapping) Enterprise mobility view	Work flow view	Business achievements view (main schedule) Enterprise service quality view	Business objectives view (SMART objectives from business script)
System model	*Developer*	System engineering view Logical data view Standards view	System engineering view System engineering view Application-to-application communication view Standards view	System engineering view Standards view Enterprise mobility view	Usage quality view Standards view	System engineering view Processing view Standards view Enterprise service quality view	Business rules view
Model with technological limitations	*Constructor*	Physical data view (*outside TOGAF scope*)	Software distribution view	Network processing / hardware view Communication engineering view	Usage quality view	Control structure (*outside TOGAF scope*)	Business logics (rules) design (*outside TOGAF scope*)
Detailed representations	*Subcontractor*	Data definitions (*outside TOGAF scope*)	Application code (*outside TOGAF scope*)	(*outside TOGAF scope*)	(*outside TOGAF scope*)	Coordination definitions (*outside TOGAF scope*)	Application (Rule specification) (*outside TOGAF scope*)
Operating enterprise		Enterprise safety view Enterprise mobility view Enterprise service quality view Enterprise management processes view	Enterprise safety view Enterprise mobility view Enterprise service quality view Enterprise management processes view	Enterprise safety view Enterprise mobility view Enterprise service quality view Enterprise management processes view	Enterprise safety view Enterprise mobility view Enterprise service quality view	Enterprise safety view Enterprise mobility view Enterprise service quality view Enterprise management processes view	Enterprise safety view Enterprise mobility view Enterprise service quality view

In the course of an analysis of the above views, considering the presented viewpoints and relationships between them and also tasks related to resolving conflicts which result from the presentation of the views, we can see the purpose of the development of a complete architecture. When preparing such views, architects should also take into account the profitability of architecture development processes. As with all kinds of modelling processes, also here, an ADM process exceeding the adopted cost limits practically undermines the usefulness of view building.

Tools and Languages for View Building

The need for developing architecture views has been explained above. It is also necessary to present the relationships between architecture views and the tools used for their manufacturing and analysis. Moreover, it is important that a standard language be used which enables a smooth cooperation of tools. In order for architecture development to be complete and reliable, architecture views and viewpoints are usually prepared, visualised and managed with the use of suitable tools. It is a good idea to encode the architecture description with templates, in a standard language. Standards related to the methods of presenting points of view (for instance templates and diagrams) were also developed so that different tools implementing the same views could cooperate. This makes the re-use of the developed architecture elements collected in the repository easier.

Example of Views – Airport

To illustrate the terms of views and viewpoints, let us consider a system of airports from the points of view of two different project partners: a pilot and an air traffic controller. Neither of these views represents the whole system but rather only a limited scope from the perspective of one project partner. The pilot's viewpoint includes some elements (*e.g.* fuel) which are not perceived by the air traffic controller of passengers, whereas the controller's viewpoint includes other elements (*e.g.* airplanes) not perceived by the pilot. There are also elements shared between the viewpoints and views (for example the model of communication between the pilot and the controller). A viewpoint, therefore, is a model or description of the information contained in a view. A view is a set of viewpoints. All pilots should use the same model and language in which information is obtained to fill the model. Controllers describe the system in different ways, with the use of an airspace model and an aircraft traffic model in the airspace. They also use one language connected with the shared modelling area.

It turns out, however, that controllers also use one language to communicate with pilots (*i.e.* their models, representing individual viewpoints, partly overlap). For a part of the shared language refers to airplane positions and traffic, and is of fundamental importance for traffic safety. As a matter of fact, each viewpoint is an abstract model for all project partners of a given type: for pilots – a view of the plane, for controllers – a view of the airport. There are tools which support partners in creating complex views and viewpoints, such as an airspace view or flight view. A user interface similar to the language model and connected with a viewpoint is such a tool. Unique variables for a pilot are: fuel, altitude, speed, position and other indicators. The main medium for a controller is a radar, while the radio is a shared channel.

To sum up, any subset of project partner views (pilot and controller) can be presented. Such a subset can be described with the use of an abstract model, which is called a viewpoint, and the description is prepared in the form of a set of sentences expressed in a specialised language (such as a pilot's or controller's speech). Suitable tools are used for supporting data exchange between project partners, and the related interface structure depends on the viewpoint. If project partners use shared tools (*e.g.* a radio for communication), the ability to use the common language of system description is indispensable.

Example of Views – IT Systems Manufacturing

Two subjects are considered which participate in the processes of IT systems manufacturing and usage: users and *developers*. The view of neither of the subjects represents the entire system since each of the subjects has a narrower perspective. The user's viewpoint includes all the ways of interacting with the system, with no consideration to detail such as the application structure or database structure. In the case of the developer, the viewpoint includes aspects such as productivity, but it does not include data provided in a real-time connection with the user.

There are also aspects which are shared, such as process descriptions which are included in the system and/or communication protocols. In this example, one viewpoint constitutes a description of the way a user sees the system (from the point of view of its operation). On the other hand, the provider considers the system from the point of view of the processes of its manufacturing. Users describe the system in accordance with the availability model, taking into account reaction time and access to information with the use of a specified language. Contractors see the system differently from users, using a model which connects hardware and software, *etc*. Therefore, there is a need to use one language and shared tools for architecture description. Such a solution may be provided by separate views and their joint analysis prior to their implementation in the enterprise architecture.

Architecture Blocks

Program blocks are a component of architectures (initially discussed in Section 3.2). In order to explain the concept of blocks and their role in building architectures, we present a more precise description of the concept together with an example, divided into three main parts:

- introduction to the concept of block building, with an overview of the overall concept of block building and an explanation of the differences between architecture blocks (ABBs) and solution blocks (SBBs)
- presentation of blocks in ADM structures
- an example of block building and their significance in ADM processes.

Below we present typical block characteristics in the category of manufacturing methods. It includes a packet of functions isolated in order to fulfil an organisation's business needs. The way in which data will be used for block building (functions, products, the ADM and non-standard methods) depends on the type of architecture. Each organisation must decide for itself which layout of blocks is most favourable. The right selection of blocks may lead to improvements in the current integration system, *interoperability* and flexibility with respect to creating new systems and applications. Since systems constitute a set of blocks, most of them must cooperate with each other. Publishing the data on interfaces makes joining the blocks easier. The blocks are constructed on different granularity levels, which depend on the EA development stage. For instance, at an early stage of block building, a block element can consist of a group of functions (like a client database) and tools for the acquisition of (client) data. At this functional level, block building is assigned in TOGAF to architecture blocks (ABBs).

Architecture blocks (ABBs), collected in the Continuum and selected in the course of architecture development within the ADM, can be described by:

- a specification of the function they will perform
- a definition of the scope of business and technical requirements
- awareness building of the need to use the ADM
- direct use of SBBs.

The specification of ABBs includes:

- basic functions and features with a semantic description which includes safety and management possibilities
- a description (also useful for the client) of the established set of interfaces (hardware, API, and the format of data, protocols and standards)
- a definition of relationships between blocks and functions and user interfaces
- a map for business (organisation unit) and policies applied.

For the purpose of manufacturing ABBs, solution blocks (SBBs) are manufactured, which (similarly to ABBs) in the supporting process are collected in the Continuum and may be applied and developed. Their characteristics include:

- a definition of products and components as well as functions performed by them
- a description of the possibilities for implementation
- fulfilment of business requirements
- a specification of the need to use blocks and sales possibilities.

A block description, based on detailed data about its structure and later applications, reflects:

- a detailed description of the block functionality and its attributes
- the type of interfaces used together with possibilities for their integration
- SBBs used with the required functions and interface names
- mapping of the processes of SBBs onto the structures of IT architectures
- a specification of attributes common for the architecture development environment, such as safety, management facility, localisation possibility and scalability
- efficiency of configuration
- limitations for managers
- development possibilities related to enterprise architecture
- relationships between SBBs and ABBs.

Specification of Processes of Block Building in the ADM

The introduction of blocks into the developed system architecture is related to the definition of services required in the company. In architecture development three classes of blocks are used:

- re-use blocks, which in their structure have an inheritance mechanism
- blocks which are to be manufactured (like new applications)
- blocks which are to be purchased as COTS.

There are general rules for using blocks in the development of specific architectures:

- an architecture should include only blocks which perform required services
- a block can perform one, a part of or all of the services specified within the architecture
- the block building process should meet the standards for service implementation.

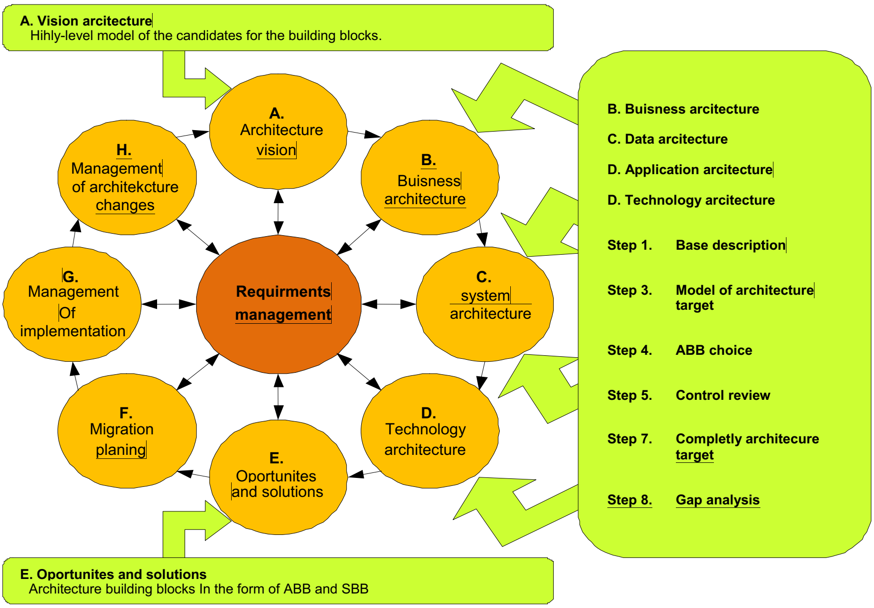


Fig. A.15 Key stages and iterations in block design in ADM processes

The process of block building occurs mostly in stages A, B, C and D of the ADM processes, as shown in Fig. A.15. It is an iterative process, since no detailed information is provided there in advance regarding the required functions and limitations imposed on the architecture and availability of products. In the course of building, ABBs are identified for fulfilling business objectives and enterprise objectives. The selected set of ABBs is implemented iteratively in accordance with the procedures available in the SBBs. ABBs can be purchased as COTS. It is also possible to manufacture them and consider them as non-standard solutions. Key stages and the iterative nature of block building within the ADM are presented in Fig. A.15.

In stage A, building blocks treated as abstract entities within the architecture vision should be started as soon as possible. In stages B, C and D the manufactured blocks (for business, data, application and technology architectures) are subject to iterative processes of change in order to manufacture a common structure in the following steps:

- Step 1. Basic description, which includes a list of considered blocks on the basis of analyses of their usefulness in development with the use of the ADM method.
- Step 3. Description of the target architecture model, manufactured iteratively. It provides for input data to build the target architecture with the use of ABBs and the description of the target system with ABBs. It includes justifications for each decision related to the selection and usage of a block in the target architecture.
- Step 4. Isolating non-conflicting services selected for the ABBs for the purposes of portfolio building.
- Step 5. Defining business objectives, which also includes a confirmation of the value and completeness of the model and a description of the service portfolio, as well as indicating that the target architecture fulfils enterprise objectives.
- Step 7. Defining the target architecture with the use of ABBs. Within this step, a fully defined list of standards and blocks which make up the target architecture is presented. A description is also prepared (in the form of diagrams) which includes the process of building blocks that implement enterprise architecture.
- Step 8. Performing a gap analysis (an evaluation of the usefulness of blocks not considered), the effect of which is the removal or moving of new blocks.

Levels of Block Modelling

Block modelling is preceded by their functional description on four levels (Fig. A.16), of which the first two are:

- level of enterprise business processes
- level of technical functions (with their limitations), which refer to tasks forming a part of the business processes.

In the TOGAF description, the level of technical functionality of blocks arises from the requirements set for them. They should provide for the limitations of the integration of the modelled blocks with systems architecture.

After the functional description is prepared on the above two levels, the modelling and manufacturing of blocks begins, which takes place on another two levels:

- level of the architectural model – system components are specified which will reflect technical functions and present relationships between them, and architecture components and relationships between them are described
- level of the model of applied solutions – individual products or components will be identified in the course of architecture implementation.

The creation of blocks on the four main levels is performed in a multi-step iterative process, where changes made on each level translate into changes on other levels (Fig. A.16).

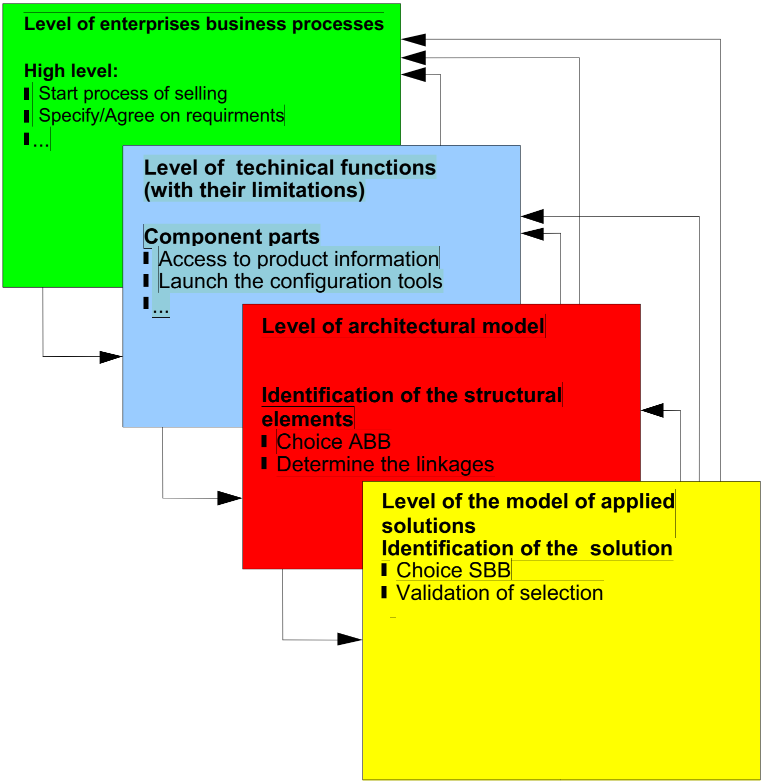


Fig. A.16 Iterative approach to block modelling: four modelling levels

Mapping of Modelling Levels in the ADM

The processes of block modelling described above will now be referred to architecture development processes (by mapping) within the ADM:

- architectures are developed in stages B, C and D on the basis of architecture models included in the Continuum in the form of architecture views
- the defining of business processes takes place in stages A and B in the ADM processes (Fig. A.15), and the conditions and technical limitations for architecture development are defined at the beginning of stages C and D; at this stage, limitations imposed on the new architecture can also be defined (with consideration to earlier solutions).

Below we present detailed data related to the mapping of block modelling processes for architecture development for the case of company XYZ. We will also present a description of the most important steps in the ADM.

Example of System Manufacturing with the Use of Blocks
The XYZ company decided to improve the effectiveness of its processes of mobile phone sales by replacing the ‘paper’ approach with fulfilling orders on the basis of electronic documents. The company team prepared a description of the existing system and revised it from a few viewpoints (establishing a number of objectives for the new system). They assumed that the main objective was for the sales process to have direct access to databases. This way a sales system can be activated in which the product configuration, prices and availability can be created and checked during direct contact with a client. Other stages in the sales process, such as initiating sales and establishing client requirements are beyond the scope of this example.

Architecture of Business Processes

In order to prepare the architecture of business processes, documents as well as processes and rules need to be considered in accordance with the description of stage B. It is also important here to establish the scope of activity and the environment of the company, the limitations and available financial resources. At this stage, redefining business processes, adopting new principles and preparing new requirements changes the system. The information collected is used for the evaluation of the base system and for establishing the yield on the investment (as a result of the changes). Diagrams of usage instances are a useful tool for describing business processes. On the basis of the described cases it was established that an improvement of the sales process involves: acceleration, quality improvement and reduction in the number of errors.

In the analysed example, neither time-related nor financial conditions imposed on the XYZ company were considered (suitability was assumed in advance). Scenarios were not used and the participants, whose roles in the process are described in Table A.3, were not considered. For the sake of conciseness, it was assumed that the scope of architecture work would not extend beyond the sales area.

Basic assumptions made by the architect of the XYZ company at stage B include:

- the need for the cooperation of the seller and client
- the client pattern available from the WWW site
- a person responsible for sales, with the possibility to close sales and synchronise own knowledge with information related to sales in the entire company
- a clearly defined set of products, which is the object of the sales process (*e.g.* car parts, IT infrastructure elements such as work stations, networking devices or external memories).

In the following steps, the business processes of sales for the XYZ company were isolated which include some of the sales processes and their support by the company's sales system:

1. start of the sales process with a client
2. discussion of client requirements
3. work with the client to create the product configuration
4. verification of whether the desired configuration can be provided
5. establishing the price for the desired configuration
6. confirmation of the intention to buy by the client
7. placing an order
8. approval by the client.

In Table A.3 a collective description of the XYZ company case is presented. Instances of usage are presented, which identify the 'actor' – participants and are shown in rows; the steps for isolating business processes are presented in columns, and the roles fulfilled – in cells.

The meanings of abbreviations used in Table A.3 are described below:

- CIPR – central system resources
- ICSys – stock control system
- LIPR – local system resources
- OrdSys – system of information on sales processes
- ProdConfig – *configuration tool* for the product system
- ProdSys – product information system
- SchSys – sales scheduling system
- $ Sys – product pricing system.

In stage A, business processes which were in compliance with detailed business requirements included:

- improvement of processing to 48 hours
- reduction of the number of order errors to three factors.

Table A.3 Use-Case of sales process

	Initiation	Discussion	Manufacturing	Verification	Price	Confirmation	Order	Approval
Sales	Greeting by client	Listening	Presentation of different possibilities	Presentation of availability to client	Presentation of prices to client	Offer	System objective	Drafting agreement
Client	Accepts sales	Discusses problems / requirements	Hears and decides on options, considers possibilities	Accepts or rejects		Accepts or rejects		Accepts or rejects
Seller			Cooperates with configuration tool	Cooperates with system provider	Evaluates		Places an order	
CIPR seller			Ensures central processing of information					
LIPR seller			Ensures local processing of information					
ProdConfig System			Configuration of ensuring sales possibilities for client needs					
ICSys System				Ensures availability				
SchSys System				Ensures delivery dates				
$Sys System					Provides information on prices			
OrderSys System							Organises and sends a fax in order to sell laptops	

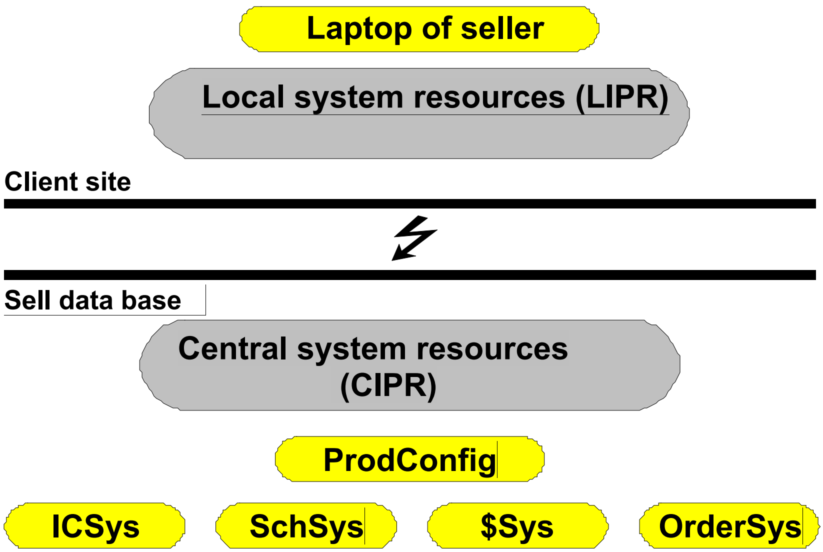


Fig. A.17 Model of blocks taken into account when creating a list of sales processes

A simplified view of blocks supporting business processes, including their location, is presented in Fig. A.17. The model was constructed of elements included in Table A.3.

Technical Functions and the Level of Limitations for Block Development (Stages B, C and D)

The aim of this stage is to prepare a high-level description of using the blocks of the existing system and of the technical functions and additional limitations. Block usage documents are a starting point for building an environment for data and information exchange between partners. The environment can be of significant importance when re-using the blocks. Technical functions and design limitations need to be specified, and the description of the analysed system, its architecture and the blocks therein should be provided (which may be derived form the evaluation of the existing, base system). The result of the work is:

- formal description of the analysed system and its functionality
- provider’s statement of design limitations (regarding blocks and architectures)
- development rules for the architectures included in the current system
- assumptions for the required technical functions
- list of blocks considered for architecture building
- model of the considered blocks.

The key feature of the work is an indicator-based evaluation of the level of enterprise architecture, expressed by two measures: the number of implemented enterprise architecture components and the progress of the development process of the implemented architectures.

Fig. A.18 presents the architecture of the XYZ company’s IT system, which includes identifiers of the architecture (price list, orders and the system of product information) and of the business process (information about the client in the form of paper cards). For the number of implemented architectures (3) and no above-mentioned business process implemented, the standard suggests indicators for block selection in order to support the architecture development process (client information system).

Since the description of measures is general (the progress level of development processes is especially difficult to specify precisely), we suggest here our own tool, which is project negentropy (described in detail in the categories of: height of the ITM section, width of the repository and length of the document catalogue in Chapters 2 and 3).

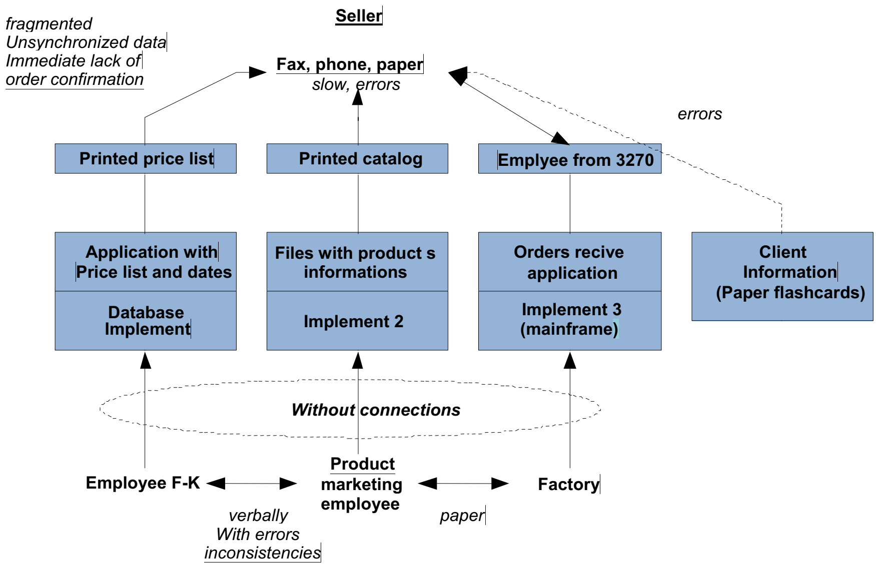


Fig. A.18 IT system architecture of the XYZ company

Indicators are also used for supporting strategic decision-making related to business and technology architectures, which should also be backed with information on:

- condition of the existing architecture
- database resources related to application logics
- safety level (important application component).

The next step focuses on business processes, particularly functionality and limitations imposed on the system. Architects of the XYZ company defined components of technical functions supporting business processes. The list in Fig. A.19 was drafted as a result of a brainstorm. Principles for the construction (function) of the system were based on it, and they specify:

- mechanisms of user access to central functions in the system
- support for simultaneous access by numerous people responsible for sales
- performance of tasks in local points of sale
- mechanism of user access to product information
- method of entering and verifying information on product configuration
- mechanisms of access to information about clients and prices
- method of goods delivery with confirmation of an order by a client
- method of ensuring a high safety level of the process.

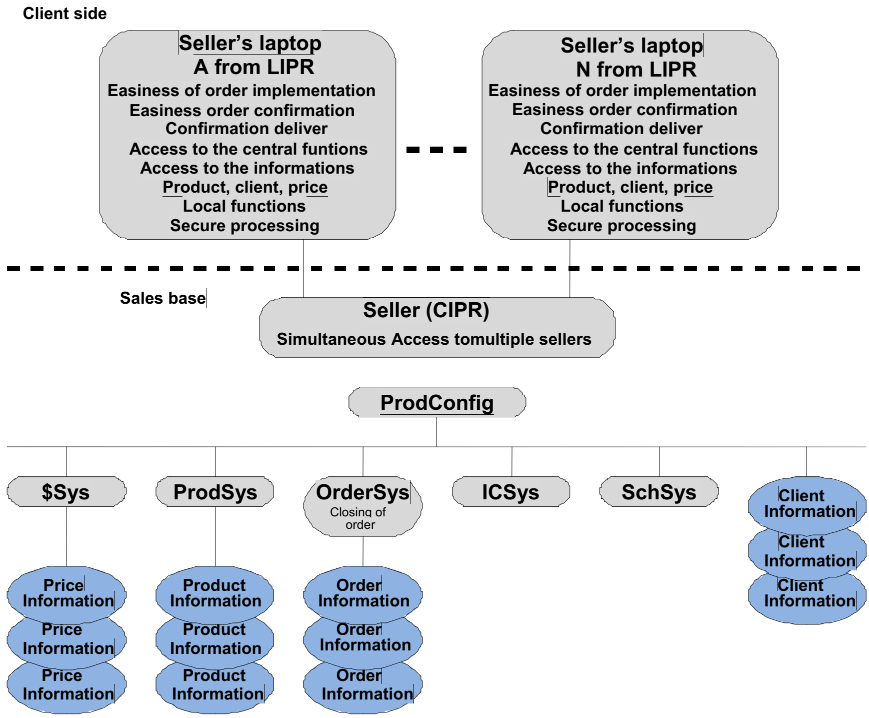


Fig. A.19 Architecture construction: block model and corresponding technical functions

An additional specification of the above-mentioned principles was also prepared with regard to:

- initiating a sales process (upon a client's consent)
- separating system functionality between points of sale and the central database
- closing a sales process in the central database
- electronic availability of product prices and information
- access acceptance and confirming orders
- combining information systems for orders and information of products and their prices.

The model of the existing system, presented in Fig. A.20, was analysed in order to prepare questions regarding its functionality. On the basis of the answers received and the discussion held with project partners, a proposal was made for the construction/selection of blocks to be considered in architecture development (Fig. A.21).

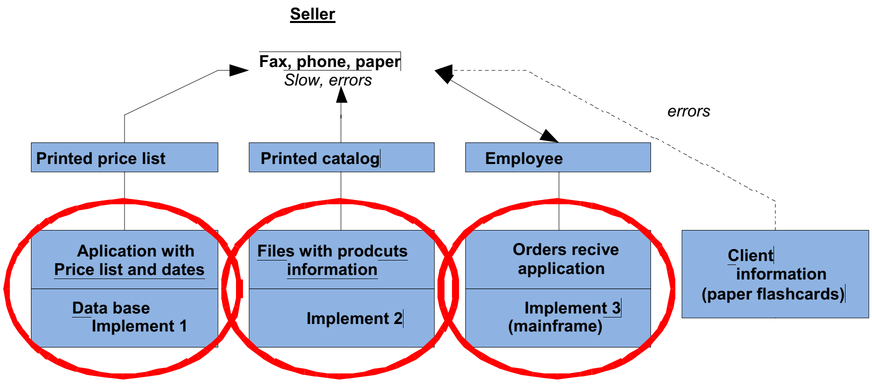


Fig. A.20 List of blocks considered for architecture development

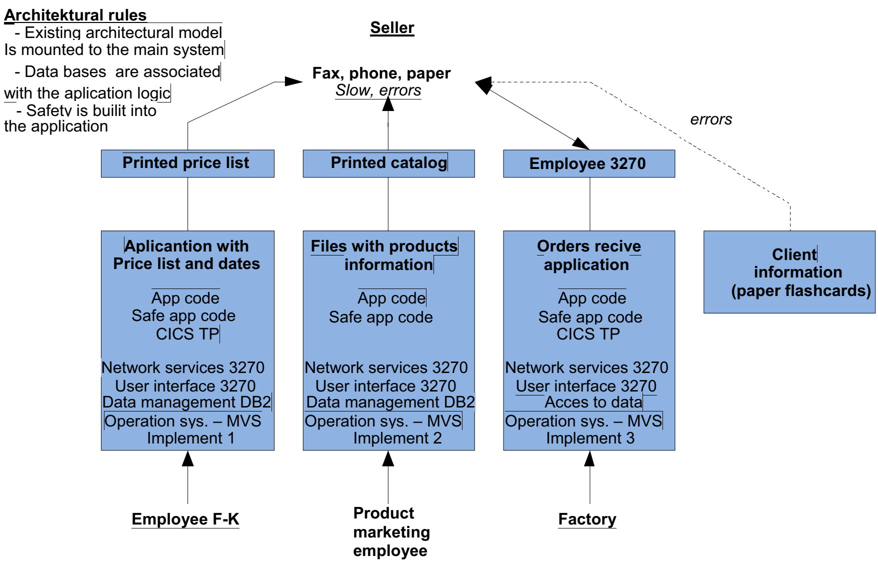


Fig. A.21 Conditions for architecture building in TOGAF

Architectures (Stages B, C and D)

Based on the discussion held between project partners, a list of blocks was prepared that can be used in the development process (Figs. A.21 and A.22). Reviewing the list, architects of the XYZ company suggested manufacturing blocks in the following steps:

1. a description of the initial company level in accordance with TOGAF
2. consideration of different architecture views (after an analysis of the list of blocks)
3. manufacturing a model of block architecture

4. selection of the required portfolio of services for the purposes of building a block model
5. declaration of fulfilment or lack of fulfilment of business objectives and main objectives after analysing the block model and services necessary for block manufacturing
6. defining criteria for the selection of architecture specification performed on the basis of the analysis of the block model and available architectures
7. completion of the definition of the architecture including block models and architectures existing within the Continuum (as part of a supporting process)
8. gap analysis for confirming the correct performance of manufacturing processes.

Architecture development was performed in two steps. In step 1 the existing architecture was evaluated (in three stages) with consideration to:

- using the existing description rules for the existing architecture
- description of the existing architecture in TOGAF conditions (if not performed earlier)
- identification of new requirements for architecture development.

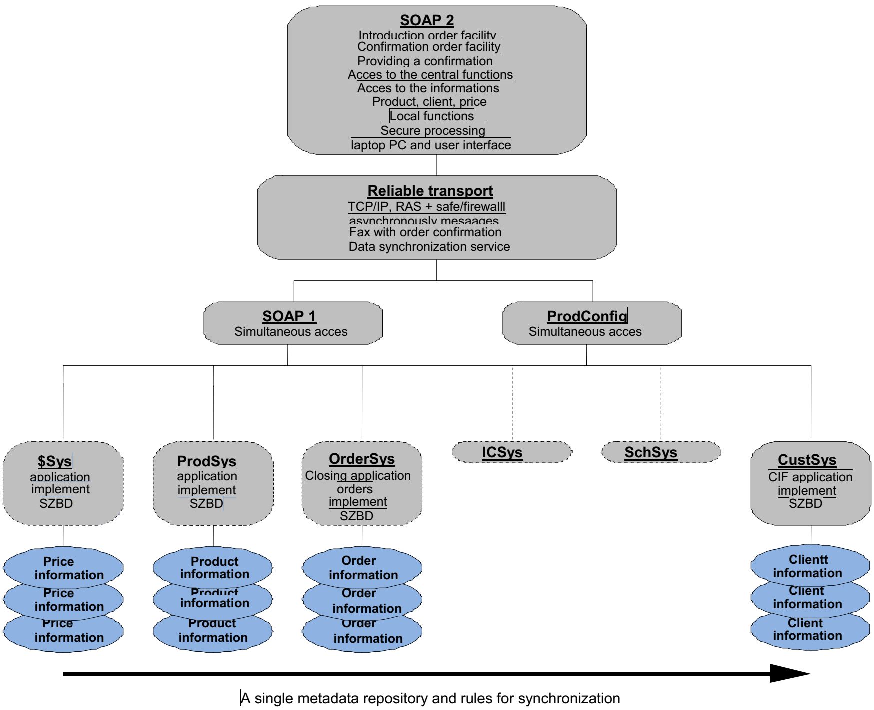


Fig. A.22 Functionalities of the future architecture

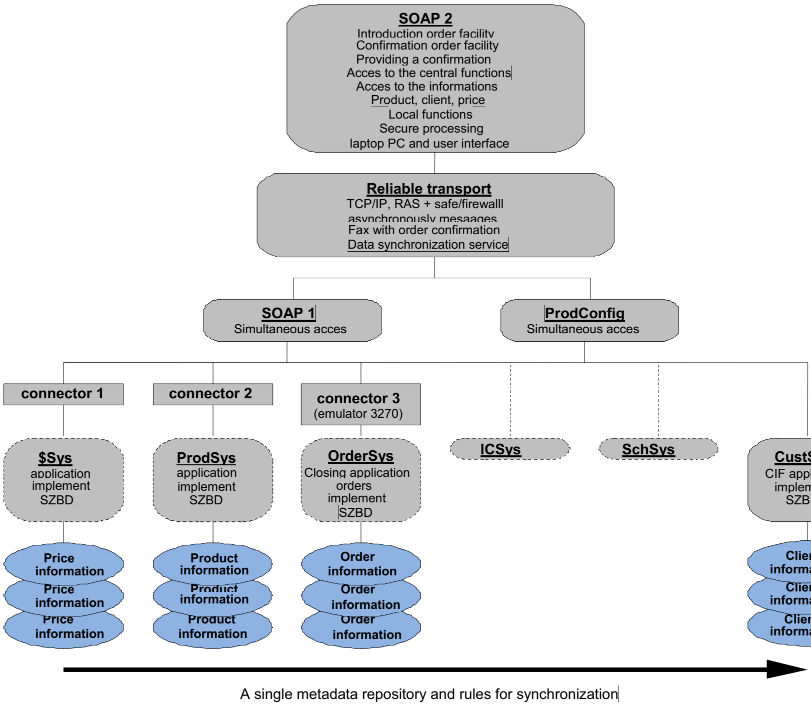


Fig. A.23 Extensions of the functions of the future system architecture

Through the execution of step 1, the implementation of new functions of system architecture in older systems entails a lower performance risk.

In step 2, system functionality is analysed (in different views) in order to establish its purpose (Fig. A.23). After a discussion with partners, a solution was adopted where the views of all proposed functions are reflected in the future architecture. Only their planning processes and their manufacturing control processes were not included in the architecture views (as this would result in an excessively extended scope of the manufactured views).

Execution of step 2 is made possible as a result of:

- identification of added components (*e.g.* functions and blocks)
- specification of resources which will be transferred from the old system
- establishing a rate of return on the investment (considering various options)
- evaluation of the risk of proposed changes in the system architecture
- verification of areas in the system which were implemented with the use of block models and architectures
- adding functions which provide completeness of the system and performing an inspection of their application (gap analysis method)
- update of business and technical requirements

- iterative addition of functionality increases for the purposes of the future architecture.

It must be noted that extending the functionality of the future system architecture should provide for the limitations previously applied in introducing other functionalities as well as the incorporation mechanisms used (Fig. A.23). The initial list of limitations included a description of behaviours of the existing sales system, which showed the improvement in the quality of the process. The existing architecture for the system contained one definition of *metadata* and rules for its synchronisation. As assumed, it was incorporated in the new ordering system (SOAP) by the inclusion of a software module. Below we present a specification of the SOAP application.

SOAP consists of two parts. SOAP1 operates in the company's head office, and SOAP2 is a portable system run by employees responsible for sales. Communication between the two systems is conducted with the use of TCP/IP protocol and synchronised information transfer. In order to improve the operation of the system, the possibility for non-synchronised messaging was suggested, planning the faxing service in the company's head office so that a client could receive a written confirmation of accepting his/her order. Obviously, a data synchronisation service is necessary for rendering sales services. The specification of those services was illustrated from different viewpoints and suitable blocks were prepared. An iterative design was used in the implementation of the sales process in order to check the results obtained with reference to the business requirements presented in the diagrams of usage cases.

Domain-Based Views of Business Processes

Below we present a set of system functionality views referring to the domain of business processes in a company. Such domain-based views are the basis for the implementation of business scenarios. A typical structure of a business scenario is presented in Table A.4.

Business Scenarios

As demonstrated in Section 3.2, business scenarios are a technique which can be used at various stages of building enterprise architecture – primarily during the design of architecture and business architecture visions, and also for acquiring data related to architecture features (if required). The purpose of the scenarios is the identification and understanding of business needs, establishing business requirements and architecture development.

A business scenario includes:

- applications (or sets of applications) which can be implemented in the architecture as well as in the business and technology environment
- partners/team members realising the scenario and software components.

Table A.4 Typical structure of a business scenario – domain-based view

Introduction	Description of the domain, its most important applications and attributes which characterise applications in the domain.
	Short description of activity including important issues related to domains and their inclusion in the target architecture.
Arrangement of applications in domains of business processes	Table of availability of arranged applications.
Assumptions, limitations and instructions	General instructions for developers, implementation specialists and system providers.
Domain structures	Target architecture/domain-based mapping of business processes: Table specifying services and blocks used for the domain. Applied description methods: diagrams showing relationships between main application elements in the domains.
	Description of the quality of services crucial for each business process in the domain and for the development of the target architecture.
	Gained experience of implementation. Migration to the target architecture which refers to business processes in the domain. Guidelines related to the development and responsibility for implementation.
Future directions	Any significant guidelines for systems within the domain.
References	Links to reference materials.

It is important that a scenario should be properly prepared (presenting the processes required from the point of view of the company's economic activity). It can then be used for solving important problems, *e.g.* problems of sales personnel and of the meaning of sales processes. "SMART" scenario can be a good example. Its analysis enables the understanding of what needs to be changed in the branch and what measurable method can be adopted to achieve this by clearly specified measures of success. Therefore, scenario documentation should include:

- a formal description of the main problem and secondary issues
- specification of scenario components, their analysis and solution plans
- presentation of a solution to the problem in conditions of limited cost and time
- identification of requirements and corresponding solutions which provide for the adopted time and cost limitations.

Advantages of Using Business Scenarios

A business scenario contains a complete description of a company's activity, from both the business and architectural point of view, which enables the establishment of individual requirements for developed architectures. Without such a tool, a complete description of the company may prove difficult, since there is a danger that the architecture could be based on an incomplete set of conditions which do not represent the whole description.

The technique of business scenarios requires company management and partners to get involved at an early stage of enterprise architecture development, but it also plays an important role in the acquisition of key employees for the ADM project. It is also important in communication with providers. Most architectures are currently developed by a wide usage of COTS components, which are acquired from many providers in the open market. The selection of COTS for the client's requirements needs business scenarios. They ensure a shared communication medium, where the seller can associate the client's problems with technical solutions and present them to the provider. Apart from much other proof of the usefulness of such scenarios, they allow optimum solutions to be obtained from providers with the use of open standards (*e.g.* offering COTS components built in accordance with those standards).

Creating a business scenario (Fig. A.24), as an identification process, includes:

1. marking documents and selecting tasks while preparing the scenario
2. classification of the company and selecting a scenario for the technical environment and its descriptions
3. identification and description of desired objectives (based on selected tasks)
4. indicating actors and their place in the process of business model manufacturing
5. identification of actors (tools and IT methods) and their place in the technology model
6. identification and presentation of tasks and duties for the purposes of the actors
7. verification of the degree of fulfilling objectives and taking relevant action in this respect.

A company's business scenario is usually manufactured iteratively at the stages of acquiring, analysing and processing the operating conditions of the company, which begin (as in Fig. A.24) with an analysis of a scenario problem and end with an improvement of objectives set for the scenario. In each iteration, an evaluation of scenario usefulness is conducted and changes are introduced in its structure. Fig. A.25 shows the stages of such an evaluation, which include: collection of data for the scenario, their analysis and a review of their usefulness.

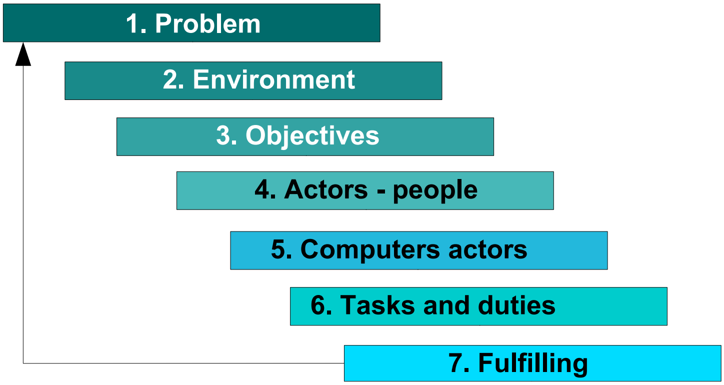


Fig. A.24 Business scenario manufacturing process

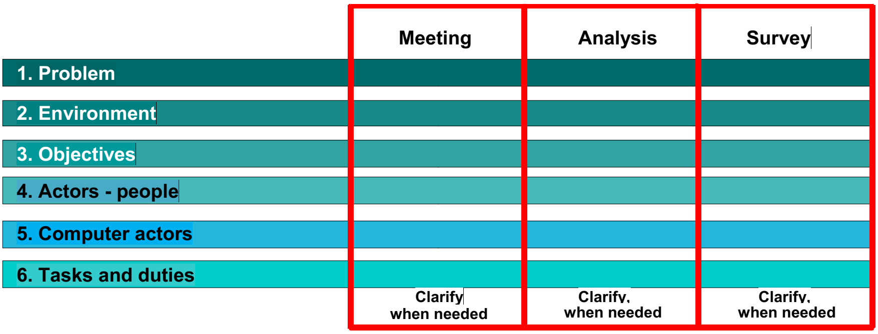


Fig. A.25 Stages of business scenario development

In Fig. A.25 the process of refinement (clarification) of business scenarios is not presented due to its iterative nature. The name is derived from the word *refine*, used in the TOGAF description to specify activities of improving development processes. While developing enterprise architectures, workshop scenarios can be used in addition to business scenarios. The information included in those can be used in the initial development of company activity (before the preparation of a business scenario).

A very good way of collecting information is to run workshops on scenario building, where a scenario consultant trains chosen groups and representatives of the business. During the workshops, questions are asked in order to obtain the information required to solve a problem. The participants have to be carefully selected (they can be representatives of senior management or of technical domains in the organisation). If a design for the activity scenario already exists, *e.g.* as a result of *pre-processing*, the collected information is processed (as described above). Workshops can also be run in order to conduct a review of the progress level of scenario designs. In the case of complex scenarios, it is a good idea to run a series of workshops. When collecting information, an architect can enhance the legibility of the activity scenario by using case studies, to which a participant of the workshops can easily refer. When using case studies of a company's activity, it is important that the anonymity of the parties involved is maintained. This provides reliability of the process of collecting full knowledge related to company operation and the maturity of its processes.

The collected information is processed and documented. On this basis models are created. The expertise of consultants is used here as well as the latest results and experience related to modelling. The models prepared and the documentation do not always directly reflect the knowledge acquired from the investigation, as it is subject to the processes of filtering, translating and adapting to actual business needs.

The discussed scenarios (workshop and business) are worth using in the analysis of those needs to create links between requirements and company activity. One of the techniques which is helpful in maintaining such links consists of creating matrices which serve the evaluation of business processes and refer to each area, the selection of team members, and the assignment of main resources (*e.g.* hardware and data). This technique is aimed at the fulfilment of a company's objectives and solving other related problems.

Business Scenario Review

Business scenario reviews are performed by project sponsors after the scenario descriptions are completed. They do it in order to make sure that there is full understanding of the problem. It is recommended that meetings and workshops are held with sponsors and partners to evaluate the company's scenarios. The aim of these meetings is the control of the project assumptions and partners' expectations by the person responsible for the preparation of architectures. It is also advised that other ideas be used, such as including a periodic examination of the degree of understanding of project needs in the agenda of Management meetings, and so drawing the interest of Board members to the project.

Table A.5 Typical list of contents of a business scenario

list of contents
foreword
summary
document – activity roadmap
activity overview
context for the scenario
purpose of the scenario
definitions / description of terminology used
processes and opinions of professionals about the processes
business setting
description of processes
actors and their roles and responsibilities
actors and roles
relationships of processes and their components
relationships of people and processes
information analysis
business scenario preparation principles
IT rules
limitations for building a business scenario
requirements related to business scenarios
problem summary
aims of building a business scenario
appendix a: business scenarios – additional information
appendix b: workshops related to additional activity

A business scenario review is presented as project documentation (including reviews and the remaining data). In the course of preparing the documents, all critical stages of the project need to be included as well as the results of the entities developing the project. Information on the entities themselves is also important. Such declarations should particularly refer to the duties and responsibilities of the entities in order to properly define the key system functions and requirements related to the project and the use of services for the purposes of those functions.

A scenario includes two main types of content: the presentation of models and their textual description:

- models of business scenarios and technology are used in the acquisition of views in graphic form in order to better understand a company's operation (in other words, they describe the entities and their interactions and constitute a starting point for the confirmation of detailed requirements)
- business scenario descriptions pinpoint architecture details and justify them.

Table A.5 presents a typical list of contents of a business scenario.

Financial Support for the Manufacturing of Business Scenarios

The manufacturing of a business scenario is not only the domain of the architect focused on its construction. Scenario manufacturing is a manufacturing process implemented as part of an IT project. Therefore, the management and project partners involved in the scenario manufacturing should together decide in what time and with what resources the objectives set for them will be fulfilled. Relations with a business partner and their engagement are obviously of high significance for project development. Involvement of a business partner in the development is usually the greatest in the initial stages. The specification of requirements (with the participation of the architect) is done in further stages of the manufacturing, where architectures have already been provisionally described (and the partner is less engaged). Sales personnel (clients of the system for the XYZ company) are engaged in the preparation of business scenarios at early stages of the manufacturing cycle. They also take part in the later stages, when technology architectures have already been developed. Architecture providers, on the other hand, suggest their solutions (for building and integration) at later stages of the manufacturing. A distribution of the costs of architecture development between the providers and clients is illustrated with the example of building business scenarios (Fig. A.26). The costs of the collaboration (*e.g.* on scenario verification) are most probably near the middle of the given scope for the cost of business scenarios.

Business Scenarios for ADM Purposes

In accordance with TOGAF, business scenarios are most useful in the initial stage of the ADM. They are used for establishing business requirements and for

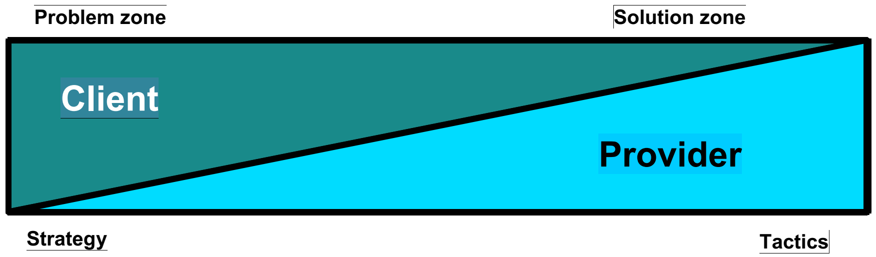


Fig. A.26 Relative costs of building business scenarios

building a relationship between the software provider and the client company representatives for the duration of the project. Since the requirements are important at all stages of the development, the applied technique of business scenario manufacturing plays an important role, which is providing complete and correct requirements by the company. On the other hand, partners (managers, end users) together with the architect should agree on the presentation of company activity (this usually happens in the course of acquiring the most important entities for the project). In the successive stages of the ADM cycle, the requirements are incorporated in the appropriate scenarios. If problems occur with obtaining the requirements, the following items need to be specified:

- time for obtaining the requirements and the method of entering them in the everyday work of company employees involved in the project
- information transfer structure for building business scenarios (the correct definition of the structure ensures the multiple use of information collected by each of the partners taking part in the project)
- the most important rules of working with domain experts on knowledge acquisition.

The described approach to scenario building forms the basis for the manufacturing of a chain of links important in obtaining business requirements for business scenarios and architectures.

A.2 Summary

In this Appendix we presented knowledge of primary (three-dimensional) project negentropy together with the relevant specification. The knowledge refers to the construction and use of the Continuum. It is good to make sure that in the case of extending the model and making changes in the specification of its variables, the model structure (Appendix 1) of the project repository is useful for business modelling. It also refers to possible changes in the business application.

At the beginning, the implementation of business applications was focused on the basic areas – from the point of view of data processing – of company operation: finance/accounting, human resources, procurement, shipping, *etc*. The role of the repository was minor. Systems were manufactured with the use of any – often incoherent – technologies (which now limits the possibility for their cooperation). Therefore, an individual business process: "ordering, stock-taking, entering in the books-sales, using client information," which could take a short time, takes even several minutes (due to the lack of integration of the company's operation areas). In this situation, how should a company act? A convenient solution to the problem is to build a repository. Although the company may decide to incur (significant) expenses for new equipment, additional software and personnel training, it is always a good idea to consider the possibility of integrating the existing data in the enterprise architecture. Patterns for this strategy are available in the repository. Implementing a uniform IT system based on integrated data is only possible if the company evolves from the strategy of considering resources as separate applications to the strategy of manufacturing a uniform business process based on customer service. This provides the company with the possibility of evolution with consideration to changes in the needs notified by the client and a chance to register those changes in the repository.

When considering the use of the repository to evaluate project negentropy, architectures can also be shown – apart from the system architectures manufactured on the basis of system models (MDA) – whose design is based on the paradigms of source code programming. An example of such an approach is using the PASCAL language in the process of architecture development, where a code is designed and the possibility for its re-use is taken into account (exclusively for the established domain). This approach is considered immature and aimed at developers, rather than the authors of the system vision. For it means creating narrow, domain-based systems instead of enterprise architectures and systems of which the development is based on the use of good manufacturing and management practices. An example of good manufacturing practices is using sets of prevalent programming styles, and an example of good management practices is the permanent availability of the client in the system manufacturing process.

Today's architecture development is based on models rather than codes. Such models and their components can obviously be re-used if they are made available in a repository (the Continuum).

In these conditions, the concept of re-using the code available in the project repository can also prove helpful. The decision about its re-use obviously comes down to the usual question: how should the code be properly used? Is it right to divide the code in accordance with system layers and create classic layered components? Or perhaps a better solution would be to manufacture each code component at the application level and treat it as a software model? In each case of component manufacturing it should be remembered, however, that the processes need to be compliant with standards, which will ensure their easier incorporation in the structures of the existing software. We will then have the possibility to present the concept of manufacturing an application which is independent of the hardware platform and the software.

Due to the above, this Appendix has been added to the work to present the possibility for reducing the (scalar) project negentropy with the use of the repository, when the implementation (or modification) of systems is based on the use of archived software and hardware components. Their specification is included in the Continuum description presented in Section 3.4.1.

B

Appendix Client Organisation Maturity

In this Appendix (as was the case with project negentropy) we will describe the maturity of the client organisation based on the COBIT standard version 4.1, which is also complementary to studies on the evaluation of client organisation maturity (discussed in Section 4.5). The description will be supported by an example of the application of this standard to evaluate an IT organisation which uses the CRM system (supporting relationships with its clients). This example is the result of the authors' own work on the applicability of information technology in companies [1, 3, 9, 19]. Because the sample questions refer to the control objectives of COBIT, before presenting the results we will make an introduction to this standard.

The production processes (manufacturing and management) are grouped into four main domains: planning and organisation, acquisition and implementation, delivery and monitoring.

The PO Domain (*Planning and Organisation*) includes the planning and evaluation of the strategy and tactics of IT management and enables the establishment of the application method of IT for business objectives of the company. The realisation of these objectives involves the company's preparation of a plan and development strategy, and the construction of the appropriate IT infrastructure (to support the PO domain).

The AI Domain (*Acquisition and Implementation*) relates to the identification of technological solutions (owned or acquired) and their integration with the company's business processes.

The DS Domain (*Delivery*) includes the planning / evaluation processes of services supporting technological solutions, which include: service level management, information security, data and IT infrastructure.

The ME Domain (*Monitoring*) is related to the periodic evaluation of the company's processes and information technologies which support these processes.

Attention is drawn to the performance management of existing IT systems, the monitoring of the internal control system and their compliance with the applicable laws.

This division also allows the planning of the implementation of information technologies for the company's purposes, as is the case with the ADM for TOGAF. It also allows the simultaneous evaluation of the company and the applied information technologies. It should be noted that this evaluation depends on the precise definition of the organisation's business objectives and the IT environment. This evaluation will also depend on the information technologies themselves, their degree of complexity and the value of information held by the organisation.

The maturity evaluation of IT processes proposed in COBIT is based on the scoring method with the scale of 0 ÷ 5, transferred from other methods of organisation maturity evaluation used by the Software Engineering Institute. Determining the level of maturity of each of the 34 processes suggested by the COBIT standard enables the evaluation of:

- the current degree of organisation maturity
- the current degree of the best organisations within the industry
- the current degree of international standards of organisation evaluation
- the strategic direction of organisation development.

6 degrees of maturity are distinguished within COBIT (Fig. B.1).

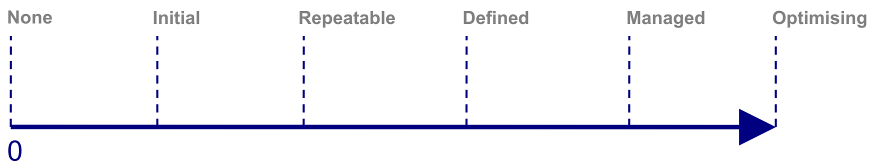


Fig. B.1 Organisation maturity degrees according to COBIT

Level "0" – marks the beginning of the development of the organisation, in which processes are chaotic, unorganised and are not identifiable. Lack of process management.

Level "1" – initial, in which processes occur ad hoc. At this level there is awareness of the need for using mature processes. There are no mature manufacturing and management processes in the organisation.

Level "2" – repeatable, in which processes exhibit a certain periodicity, and repeatable component actions of these processes can be clearly observed. They are not formally defined and there are no mechanisms to enforce their periodicity. Still, they are developed enough so that within them similar products are made by people undertaking the same tasks. However, there are no formal training sessions or applying of standard procedures for these processes. The responsibility for their implementation is distributed between individual employees. There is a clear correlation between the use of these processes and the individual knowledge of the employees. This leads to the occurrence of errors.

Level "3" – defined, in which the processes in the organisation are defined by formal documentation, which enables their uniform application throughout the organisation. At this level, it is assumed that the processes are standardised and documented. Their application was explained to employees and supported by appropriate training. The manufacturing of systems in accordance with the defined processes depends on the knowledge of the manufacturing team members. This condition significantly impedes the detection of deviations and errors.

Level "4" – managed, where processes are monitored and controlled. It is possible to introduce corrective actions in the use of ineffective processes and to ensure continuous improvement of processes and best practices.

Level "5" – optimising, in which best practices are applied. As a result of the continuous improvement of processes and comparison to other organisations, processes have been promoted to the rank of best practices. Information technologies are applied (integrating manufacturing and management processes) which automate workflow, improve their quality and performance, and enable organisations to quickly adapt to market conditions.

B.1 Specification of Client Maturity in the IPP Model

Following the introduction of the standard, we are going to present a description, complementary to this work, of a client organisation evaluation using the IPP model. This description contains domain specifications in the form of control questions. The description of the study concerns the planning and organisation domain. Figures B.2 - B.3 present exemplary questions relating to the general objective and the specific objectives for the other domains defined in COBIT.

Below we present the remaining results of the COBIT analysis (from the questionnaires of Figures B.2-B.3 and others not presented it this book), divided into three main domains and 34 processes within them. The obtained description is consistent with the discussed IPP model.

The AI Domain: Acquisition and Implementation

AI1 . Identification of the Defined and Repeatable Processes

Main Objective. The control of the IT process of the identification of the defined and repeatable processes, for efficient and effective software development which meets user requirements.

						Ocena
	ACQUISTION AND IMPLEMENTATION					
1	**IDENTIFICATION OF AUTOMATIC SOLUTIONS**					
AI1.1	Have been identified, desiganted priorites and specifed funcional and technical requirments, necessary to achive expected results of IT investment program?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	3
AI1.2	Is possible to determinate the expected risk of the implementation of the project?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	1
AI1.3	Is the project feasible and are there possibile alternative measures?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	3
AI1.4	Is required consent of buisness sponsor for aproval of the project , are there any technical limitation in decision making?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
2	**ACQUISTION AND MAINTENANCE OF APPLICATIONS**					
AI2.1	Does the board is accept overall desgin of the system?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
	Are the buisness requirments translating into specification of high-level project for purchased application?	◉ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	0
AI2.2	Are criteria defined for acceptance of the system requirments?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
	Are prepared detailed and technical project requirments of application are compatible with high level design?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	3
AI2.3	Are specific measures to control application?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
	Is authenticated buisness control implement of application?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	0
AI2.4	Are there any specific requirments for application security?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	3
	Does the application is exposed to danger and excessive availability against defined risk?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	0
AI2.5	Is it known, what objectives are achived by the implementation and application configuration?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	3
	Are the buisness objectives achived through the implementation of previously purchased software?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	1
AI2.6	Are updatings required to an existing application?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	3
	Czy są wykorzystywane komponenty rozwoju nowego systemu w przypadku głównych zmian istniejącego systemu, które skutkują istotnymi zmianami w bieżących projektach?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2

Fig. B.2 COBIT application for evaluating the organisation in the AI domain (part VI)

AI2.7	Is development of application ensured in accordance with all the requirments?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
	Is software development by adding functionality carried out in accordance with the standards and project specifications?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
AI2.8	Is application undergone to quality control of software?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	0
	Are refined resources of created plan determinated in accordance with requirmnts definition?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	1
AI2.9	Are the application requirments managed in an appropiate way(ie.accrodance with the guidelines to each department)?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
	Whether in the change management process is tracked state of the all requirments and also if the changes are approved?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
AI2.10	Is the application maintenance carried out?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
	Are the startegy and plan application storage improving?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	0
3	**TECHNOLOGY INFRASTRUCTURE ACQUISITION PLAN**					
AI3.1	Is there any plan to achive technology infrastucture?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	1
	Is there any plan of aquire, implement and maintain technical infrastructure conformed the requirments?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	1
AI3.2	Was it specifed what should be done to achive control and protection of resources and assets of infrastructure?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	1
	Is the internal audit carried out of infrastructural resources?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
AI3.3	Is it possible to define strategic actions, what has to be takento maintenance of infrastucture?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
	Is any developed strategy and plan of infrastructure maintenance, where changes are controled in conjuction with the procedure of changes management in the organization?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	1
AI3.4	Will be funcional test carried out of the environment?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	2
	Is creation and development of environmental test supports effective testing and ensure ingerity of the infrastructure components?	☐ Nie spełnia	☐ Niewystarczająco	☐ Dostatecznie	☐ W całości spełnia	1

Fig. B.3 COBIT application for evaluating the organisation in the AI domain (VII)

Level: Optimising

Evaluation of the Capability of Achieving the Assumed Objective. This is one of the five implemented AI processes (ITIL) of the organisation which has achieved the highest rating (Fig. B.4). It means that the organisation – using management methods based of its own developed best practices (although without creating special documentation) – can realise projects of varying scales, including large-company applications and specific tactical projects.

Table B.1 AI1 process analysis: identification of the defined and repeatable processes

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	AI1.1	Process priorities have been identified and assigned in the organisation. These priorities provide the possibility to highlight and repeat the specification process of functional and technical requirements. Their goal is to implement the IT investment programme. By identifying and defining the repeatable processes the expected risk of project implementation can be determined. In addition, the project risk is identified and alternative scenarios are considered.
	AI1.2	
	AI1.3	
	AI1.4	

The evaluation of the implementation degree of the defined and repeatable processes indicates that the organisation is able to recognise the applicability of IT and evaluate its impact on the changes of business processes to improve their overall performance. Details of the evaluation are provided in Table B.1 and Figure B.4.

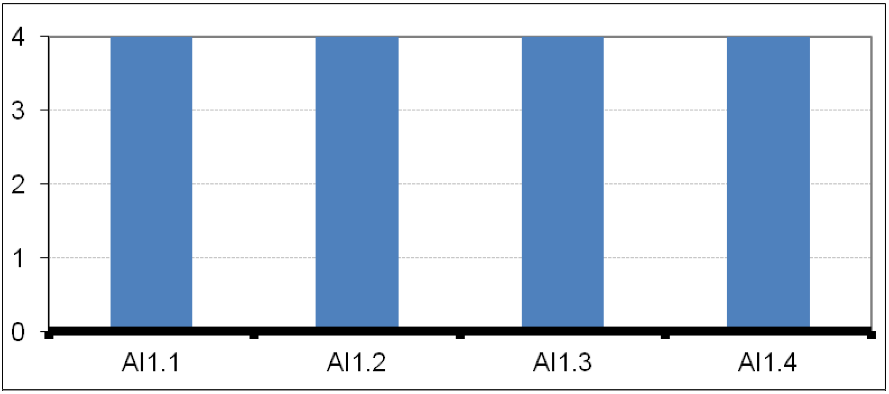


Fig. B.4 Level of meeting the control objectives of the AI1 process: identification of the defined and repeatable processes

AI2. Acquiring and Maintaining Applications

Main Objective. The control of the IT process of acquiring and maintaining an application in line with the business objective of the organisation.

Level: Managed

Table B.2 AI2 process analysis: acquiring and maintaining an application

<table>
<tr><th>PROCESS EVALUATION</th><th>PROCESS_ID</th><th>CONCLUSIONS ON PROCESS EVALUATION</th></tr>
<tr><td rowspan="2">Processes which do not require improvement</td><td>AI2.1</td><td rowspan="2">Prior to the implementation of a new system in the surveyed organisation, the Board must approve the project. And prior to the acquisition of the system, the organisation’s business requirements should be converted into project specifications. This will facilitate the acquisition of the application. The acceptance criteria of the requirements towards the system will be defined. Detailed project and technological assumptions for the application will also be prepared. Then the acquisition and implementation of the application should be compliant with the project.</td></tr>
<tr><td>AI2.2</td></tr>
<tr><td rowspan="8"></td><td>AI2.3</td><td rowspan="8">The application security requirements are clearly defined. Objectives to be achieved during the implementation and configuration of the application are also defined. Through the implementation of the previously acquired software, the business objectives of the organisation are achieved.
In the case of the construction of the system (and significant changes in the existing system) components are used.
For the AI2.6 process (system upgrade) there is no need to update the CRM system used by the organisation.
The development of the organisation’s application complies with its requirements. The addition of new features is carried out according to standards and the project specification.
The AI2.8 process (resource improvement) is carried out in accordance with the definition of requirements.
The AI2.9 process of obtaining requirements for the application is managed in accordance with the guidelines of the various departments of the organisation.
The AI2.10 process (application maintenance) is realised.</td></tr>
<tr><td>AI2.4</td></tr>
<tr><td>AI2.5</td></tr>
<tr><td>AI2.6</td></tr>
<tr><td>AI2.7</td></tr>
<tr><td>AI2.8</td></tr>
<tr><td>AI2.9</td></tr>
<tr><td>AI2.1</td></tr>
<tr><td>Processes which require slight improvement</td><td>AI2.3</td><td>In the surveyed organisation, in the AI2.3 process (authorisation of the control of business processes) authorisation is carried out at the level of the contract with the provider</td></tr>
<tr><td rowspan="2"></td><td>AI2.8</td><td rowspan="2">For the AI2.8 process (application control) applications are subjected to quality control in an unorganised manner, i.e. not every application is subject to the inspection.
For the AI2.10 process (strategy and plan for storing applications) a strategy or plan for storing applications does not exist in the firm, however, there is a simple procedure regulating these activities.</td></tr>
<tr><td>AI2.10</td></tr>
<tr><td rowspan="2">Processes which require significant improvement</td><td>AI2.3</td><td rowspan="2">In the surveyed organisation, in the AI2.3 process, application control measures are not strictly defined, but are used.
In the CRM system used in the organisation, the change management process does not exist (AI2.9), although changes are made. This lack makes it impossible to track the level of requirements and the system level</td></tr>
<tr><td>AI2.9</td></tr>
<tr><td>Processes which do not function in the organisation</td><td>AI2.4</td><td>In the studied organisation there is no defined process evaluating the risk connected with the operation of the application (AI2.4). This lack disenables any preventive action to be taken</td></tr>
</table>

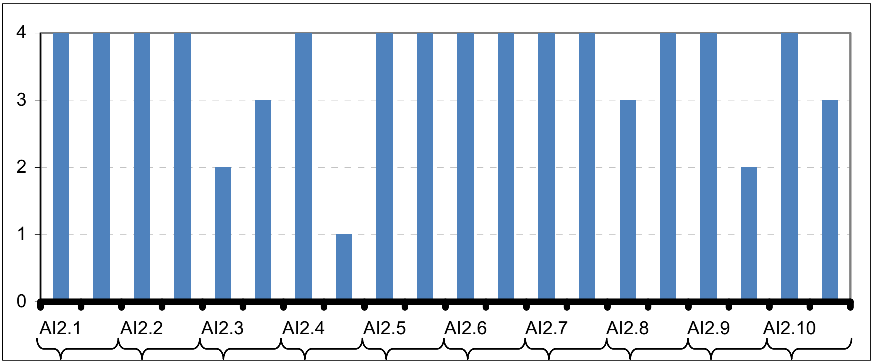


Fig. B.5 Meeting control objectives of the AI2: acquiring and maintaining the application.

Evaluation of the Possibility of Achieving the Assumed Objective. After an analysis of specific processes, it can be stated that there is a method of acquisition, deployment and maintenance of applications. It includes, among others, the formal development process and parameterisation of the application or the criteria for its acquisition, and the processes of testing, documenting and maintaining the application. This method streamlines the process of acquiring and maintaining most of the applications in an organisation. However, it requires some changes in the control of the application level; there is a risk of deploying applications and business processes (current control of an application is random; it applies to resources and change management processes). The details of the proposed comprehensive solution are provided in Table B.2 and Fig. B.5.

AI3. Plans of Manufacturing and Maintaining the Technological Infrastructure

Main Objective. The control of the IT process of the manufacturing and maintenance of the technological infrastructure in line with the business objective – providing platforms to support business applications.

Level: Defined

Evaluation of the Possibility of Achieving the Assumed Objective. The analysis of the organisation level shows clear and understandable IT infrastructure administration processes included in it. Their aim is to support the organisation in the functioning of critical applications. However, the repeatability of the processes can not be determined, thus it can not be univocally determined if the manufactured and maintained infrastructure supports the needs of business applications. The details of the evaluation are described in Fig. B.6 and in Table B.3.

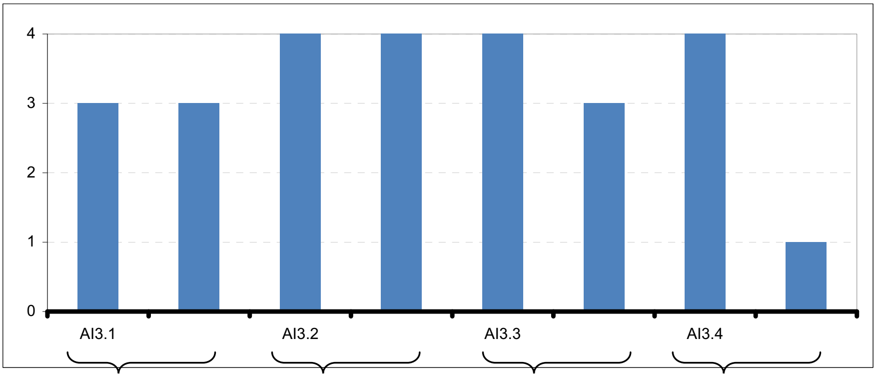


Fig. B.6 Meeting AI3 goals: manufacturing and maintaining the technological infrastructure

Table B.3 AI3 process analysis: manufacturing and maintaining the technological infrastructure

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	AI3.2	It is clearly defined in the organisation what needs to be done to gain control over the resources and means of the IT infrastructure. Random inspection of the infrastructure resources.
	AI3.3	For the AI3.3 process (strategic tasks) it is possible in the firm to establish strategic tasks in order to maintain the infrastructure.
	AI3.4	During the study of the AI3.4 process in the organisation (before delivery) tests of the environmental impact are carried out.
Processes which require slight improvement	AI3.1	In the surveyed organisation, a plan for manufacturing a technological infrastructure does not exist. As it is developed, the needs for its operation are determined.
	AI3.3	For the AI3.3 process (strategy construction and development plan), in the studied organisation, a strategy and plan for infrastructure maintenance is partially developed. Control over the changes in the plan along with the use of change management procedures are taken into account.
Processes which do not function in the organisation	AI3.4	For the AI3.4 process (development of the testing environment), in the organisation, a diagnostic environment is created and developed. Yet, the testing processes are not well supported, thus the integrity of the infrastructure components is not perceived.

AI4. Proper Use of the Application

Main Objective. The control of the IT process of ensuring the use of the application in line with the business objective of the organisation and ensuring end-user satisfaction with the quality of the application. It is important to provide the option of installing other technological solutions.

Level: Managed

Evaluation of the Possibility of Achieving the Assumed Objective. This process is well organised, *i.e.* users are given adequate knowledge of the systems and the used applications. There are no formally defined procedures for the use of the application. The users act in accordance with the practices adopted in the organisation or acquire support from instructions provided with the software. The procedure adopted in the organisation refers to all systems and organisational units which facilitates the analysis of the realised application from a business perspective. Users gather feedback about the application. If problems arise, corrective action is taken. The details of the evaluation are given in Fig. B.7 and Table B.4.

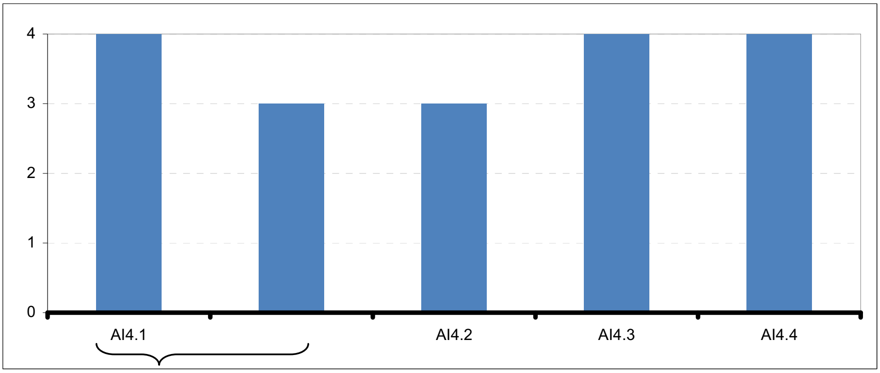


Fig. B.7 Meeting control objectives of the AI4 process: ensuring the use of the application

Table B.4 AI4 process analysis: ensuring the use of the application

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	AI4.1	For the AI4.1 process (preparation of the operation plan) in the studied organisation, responsibility aspects connected with the use of the application have been included.
	AI4.3	Adequate knowledge is delivered to end users of the system for effective and efficient use of the application.
	AI4.4	The technical department and the support organisations receive the necessary knowledge for the effective delivery, support and maintenance of the system and the structure related to it.
Processes which require slight improvement	AI4.1	In the surveyed company, in the AI4.1 process, in the identification plan and the application documentation (technical, operational, functional), the responsibility of the project manager is assumed, covering all aspects of the business and the organisation within the realised project. The Board of the studied organisation receives complete knowledge which enables applications to be managed, even during a break-down.
	AI4.2	This knowledge is essential to the Management Board. With it, full responsibility for the ongoing project can be assumed, internal monitoring can be conducted and the organisation's applications can be administered.

AI5. Acquisition of IT Resources

Main Objective. The control of the IT process of the acquisition of IT resources in line with the business objective - to effectively develop IT investments and business activities.

Level: Optimising

Evaluation of the Possibility of Achieving the Assumed Objective. This process is one of the five processes (Fig. B.5) which reached the highest level of maturity in the studied organisation. Thus, the acquisition of IT resources is in line with the policy of the organisation and good relationships with providers and partners are established. The quality of these relationships is measured and monitored. The details of the evaluation are given in Fig. B.8 and Table B.5.

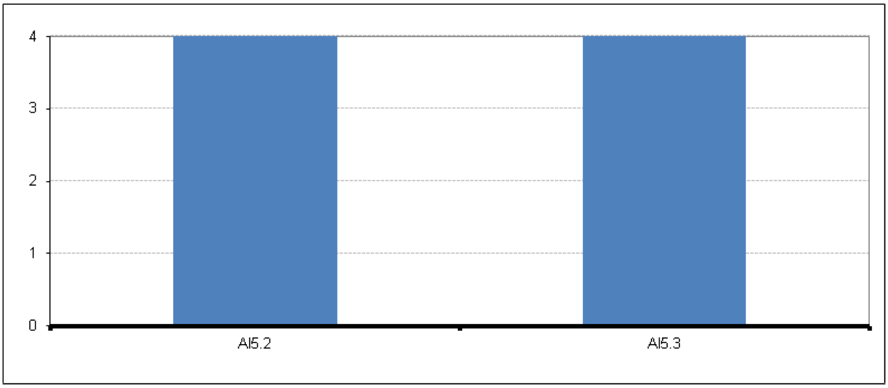


Fig. B.8 Meeting control objectives of the AI5 process: acquisition of IT resources

Table B.5 AI5 process analysis: acquisition of IT resources

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	AI5.2	In the studied organisation, the procedures of manufacturing, modifying and terminating contracts with providers are established. All contracts and changes made in them are analysed by the lawyers of the organisation.
	AI5.3	

AI6. Change Management

Main Objective. The control of the IT process of change management in line with the business objective, which involves minimising the likelihood of interference in the operation of the system and making unauthorised changes.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. This process requires significant improvement, although there is an informal process of change management. Some changes are implemented according to this process. For some applications, however, there are more formal change management solutions. The company should develop in this direction. The lack of a complete formal change management system can contribute to errors and reduce the performance of the applications. The documentation of the application configuration must be kept up to date. Otherwise, the functions of the system should be rebuilt and the manufacturing process should be repeated. The details of the evaluation are given in Fig. B.9 and in Table B.6.

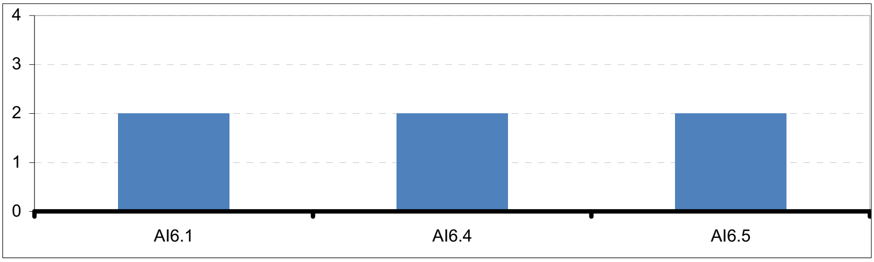


Fig. B.9 Meeting control objectives of the AI6 process: change management

Table B.6 AI6 process analysis: change management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which require significant improvement	AI6.1	In the studied organisation, there are no established change management procedures for the CRM system. If changes occur, all the processes of defining, testing, documenting, evaluating and approving these changes are subject to the project manager.
	AI6.4	In the studied CRM system, there is no change tracking or reporting system, although such reports exist in the studied company.
	AI6.5	After making changes to the system, procedures or change documentation are not updated. This is due to the fact that there are no procedures and documentation of the CRM system in the organisation. In the case of the analysed system, it is not required, as all instructions are included in the universal OpenSource documentation available on the website of the system.

AI7. Implementation and Accreditation of Changes

Main Objective. The control of the IT process of implementation and accreditation of changes in line with the business objective – construction of applications which operate without interruption after changes.

Level: Defined

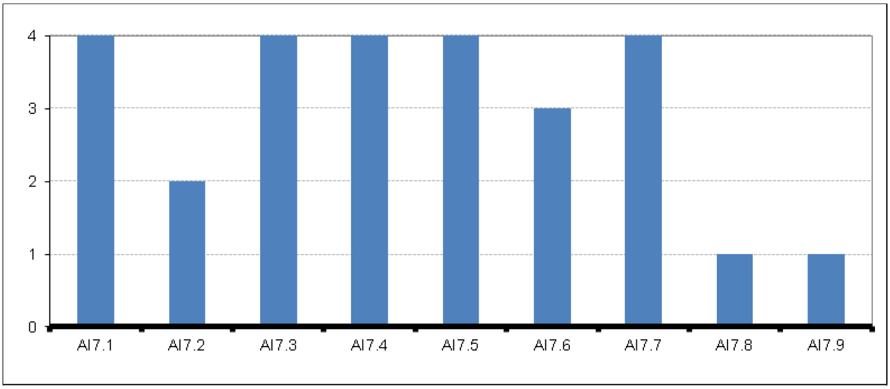


Fig. B.10 Meeting control objectives of AI7: implementation and accreditation of changes

Table B.7 AI7 process analysis: implementation and accreditation of changes

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	AI7.1	When, in a given department of the test organisation, an IT system is implemented, modified or developed, the staff (system users) are subject to brief training. Their goal is to provide knowledge on the applications or the need to implement new system functionalities. A system implementation plan is always set, and it is subject to the approval of the relevant team members. The organisation has a defined and secure testing environment for the deployed applications. The exchange of data and its migration within the infrastructure is planned as part of the development of the organisation. Results obtained during tests of the application are analysed in order to identify its errors.
	AI7.3	
	AI7.4	
	AI7.5	
	AI7.7	
Processes which require improvement	AI7.6	The organisation has no plan to test the application (after changes are made). Such tests are performed only when necessary.
Processes which require significant improvement	AI7.2	In the studied organisation, there is no testing plan, and in result the roles and responsibilities of team members are not defined. The input and output client criteria of the application are not defined. Application acceptance tests are partly used.
Processes which do not function in the organisation	AI7.8	The AI7 did not apply to the organisation. The CRM system used in this organisation was not subject to change, which resulted in the company's lack of conviction that change is necessary.
	AI7.9	The company does not evaluate the application through its implementation. The level of implementation cannot be specified, and corrective action cannot be designed. Implementation errors (modifications) are registered during use or during testing by users.

Evaluation of the Possibility of Achieving the Assumed Objective. The AI7 process analysis (Table B.7) identifies the method of application implementation (after changes are made in the organisation) and data migration. Then the change accreditation process in the application is integrated with the system life cycle and can be (partially) automated. The application diagnostics are realised according to an appropriate plan of tests. The details of the evaluation are described in Figure B.10 and in Table B.7.

DS Domain: Delivery

DS1. Defining the Level and Service Management

Main Objective. The control of the IT process of defining the level and service management in line with the business objective – the implementation of a uniform level of service in the organisation.

Level: Initial

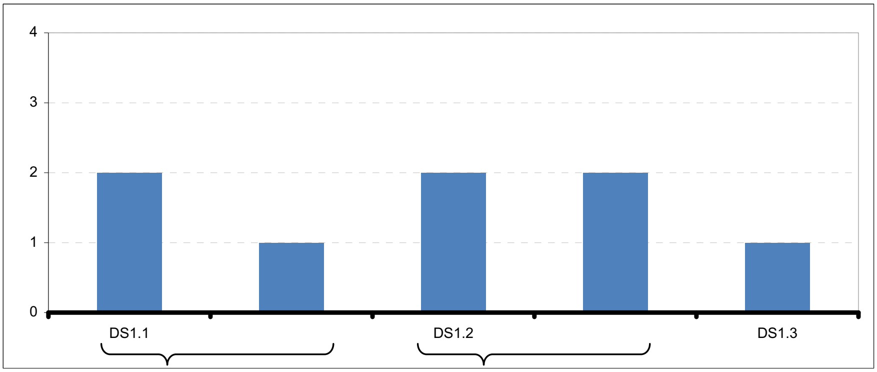


Fig. B.11 Meeting control objectives of the DS1 process: defining the level and service management

Table B.8 DS1 process analysis: defining the level and service management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which require significant improvement	DS1.1	In the studied organisation, the DS1.1 process (service management) is contained in a framework which structures this process. This framework does not guarantee the formalisation of the service management process and is not sufficient for the needs of the organisation.
	DS1.2	The DS1.2 process within service monitoring and reporting: Services are constantly monitored but their reporting occurs without any regularity.
Processes which do not function in the organisation	DS1.1	In the studied organisation, the DS1.1 process (service management structure) is informal. As was mentioned previously, this framework does not adequately reflect the nature of the organisation's management structure.
	DS1.3	The process does not apply to the studied organisation. In the studied organisation, there is no systematic way to monitor service providers. It can not be stated if they are providers of efficient services and if the services conform to the requirements of the clients.

Evaluation of the Possibility of Achieving the Assumed Objective. This process in the studied organisation needs considerable improvement, since most of the specific objectives are not met. At this level of the organisation, it is concluded that there is a need to implement the state of services management process (in order to achieve the objective). The service performance measurement process is not credible, because it is also conducted in an informal manner, with vaguely defined objectives. Reports on service performance are prepared very rarely. The details of the evaluation are given in Fig. B.11 and Table B.8.

DS2. Service Management of External Partners

Main Objective. The control of the IT process of service management of external partners in line with the business objective – allowing for clear definition of the roles and responsibilities of external partners in relation to the IT systems of the organisation.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. The studied organisation uses the services of external software providers for realising its needs to a very small degree (budget system). The monitoring process (in the studied organisation) of these providers is not fully formalised. Performance and the quality of external service providers are measured. The results presented in reports on cooperation with external partners reveal that applications built by external partners do not provide support for business processes. The details of the evaluation are given in Fig. B.12 and in Table B.9.

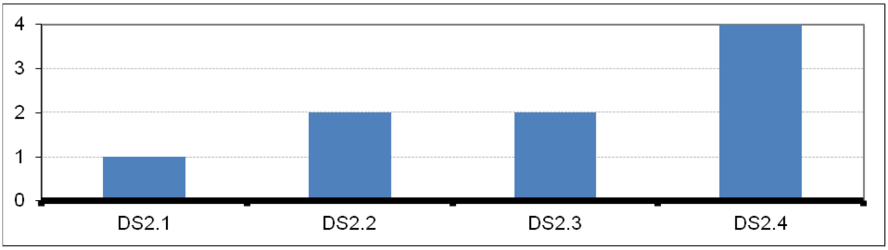


Fig. B.12 Meeting control objectives of the DS2: service management of external partners.

Table B.9 DS2 process analysis: service management of external partners

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	DS.2.4	In the studied organisation, the performance of service providers is monitored (especially for the budget system).
Processes which require significant improvement	DS.2.2	The management processes of relationships between service providers and the organisation are not formalised in the firm. However, they can be considered repeatable.
	DS.2.3	In the analysed organisation, the risks associated with the providers' ability for continuous and effective service delivery is regularly (informally) identified.
Processes which do not function	DS.2.1	In the firm, there is no formal technical and organisational documentation of relationships between service providers

DS3. Resource Performance Management

Main Objective. The control of the IT process of resource performance management in line with the business objective - guarantee that optimally applied and adequately efficient resources are available in the company.

Level: Initial

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, the resource performance and capacity management is insufficient. This process is ad hoc and sporadic. There are situations in which users perform work without taking into account the performance and capacity constraints of the organisation's resources. The owners of the organisation's business process do not always understand and see the need for IT services. IT management is aware of the need to manage both the performance and capacity of resources; however, the actions taken tend to be ad hoc and inadequate. The evaluation process of the resource capacity requires significant improvement. The details of the evaluation are given in Table B.10 and Fig. B.13.

Table B.10 DS3 process analysis: resource performance management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which require significant improvement	DS3.4	In the studied organisation, the current load and the potential resources as well as the IT capabilities are monitored. The monitoring process, however, is inadequate, making it impossible to obtain a reliable and meaningful evaluation of the resources.
Processes which do not function in the organisation	DS3.1	In the studied organisation, the process of application performance planning does not exist. Models of the current and the projected performance are not created. Application performance results from the operating activities of studying this performance. In this organisation, there is no system of ongoing evaluation of IT performance. A prediction of the performance is not carried out in the firm. This prevents the establishment of efficiency of any improvement activities.
	DS3.2	
	DS3.3	

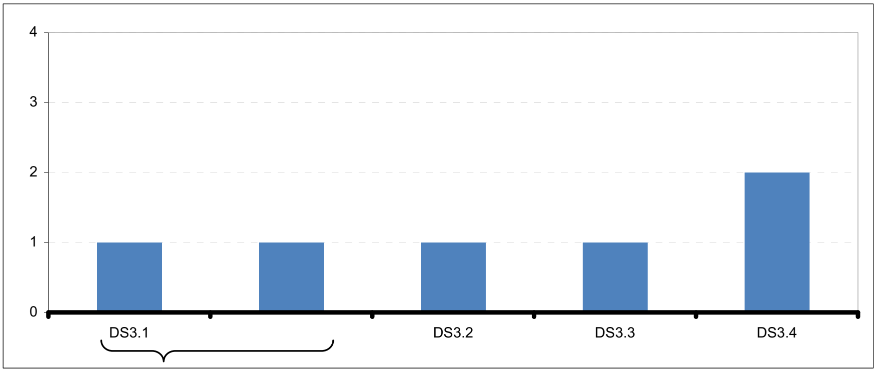


Fig. B.13 Meeting control objectives of the DS3 process: resource performance management

DS4. Ensuring Continuity of Services

Main Objective. The control of the IT process of ensuring the continuity of services in line with the business objective - a promise of the availability of services and of the minimal impact of interference on the company results.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, optimal IT service continuity is not provided. A partial approach is used. Many important aspects of the continuity of access are not included. The preparation of reports on the availability of services is incomplete. The report does not take into account the impact of the continuity of access on the company's operation. The service continuity plan or the user's plan of action related to the provision of such continuity are not defined. The organisation argues that all measures relating to the continuity of access to services are made in accordance with practice (which is based on the knowledge of the employee who is responsible for the restoration of the system). It also assures that there are obligations on the availability of services, and their main principles are known. The description of critical systems built on the basis of these services and their components is known (in terms of availability). The continuity of services and process monitoring are carried out in accordance with standard practices in the organisation. Their use depends on the person responsible for the implementation (carrying out the support processes) and does not depend on the defined and developed plan of services (delivery). The details of the evaluation are illustrated in Fig. B.14 and in Table B.11.

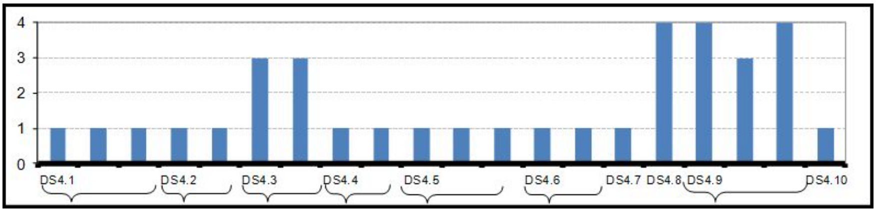


Fig. B.14 Meeting control objectives of the DS4 process: ensuring continuity of service

Table B.11 DS4 process analysis: ensuring continuity of service

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	DS4.8	The organisation plans necessary actions to be taken during the recovery and resumption of services. In the studied organisation, the DS4.9 process is realised in relation to the storage and archiving of services.
	DS4.9	In the analysed organisation, all critical backups, system documentation and other essential resources are stored in the offsite mode. Back-up copies are created and regularly updated. This ensures the validity of the data and minimises the risk of loss of the available resources.

Table B.11 (*continued*)

Processes which require slight improvement	DS4.3	In the studied organisation, in the event of resource recovery, the attention is focused on the systems critical for the functioning of the organisation. They should be set out in the Continuity Plan. Such a plan does not exist in the studied organisation, therefore, defining critical systems depends on subjective decisions.
	DS4.9	In the analysed organisation, the DS4.9 process (application stability) refers to the creation of backups for key resources. Establishing the stability of the hardware and software, whose task is to keep these copies, is not done.
Processes which do not function in the organisation	DS4.1	The organisation did not introduce a Continuity Plan, and emergency action is based on the experience of the employees. Critical resources and alternative processes are not defined. The rules for creating backups are not defined and a resource recovery policy is not established.
	DS4.2	The Continuity Plan should refer to the flexibility of services and alternative procedures, and the ability to recreate critical services. Lack of such a plan means that corrective action will not be realised. In such a situation, roles, responsibilities, procedures and processes of communication in emergency situations are not defined.
	DS4.4	The Continuity Plan should also reflect the business requirements of the organisation. Because of its lack (in the studied organisation) there is no certainty as to whether the actions taken are in compliance with those requirements.
	DS4.5	In the studied organisation, a defined Continuity Plan is not tested. It does not provide the effective and optimum performance of system recovery processes is not provided.
	DS4.6	There is no reporting system of the Continuity Plan in the organisation, which prevents tracking and implementing support. A Plan of Action supporting the Continuity Plan does not exist.
	DS4.7	Regular training sessions in the use of procedures, the assignment of roles and responsibilities in the event of a failure/break-down are not organised. Only one person deals with that.
	DS4.10	In the absence of the Continuity Plan, there is also no strategy of a distribution of the plan for the team. Such a distribution allows equal access to knowledge on emergency actions. The organisation does not have a database with information about emergency cases and action taken.

DS5. Ensuring System Security

Main Objective. The control of the IT process of ensuring system security in line with the business objective — protecting information from unauthorised access, modification and publication

Level: Defined

Evaluation of the Possibility of Achieving the Assumed Objective. It was determined that, in the studied organisation, there is awareness of the need to ensure security. Although this awareness is promoted by management, IT security procedures in all areas of the organisation are not defined. IT security reporting is performed. However, it focuses exclusively on the security of IT processes, and not on the security of the organisation itself. Testing procedures used in the organisation are only ad hoc. The details of the evaluation are given in Fig. B.15 and Table B.12.

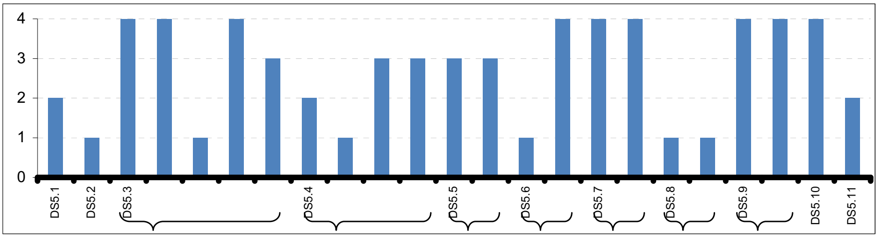


Fig. B.15 Meeting control objectives of the DS5 process: ensuring system security

Table B.12 DS5 process analysis: ensuring system security

<table>
<tr><th>PROCESS EVALUATION</th><th>PROCESS_ID</th><th>CONCLUSIONS ON PROCESS EVALUATION</th></tr>
<tr><td rowspan="5">Processes which do not require improvement</td><td>DS5.3</td><td rowspan="5">In the studied organisation, in terms of user uniqueness and identifying user identity, the DS5.3 process shows that all users (internal, external and temporary) and their activity in the IT system (business applications, IT environment, system operations, development and maintenance) are unique phenomena. Identification is thus possible via authentication mechanisms. In the case of the CRM, which is used in the organisation, the identification is insufficient, because the system in the OpenSource version has authorisation restrictions. Rights of access to other systems are given to managers and are confirmed by the owners of the systems, they are also guarded by the person responsible.
The DS5.6 process refers to awareness of the risks in the system. Members of the organisation are always informed about them, and their duty is to minimise these risks. The analysed company has the technology to prevent damage to the system, and access to data is strictly controlled.
The studied organisation has the means to detect viruses, spam and spyware. There is also a central database which provides up to date virus definitions.
The organisation possesses and uses firewall techniques.</td></tr>
<tr><td>DS5.6</td></tr>
<tr><td>DS5.7</td></tr>
<tr><td>DS5.9</td></tr>
<tr><td>DS5.10</td></tr>
<tr><td rowspan="3">Processes which require slight improvement</td><td>DS5.3</td><td rowspan="3">In the studied organisation, the DS5.3 process refers to the repository. The identity of users and their access rights are stored in a central repository, IDAP.
In the case of the tested system, the CRM is not integrated with the IDAP.
In the studied organisation, the DS5.4 process (rights and obligations connected with the access and review of rights) means the rights and obligations related to access to the information system, which is managed through an appropriate application. A regular review of user accounts and rights associated with them for the CRM system is not carried out in the organisation. This access is monitored.
IT security processes are tested and monitored. There is, however, no possibility of estimating the quantity of these processes.
The studied system does not contain functions of data recording and monitoring which allow for the early prevention or detection of the system malfunctioning.</td></tr>
<tr><td>DS5.4</td></tr>
<tr><td>DS5.5</td></tr>
</table>

Table B.12 (*continued*)

Processes which require significant improvement	DS5.1	The organisation does not have a Security Plan. In the studied system, user authentication is present. This system is local, and its application and security are individual in their nature. It is not treated as a set of best practices or as a pre-prepared security plan.
	DS5.4	The DS5.4 process (account management procedures) applies to the evaluation of the level of account availability in the company. It was determined that a request for an address, establishing the user, data publication, suspension of access and the related rights are not executed according to user account management procedures. It was noted that the organisation is currently working on a solution which will not require any procedures for managing user accounts.
	DS5.11	The exchange of data through an internal network of the organisation (intranet) is controlled to ensure the reliability of the data. However, it is not controlled if the data comes from an external network.
Processes which do not function in the organisation	DS5.4	In the studied organisation, the DS5.4 process (procedure validity) shows that there is a lack of user account management procedures. Establishing the user groups which undergo these procedures is thus difficult.
	DS5.6	The DS5.6 process (determining the characteristics of potential incidents) does not occur in the organisation. No signs of potential incidents connected with security which block the functioning of the CRM system have been identified.
	DS5.8	The organisation does not have procedures for generating cryptographic keys, their modification, cancellation, distribution, storage, access, use and archiving. It is assumed that those keys have a task to provide protection against modification and unauthorised access to the system resources. Consistently, the studied organisation has no publicly available catalogue of keys.

DS6. Cost Identification and Allocation

Main Objective. The control of the IT process of (cost identification and allocation in line with the business objective) ensuring the proper allocation of costs to IT services.

Level: Managed

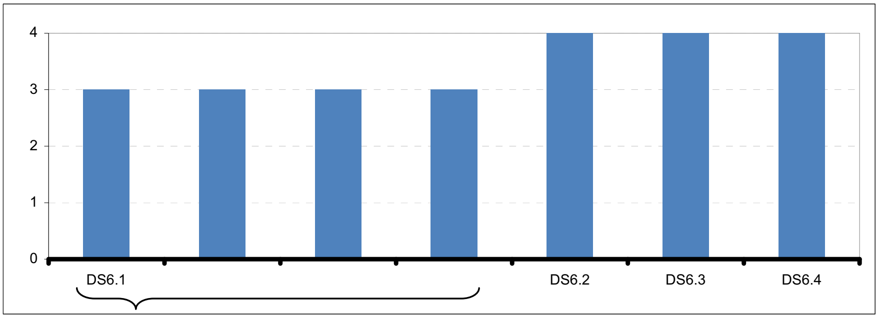


Fig. B.16 Meeting control objectives of the DS6 process: cost identification and allocation

Evaluation of the Possibility of Achieving the Assumed Objective. The studied organisation defines the duties and responsibilities within managing the cost of IT services. The direct and indirect costs are identified and reported through a budgeting system for the benefit of the management and project partners. The organisation has processes to monitor and evaluate the costs, and when the processes are ineffective or exceed certain limits, appropriate action is taken. This does not occur in all the cases when costs overrun. Cost management processes are in line with the best internal practices. Reporting of the level of IT costs is integrated with business objectives and SLAs. The details of the evaluation are given in Fig. B.16 and in Table B.13.

Table B.13 DS6 process analysis: cost identification and allocation

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	DS6.2 DS6.3 DS6.4	The tested organisation uses a model to track the costs, which allows for making prognoses about them, and a model showing the distribution of future costs. The organisation regularly controls the costs of the applied IT system (CRM).
Processes which require slight improvement	DS6.1	In the studied organisation, there are no breakdowns of the IT cost. It is assumed that it is constantly monitored. IT expenditures are analysed when they exceed a certain budget. This process, however, does not function across the organisation. Low costs of investments, fixed fees and royalties are distinguished on the level of the accounting system. Because of their small scale, the fragmenting of costs into separate organisational units is not necessary.

DS7. User Training

Main Objective. The control of the IT process of user training in line with the business objective – ensuring that users apply information technologies in the correct way and that they are aware of the risk and responsibilities connected with their application.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. In the organisation, there is a faint awareness of the need for the training and education of staff. These training sessions are relevant to the individual plans of employee development. The knowledge level of individual workers is monitored. However, advanced information technologies are not used to formalise this information. The details of the evaluation are presented in Fig. B.17 and Table B.14.

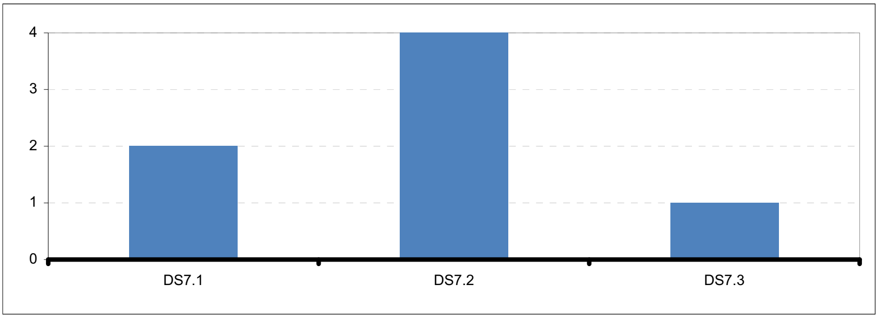


Fig. B.17 Meeting control objectives of the DS7 process: User training

Table B.14 DS7 process analysis: User training

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	DS7.2	The organisation identifies specific groups of employees to increase their competence.
Processes which require significant improvement	DS7.1	The organisation provides training for employees on the current and future trends of its development. These training sessions are not regular. During training, possible upgrades of hardware and software are considered with regard to future business trends.
Processes which do not function	DS7.3	The analysed organisation does not conduct a valuation of the commissioned training, or its evaluation in terms of the benefits received because these activities are small scale.

DS8. Incident Management in the Service Desk

Main Objective. The control of the IT process of incident management in the Service Desk in line with the business objective – effective use of IT systems, ensuring the resolution and analysis of end-user queries and emerging incidents

Level: Repeatable

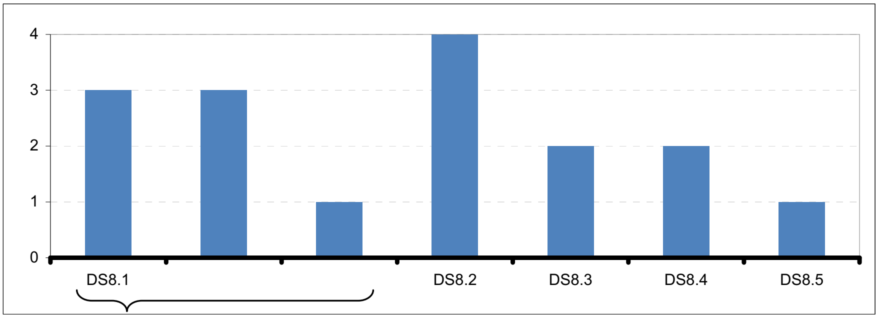


Fig. B.18 Meeting control objectives of the DS8: incident management in the Service Desk

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, there is awareness of the need for the functioning of a Service Desk and an incident management process. The analysis of a current case when support was needed (involving the people responsible) points to its informality. The organisation does not provide any formal training, nor promote standard support procedures. The functioning of a support system relies on the employees. It is worth noting that the exchange of information on the functioning of the support system is constant in the company. The details of the evaluation are described in the material shown in Fig. B.18 and Table B.15.

Table B.15 DS8 process analysis: incident management in the Service Desk

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	DS8.2	The organisation has a system for collecting user queries and comments.
Processes which require slight improvement	DS8.1	Service Desk functions are only assumed for external projects. In internal projects, all comments are directed to the appropriate Project Manager and then to the administrator of the system. However, there is no formal system for reporting service needs. This may hinder the process of finding the person responsible, and thus lengthen the response time to a specific request.
Processes which require significant improvement	DS8.3	In the studied organisation, the monitoring of a problem is carried out on a current basis by the person who reported the problem. The notification is possible only by establishing direct contact with the person responsible for providing support. In an organisation in which departments are not very large, this is not an issue. As the organisation develops, however, this approach may be less effective. For example, when resolving new queries, it is not considered whether a similar problem has already been solved before, and whether the problem's solution can be based on previous experience. Lack of such verification is connected with the lack of a knowledge base collecting this type of data and of the employment of people who have this experience.
	DS8.4	
Processes which do not function	DS8.1	The organisation does not measure user satisfaction on the basis of the quality of services provided by the Service Desk.
	DS8.5	The process is not applicable to the studied organisation. Errors which often recur, are not identified.

DS9. Configuration management

Main Objective. The control of the IT process (configuration management) in line with the business objective, which is: accounting for all IT components, the prevention of unauthorised changes to the system, and the proper management of change

Level: Defined

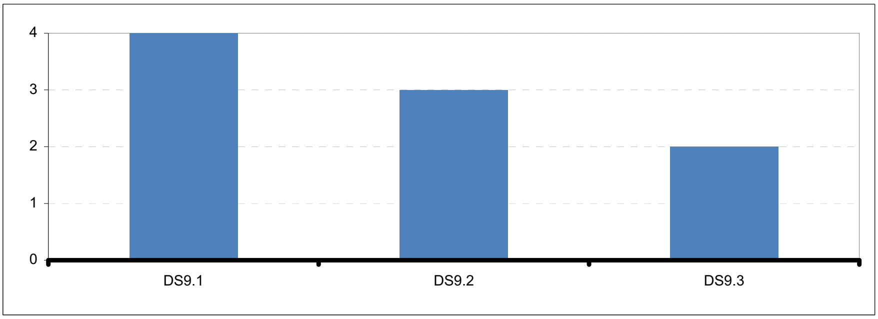


Fig. B.19 Meeting control objectives of the DS9 process: configuration management

Table B.16 DS9 process analysis: configuration management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	DS9.1	In the studied organisation, the initial state of system configuration is preserved. It is assumed that for each system, this is the point to go back to after changes. This ensures continuity and minimises the risk of data loss.
Processes which require slight improvement	DS9.2	In the studied organisation, there are no procedures to support the management of change. Still, in the existing configuration repository, each project has its resources. Each project manager approaches the change management process in an individual way, making it impossible to standardise the process. Maintaining a uniform level of project quality is therefore difficult.
Processes which require significant improvement	DS9.3	The studied organisation does not conduct a periodic review of the installed software to detect private, illegal or unnecessary software. Every employee signs a document obliging them to refrain from installing illegal software, but there is no control system to ensure that this rule is complied with.

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, there is thorough and competent information about the system configuration. The procedures and work practices applied are not documented, nor are subject to the procedures of standardisation. It is assumed that any training in this area is individual. There are cases in which tools can be used to automatically track changes in the software. In the organisation, this is done manually (if at all). The details of the evaluation are shown in Fig. B.19 and Table B.16.

DS10. Incident Management

Main Objective. The control of the DS10 IT process (incident management), to ensure that incident problems are resolved and the reasons for their occurrence are identified.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. The organisation is aware of the need to manage problems associated with the presence of incidents in the area of IT and business. This process is ad hoc, without including key employees. The level of services differs, and their performance is hampered by insufficient technical and organisational knowledge. The process is in need of improvement in the areas of analysing the problem of incident occurrence and of reporting them. The details of the evaluation are described in Fig. B.20 and in Table B.17.

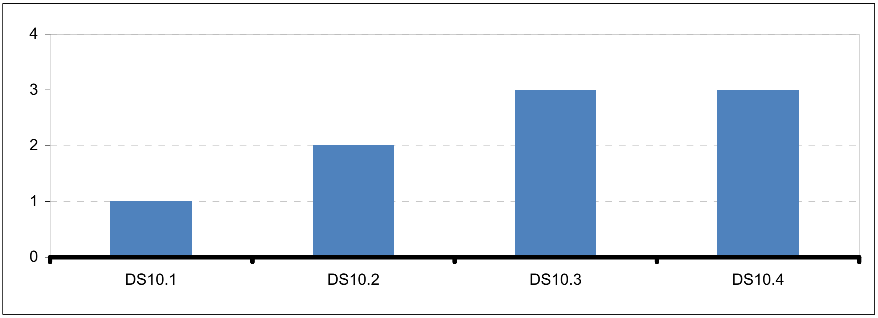


Fig. B.20 Meeting control objectives of the DS10 process: incident management

Table B.17 DS10 process analysis: incident management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which require slight improvement	DS10.3	In the studied organisation, there are no closing procedures of the incident occurrence problem. These procedures could be used after confirming that errors have been eliminated, or after applying an alternative solution. The organisation uses practices to solve problems and when incidents occur.
	DS10.4	Configuration and unexpected event management processes are applied in the organisation. These processes are performed by a single employee. With the development of the organisation and with the increased number of employees, it will be necessary to use formal procedures for configuration management.
Processes which require significant improvement	DS10.2	The organisation often ignores the main causes of issues which emerge. This makes it impossible to analyse them later and take preventive action.
Processes which do not function in the organisation	DS10.1	Defined procedures for problem reporting and classification are not applied in the organisation.

DS11 . Data Management

Main Objective. The control of the IT process of data management in line with the business objective – to ensure that the entire process of entering, updating and storing data remained complete and at a high level of accuracy.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, data properties can be analysed only to a limited extent. Methods allowing the prevention and detection of errors were selected. However, their use is limited to individual employees and not the entire organisation. Data processed by the IT system existing in the organisation is in line with business requirements, but its safety is not supported by procedures. The details of the evaluation can be found in Fig. B.21 and in Table B.18.

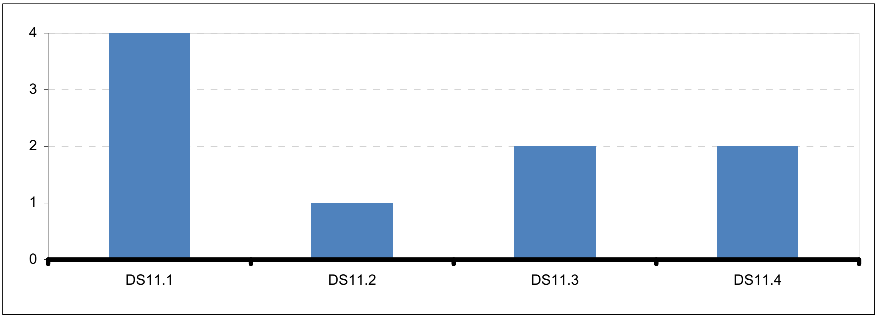


Fig. B.21 Meeting control objectives of the DS11 process: data management

Table B.18 DS11 process analysis: data management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	DS11.1	In the studied organisation, all the data which is delivered as a result of the realisation processes is processed entirely in accordance with the business requirements in the allocated time.
Processes which require significant improvement	DS11.3	In the organisation, there are no procedures to ensure the protection of data and software when is it transferred or uninstalled.
	DS11.4	There are procedures for restoring the application, its data and documentation in accordance with the business requirements (they are considered as best practices of the organisation).
Processes which do not function	DS11.2	In the analysed organisation, there are no procedures for manufacturing effective and efficient databases and procedures for their archiving in accordance with the business objectives. The organisation does not see the need for the preparation of such procedures, provided that the current situation is sufficiently effective. Such evaluation, however, can be subjective and may not reflect reality. It can also block the improvement of the functioning of the studied organisation.

DS12. IT Infrastructure Management

Main Objective. The control of the process of IT infrastructure management. Control processes are made possible by the application of a suitable environment the task of which is to monitor and control the IT infrastructure and staff.

Level: Managed

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, it can be clearly stated that the intention to maintain a controlled IT environment is fully justified. Security requirements have been documented and access to this environment is monitored and restricted. Those responsible for the infrastructure have been properly trained to make decisions in emergency situations. In the studied organisation, standard control mechanisms have been installed, to limit access to certain elements of the infrastructure (*e.g.* proximity cards). The details of the evaluation of this (DS12) aspect of the domain: delivery of IT services, have been included in Fig. B.22 and in Table B.19.

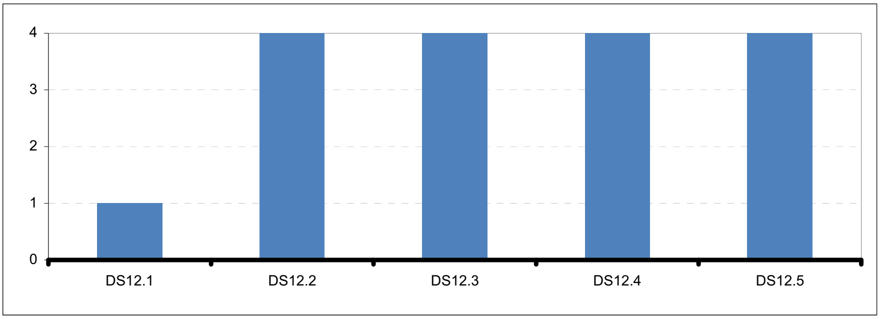


Fig. B.22 Meeting control objectives of the DS12 process: IT infrastructure management

Table B.19 DS12 process analysis: IT infrastructure management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	DS12.2 DS12.3 DS12.4 DS12.5	In the studied organisation, measures are taken to prevent external risks (theft, high temperature, fire, water, terrorism, vandalism), such as proximity cards. The use of cards allows access to buildings in the organisation to be individually assigned. Such access should be justified, requiring a login, and monitored. Special equipment and devices to monitor and control the environment, including cameras, have been installed.
Processes which do not function in the organisation	DS12.1	GUI designers did not take into account the risks associated with human error. Therefore, the system used in the organisation is sensitive to frequently occurring simple user errors.

DS13. Operations Management

Main Objective. The control of the IT process of operations management is to ensure that important IT tasks referring to support are performed regularly and in an organised manner.

Level: Managed

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, the assignment and ownership rights of the ongoing operations are well defined. These operations are supported by the organisation resources allocated to investments and the employment of appropriate personnel. Training is not always connected with the development of employees. Daily consumption of resources and the degree of fulfilment of the assigned tasks are measured and monitored in the organisation. Work on the automation of the process and its continuous improvement is carried out. It was determined that there are some deficiencies in the planning of activities and procedures to ensure IT continuity in the organisation. The details of the evaluation of the process are given in Fig. B.23. and Table B.20.

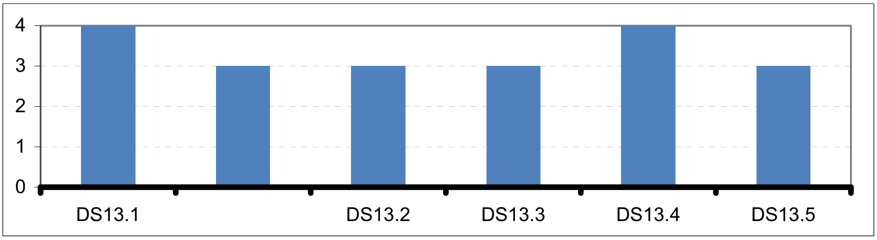


Fig. B.23 Meeting control objectives of the DS13 process: operations management

Table B.20 DS13 process analysis: operations management

<table>
<tr><th>PROCESS EVALUATION</th><th>PROCESS_ID</th><th>CONCLUSIONS ON PROCESS EVALUATION</th></tr>
<tr><td rowspan="2">Processes which do not require improvement</td><td>DS13.1</td><td rowspan="2">The DS13.1 process (defining responsibility) is very well defined in the studied organisation, although there are no repeatable, implemented procedures for IT operations. The employees of this organisation are only familiar with the tasks relating to the implementation of operations which were assigned to them. Appropriate procedures for handling secret documents are present here, providing safe storage, data archiving and IT resource management.</td></tr>
<tr><td>DS13.4</td></tr>
<tr><td rowspan="4">Processes which require slight improvement</td><td>DS13.1</td><td rowspan="4">In the studied organisation, the DS13.1 process (application of continuity procedures) showed that there are no such procedures. Therefore, the organisation does not guarantee that the continuity of IT operations will be preserved.
Practical considerations show that the initially defined responsibility is sufficient here.
In the organisation, a specific work plan, a plan of processes and a plan of tasks, ensuring their effective implementation order and increasing the capacity of the processes, do not exist.
In the studied organisation, IT infrastructure monitoring procedures are largely defined and implemented.
There are no procedures for preventive action to reduce the frequency of failures and the unavailability of equipment.
To solve this problem, an appropriate virtualisation system is used.</td></tr>
<tr><td>DS13.2</td></tr>
<tr><td>DS13.3</td></tr>
<tr><td>DS13.5</td></tr>
</table>

Domain ME: Monitoring

ME1. Monitoring and Evaluation of IT Performance

Main Objective. The control of the IT process of monitoring and evaluation of IT performance – the aim of this process is to ensure the transparency of determining IT costs, benefits, strategy and policy in relation to the information technology applied in the organisation

Level: Defined

Evaluation of the Possibility of Achieving the Assumed Objective. The organisation management establishes and provides standard processes for monitoring the degree of information technology application. For this, the organisation has implemented a monitoring programme, under which an evaluation of individual processes and IT projects is carried out from time to time. This evaluation does not apply to all processes. Tools are implemented to monitor the internal IT processes and services provided. The contribution of IT to improve the organisation's performance has been defined by traditional financial and operational criteria. Problems occur in defining the criteria for achieving the organisation's objectives and the integration of monitoring with the organisation management processes. The details of the ME1evaluation process, in the studied organisation, can be found in Table B.21 and Fig. B.24.

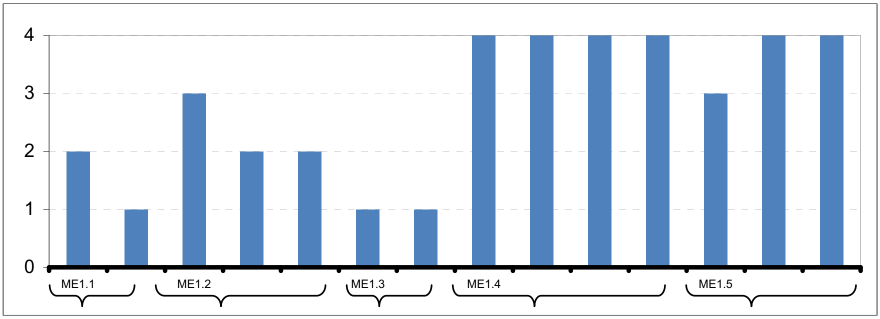


Fig. B.24 Meeting control objectives of ME1: monitoring and evaluation of IT performance

Table B.21 ME1 process analysis: monitoring and evaluation of IT performance

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	ME1.4 ME1.5	The organisation makes periodical comparisons of the results achieved with the set objectives and analyses the causes of performance deviations from the targets. The project manager is responsible for this analysis. As the needs of the organisation develop, preventive action is initiated so that achievements overlap with the set objectives as much as possible. Then comparisons are made again to evaluate the condition. Reports determining the risks incurred in the implementation of the set objectives are created as part of the ME1.5 process (risk reporting and remediation) in the organisation. On this basis, preventive actions are suggested to the senior management.
Processes which require slight improvement	ME1.2 ME1.5	In the analysed organisation, the ME1.2 (defining objectives) needs to be improved. The objectives of the organisation are defined and accepted. The environment affects the organisation and has an impact on the achievement of the operational objectives (not strategic). The ME1.5 process (reporting achievement of objectives) also requires minor changes. Senior management receives reports containing information on the degree of achievement of the set objectives. The expenditure/resources of the organisation are taken into account. The degree of objective achievement is reported verbally, indirectly by the project manager (not as a report to management).
Processes which require significant improvement	ME1.1 ME1.2 ME1.5	In the studied organisation, the ME1.1 process (IT monitoring) needs considerable improvement. The firm has an incorrect approach to defining the scope and methodology of monitoring. Within the monitoring system, notifications and their reporting time are monitored. The ME1.2 in the studied firm (adopting criteria for objectives and reporting their realisation) also requires significant improvement. The criteria for achieving the firm's objectives are not defined, and the achievement degree of the goals is not be measured. Reports which reflect the realisation of objectives do not exist. Such information can be obtained, but first the resources in which it is located must be identified.
Processes which do not function in the organisation	ME1.1 ME1.3	In the organisation, within the ME1.1 process (integration of the monitoring framework), the IT monitoring framework is not integrated with the company performance management system. The performance monitoring method has not been determined and specified in the organisation, which prevents such an evaluation.

ME2. Monitoring and Evaluation of the Internal Control Performance

Main Objective. The control of the IT process of the monitoring and evaluation of the control performance in the organisation – aims at determining whether these objectives have been met.

Level: Defined

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, the internal control monitoring process has not been applied to the whole institution. This process has not been supported by evaluation procedures of the reporting level of the actions taken to monitor this control. However, the organisation has mechanisms and tools for the control process. Still, they are not integrated with all processes. The details of the evaluation are given in Fig. B.25 and in Table B.22.

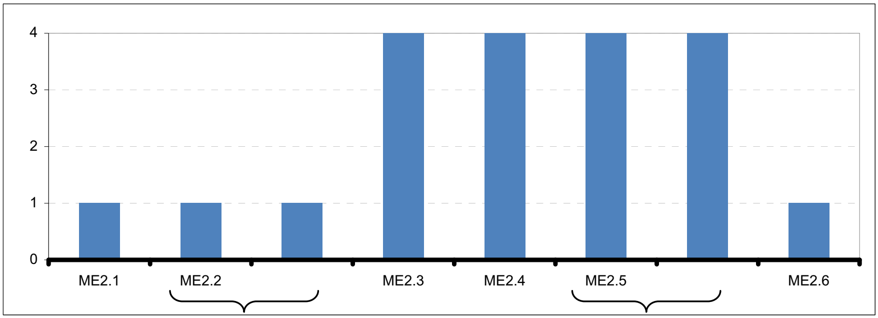


Fig. B.25 Meeting control objectives of the ME2 process: monitoring and evaluation of the internal control performance

Table B.22 ME2 process analysis: monitoring and evaluation of the internal control performance

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	ME2.3	The studied organisation evaluates the effectiveness of the IT process management control.
	ME2.4	This evaluation refers to the status of external providers and internal processes of the organisation.
	ME2.5	This evaluation is not carried out in relation to the CRM system used in the organisation.
Processes which do not function in the organisation	ME2.1	In the analysed organisation, the monitoring process and the process of evaluating the efficiency and effectiveness of IT, the results of which were expected by the evaluator, do not exist.
	ME2.2	The regulating exceptions were not defined and thus cannot be analysed.
	ME2.6	The organisation also does not introduce repair tools for the information technologies applied, as their condition is not monitored continuously.

ME3. Ensuring Compliance with External Requirements

Main Objective. The control of the IT process of ensuring compliance with external requirements for the growth of trust and cooperation between the organisation and the external partners.

Level: Repeatable

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, the IT management has implemented operations management processes, aiming at obtaining an independent evaluation of the compliance level with external constraints, which are directly related to the needs and requirements of the business. The details of the evaluation are presented in Fig. B.26 and Table B.23.

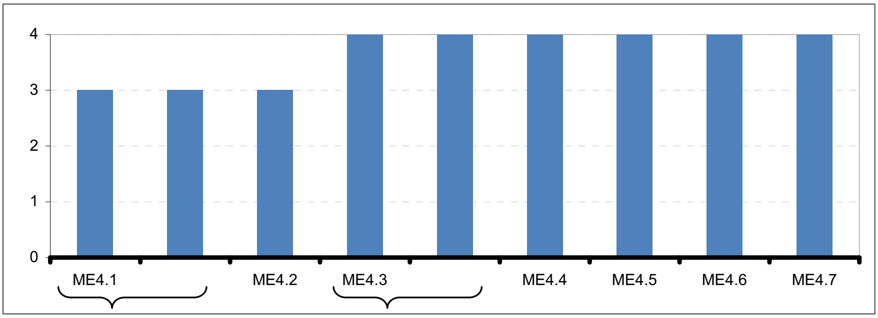


Fig. B.26 Meeting control objectives of the ME3 process: ensuring compliance with external requirements

Table B.23 ME3 process analysis: ensuring compliance with external requirements

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	ME3.1	In the studied organisation, the ME3.1 process (compliance with regulatory requirements) is fully realised. All legal requirements are clearly defined.
Processes which require slight improvement	ME3.2	In the studied organisation, despite the lack of an IT policy, the compliance of any IT operations with legal requirements is confirmed.
Processes which do not function in the organisation	ME3.1	In the studied organisation, no international standards (*e.g.* ISO) are used in the IT area.
	ME3.2	The process is not applicable to the studied organisation.

ME4. Ensuring IT Process Management

Main Objective. The control of the IT process of ensuring IT process management. As a result of this process, the integration of IT management processes with corporate management objectives, consistent with the requirements and regulations, should be achieved.

Level: Managed

Evaluation of the Possibility of Achieving the Assumed Objective. In the studied organisation, there is a full understanding of the need to manage IT at all levels. It has been specified who the client is, and the responsibility within the SLA has been defined and monitored. The IT processes and their management are integrated with the management of the organisation. Appropriate mechanisms for IT investment management and resource allocation have been introduced, to ensure the optimum functioning of the organisation. The details of the evaluation are given in Table B.24 and Fig. B.27.

Table B.24 ME4 process analysis: ensuring IT process management

PROCESS EVALUATION	PROCESS_ID	CONCLUSIONS ON PROCESS EVALUATION
Processes which do not require improvement	ME4.3 ME4.4 ME4.5 ME4.6 ME4.7	The studied organisation conducts IT investment management to ensure the best effects of these investments. All investment decisions made in the organisation are economically justified. It was also determined that there is an appropriate allocation of IT resources in order to ensure that the strategic objectives and the cooperation between IT managers and organisation managers is realised to reduce the risk of unjustified decisions. In the analysed organisation the objectives have been achieved and the results are in line with expectations. The decision-makers in the organisation obtain independent information that the IT area complies with the generally accepted legal requirements and practices applicable in the organisation.
Processes which require slight improvement	ME4.1 ME4.2	In the studied organisation, the IT management structure has not been defined, nor adapted to the overall management of the organisation, due to the very small size of the organisation.

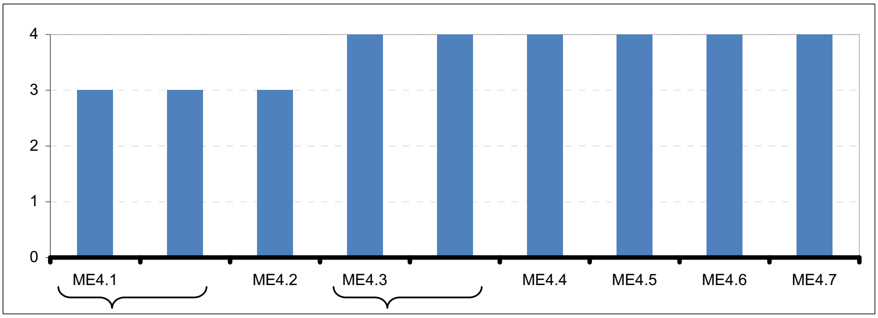


Fig. B.27 Meeting control objectives of the ME4 process: ensuring IT process management

The Organisation Maturity Model in the PO, AI, DS and ME Categories

The organisation maturity model, taking into account the 34 control processes within the 4 main domains (PO, AI, DS, ME), allows the evaluation of the level of the organisation according to the introduced structure. For the qualification of maturity, limit values have been used to provide level differentiation, according to the material presented in Table B.25.

Table B.25 Limit values used in the maturity qualification of the organisation

Number (Level)	The degree of fulfilment of specific objectives (extreme values)	Name of the level
0	0.00%	non-existent
1	20.75%	initial
2	41.50%	repeatable
3	62.25%	defined
4	83.00%	managed
5	100.00%	optimising

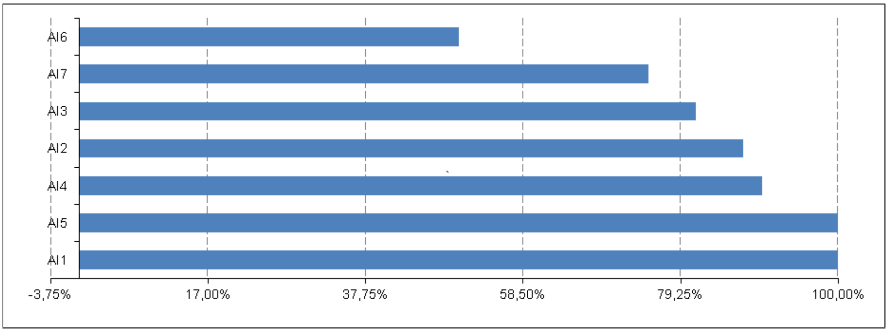


Fig. B.28 The degree of maturity of the control processes in the AI domain

The evaluations of the control processes in the domain of planning and organisation, presented in the main part of the book (Section 4.5), exhibit a uniform distribution of the degree of maturity (Table 4.13) of each of the processes, *i.e.* there are both processes that require significant improvement, as well as those which are working almost optimally.

The organisation has achieved much better results in the area of AI – acquisition and implementation (Fig. B.28), in which the maturity of all the processes exceeds the initial level while two control processes:

- identification of defined and repeatable processes
- IT resource acquisition,

have obtained the highest evaluation, which means that they were managed in an optimal way.

The maturity of the remaining control processes of this domain has reached high reckons, thus it can be argued that the company is fully mature in the area of software implementation.

The maturity of the processes within the DS domain – delivery (Fig. B.29) is similar to that in the domain of planning and organisation (PO). The detailed evaluations of these processes are distributed almost evenly into three stages of maturity. If the company management is considering a transition to a higher level of maturity, they should pay attention to the improvement of these processes:

- defining services and managing their level (DS1)
- managing the organisation potential (DS3)
- ensuring continuity of service (DS4).

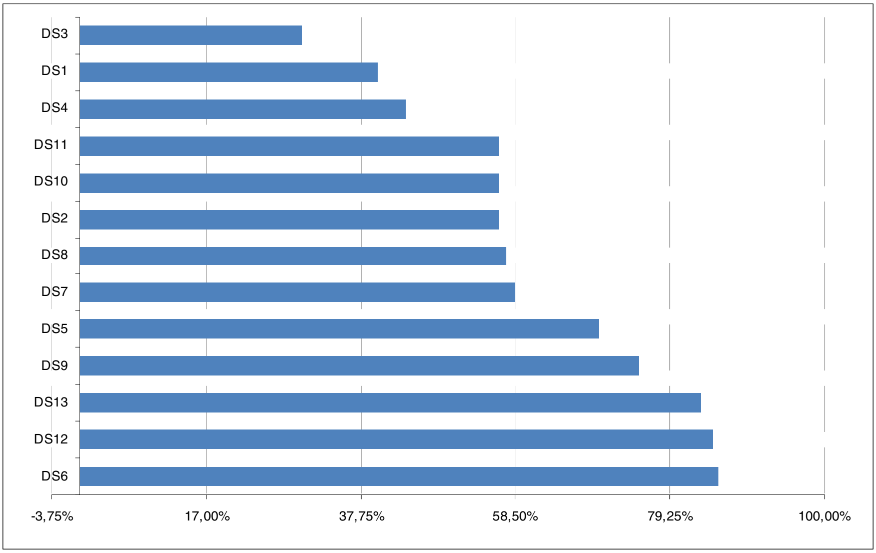


Fig. B.29 The degree of maturity of the control processes in the DS domain

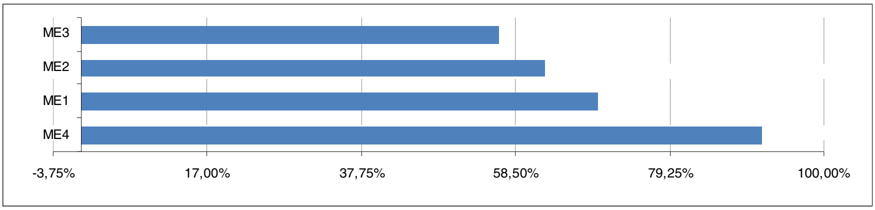


Fig. B.30 The degree of maturity of the control processes in the ME domain

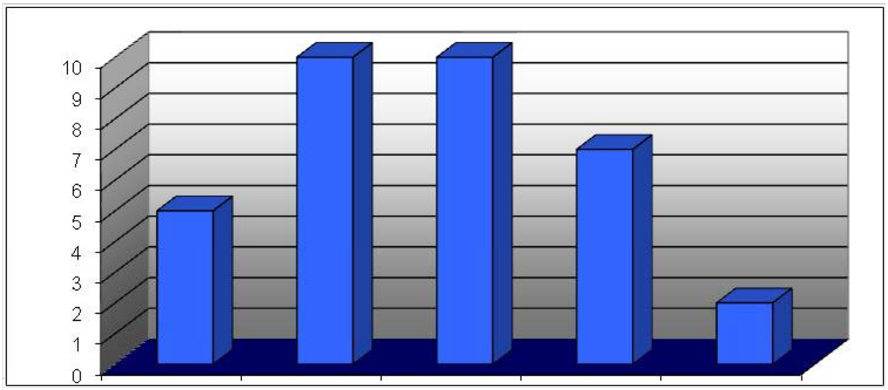


Fig. B.31 The degree of organisation maturity

The evaluation of processes carried out in the ME domain – monitoring (Fig. B.30) has very average results, because there is only one well-managed process (IT management) and three processes at a close level, within the Defined level.

Arranging the above evaluations, we conclude that in order to improve management maturity, the organisation should improve the processes from the ‘planning and organisation’ domain as well as the ‘delivery and implementation’ one. The maturity degree of the organisation, expressed by the specification of evaluations of its process, is illustrated in Fig. B.31: it presents a number of processes, at an established level of maturity of the organisation. More specifically, the number of processes in the control areas were presented which have reached a given maturity level in the studied organisation, *i.e.*:

- 5 processes at the initial level (domains DS, PO)
- 10 processes at the repeatable level (domains AI, DS, ME, PO)
- 10 processes at the defined level (domains AI, DS, ME, PO)
- 7 processes at the managed level (domains AI, DS, ME, PO)
- 2 processes at the optimising level (domains AI).

Therefore, the above list, and Figure B.31, contain processes which require significant improvement (the initial and repeateble levels), and those which should be developed (the defined, managed and optimising levels).

B.2 Summary

This Appendix contains the remaining study results (apart from the PO domain – organisation and planning) relating to the maturity evaluation of the client organisation which is implementing a CRM system. This material can be support for those who carry out audits of IT organisations on the basis of the COBIT standard. The evaluation was performed according to an extended set of questions, comparing with the standard approach, referring to the realisation of objectives in the process of particular domains. The significant simplification of the client organisation evaluation procedure and the extent to which the control objectives were realised was achieved through the use of the linguistic description.

The results presented in this Section are grouped according to domains (AI, DS, ME), with the use of the classical COBIT approach. The novelty here are the tables containing the description of the processes and conclusions prepared for the linguistic models of the IPP.

This Appendix is a compendium of knowledge about the organisation implementing the CRM system. Which was used in a studied organisation. The linguistic analysis carried out facilitated the evaluation of processes in this organisation: On the basis of the analysis, those that require significant improvement were identified. The aggregated lists contained in the tables and figures summarising the evaluation, may constitute a starting material for establishing the strengths and weaknesses of the organisation.

The material contained in this Appendix may also be useful in expanding the MITM model presented in this book, and the given detailed specification of processes of the organisation can be a model for re-examination of the IPP and the maturity capsule. This Appendix may provide an example of how to carry out the evaluation of a company which is implementing IT systems.

In view of the fact extensive material on this subject is available on the internet, we do not present the full specification of COBIT here, but only what has already been directly used in the construction of the model, and what seemed to be the most useful.

C

Appendix
Provider Organisation Maturity

In order to support the description of the ‘provider organisation maturity’ variable, in this Appendix we shall present supplementary material prepared in accordance with the ITIL and TOGAF standards. The method of adapting IT standards to users’ needs will be illustrated on the basis of the ITIL standard, version 3. Although both standards are currently valid, the choice made is related to the intention to present a strategy which is clearer for recipients.

At the beginning of this Chapter we shall present a short description of the ITIL standard (the CMMI model is described in Section 2.2). After introducing the ITIL standard, we shall focus on the presentation of manufacturing and management processes implemented in the provider organisation, since familiarity with these processes is important from the point of view of the evaluation of the organisation’s maturity. We will also present frameworks for architecture development compliant with ADM processes. They are useful when an organisation examines and improves its own processes (which is expressed in the categories of provider organisation maturity). Therefore, data related to development processes is also included in this Appendix. Taking into account the ongoing examinations of both standards (CMMI and ITIL), in this Appendix we shall only present their basic description, necessary to understand the terms and variables of the considered model [4, 28, 53, 60, 41, 63].

C.1 Specification of Provider Maturity in the IPP Model

The general definition of the ITIL standard was presented in the main part of the work as the equation (2.14). The definition refers to the general description of the standard in version 2. In this version, services were divided according to service management areas. The two areas that are most widely-recognised and most often used for the evaluation of an organisation providing services are *Service Delivery* and *Service Support* [41].

Version 3 is based on the philosophy of a service life cycle divided into five stages:

- preparation of a service strategy for the organisation (*Service Strategy*)
- designing services provided by the organisation (*Service Design*)
- implementation of the designed services (*Service Transition*)
- operation of the implemented services (*Service Operation*)
- continual improvement of the services provided (*Continual Service Improvement*).

In the standard, relationships between the different components of services are also defined, as well as the influence of changes in those components on the service life cycle. Below we present a short description of individual phases in a service life cycle.

Service Strategy

This phase corresponds (in accordance with TOGAF) with the phase of building an architecture vision [18]. The main objectives for organisation functioning are established which are to be fulfilled with the help of services offered by the company. The strategy defines the resources required for the performance of the services. To plan the strategy the following are assumed:

- perspective – approaches to fulfilling the objective
- position – adopting a point of reference
- plan – defining the method of approach
- patterns – related to using manufacturing and management standards.

When defining services on the strategy level, the following service assets are needed:

- resources, which include IT infrastructure, people, finance, *etc*.
- capabilities, which include the possibility to use the resources.

The result of this phase is a service strategy and a *Service Portfolio*, which includes:

- catalogue of ready-made services (*Service Catalogue*)
- plan of new services (*Service Pipeline*)
- list of services no longer provided (*Retired Services*).

In order to perform provider services (as part of building the service strategy), the following processes are required:

- *Financial Management*
    - ➢ the process is focused on the evaluation of the actual costs of performing the services and supports the planning of resources for the improvement of organisation maturity

- *Demand Management*
  - the process includes actions whose purpose is to organise suggestions and expectations of the offers of provider organisation
- *Service Portfolio Management*
  - the aim of the process is to support services with the help of relevant means.

Service Design

The main objective of this phase is the design and manufacturing of services on the basis of business requirements and the objectives of the strategy phase. The phase includes the following service design and manufacturing stages:

- manufacturing of new services / modification of existing services
- design of a service catalogue
- preparation of technology architecture for the designed services and their management systems
- preparation of support processes for the services
- design and preparation of processes
- preparation of indicators and service measuring methods with the use of the indicators.

In accordance with the ITIL ver. 3, it is recommended that during the design and manufacturing of services, resources are considered which include human resources, products which support manufacturing processes, specifications of those processes and partners taking part in the manufacturing processes. The product of the service design phase is a description of the SDP service (*Service Design Package*), which provides for the specification of resources which are indispensable in the manufacturing process.

In the service design phase, the following processes are used: *Service Level Management, Service Catalogue Management, Availability Management, Information Security Management, Capacity Management, IT Service Continuity Management, Supplier Management.*

The aim of the process of service level management is to align the delivery conditions for a given service with the needs of a specific client. The conditions are described in the SLA. The agreement is negotiated between the client (conditions defined in the service catalogue) and the provider on the basis of client requests specified in the document of *Service Level Requirements*.

The aim of the service catalogue management process is to collect and maintain a consistent service catalogue for the purposes of the client organisation which includes both available services and services which are being manufactured.

The aim of the availability management process is to monitor the availability of the performed services on the basis of initially drafted SLAs. On the other hand, the purpose of the capacity management process is to monitor resources in this area. Similarly, the information security management process is aimed at the manufacturing of security rules for the services delivered to the client which are compliant with the SLA protocol and general principles implemented in the company. The IT service continuity management process is also intended to support the availability of resources in the event of discontinued service delivery. In the course of service design, provider management is also considered, which creates an environment for cooperation with providers in order to ensure a service level compliant with the SLA.

Service Transition (Implementation)

The aim of this phase is to ensure conditions for the designed and manufactured services to be implemented in the provider organisation in accordance with the SLA. While implementing the services, 12 rules are adopted whose purpose is to improve the processes:

Rule 1. *before you start implementing services, adopt, define, present and monitor the observance of the service implementation rules*
Rule 2. *remember that changes in the services and manufacturing processes implemented by the IT support organisation should be implemented in compliance with rule 1*
Rule 3. *during the implementation of services, use common templates and standards*
Rule 4. *during the transition, use the processes and systems existing in the organisation*
Rule 5. *successful transition is ensured by matching it with the client business objectives*
Rule 6. *during the implementation, create and cultivate relationships between partners*
Rule 7. *create control tools for the used resources, tasks and the distribution of responsibilities*
Rule 8. *create environments for collecting and processing knowledge to support decision making*
Rule 9. *use the processes of configuration management when implementing new versions of services*
Rule 10. *together with the partners, coordinate plans for changes in the implemented services*
Rule 11. *combine the implementation of services with the financing of processes which support the implementation*
Rule 12. *ensure the quality of implemented services and use good practices of the organisation in this regard.*

If the services in the organisation are being improved, the processes need to be justified with consideration to the following aspects:

- preparation of a short justification, whose main purpose is to indicate the need for using the services and to translate them into business objectives of the organisation

- emphasis on the importance of design in accordance with the existing standards, instructions and relationships
- preparation of an introduction with the intention of demonstrating that the services will be implemented without disrupting the processes and projects developed in the organisation
- consideration of cultural aspects of the organisation and their influence on the service implementation process (it is important for the improvement of the process)
- demonstration of the risks and advantages of the service implementation.

To implement services, the following processes are used: Transition Planning and Support, Change Management, Configuration Management, Asset Management, Release and Deployment Management, Service Validation and Testing, Knowledge Management, and Evaluation.

The transition planning and support processes focus on ensuring the resources and their coordination in order to specify the services previously prepared within the SDP. The change management process includes the planning, monitoring and control of the compliance of changes with the business processes of the organisation and their influence on those processes. The aim of the asset management processes is to build and maintain organisation assets (databases) necessary for service performance. A *Configuration Management Database* is created for the purposes of service manufacturing. The database includes components – *Configuration Items* – required for service manufacturing. The next process – *Release and Deployment Management* – is intended to support the building of a packet of tested and implemented services. The packet also includes versions of services built on the basis of change specifications. Two processes include the validation and testing of services and their evaluation. Their purpose is to create conditions for the processes of verification and method selection as well as to perform the evaluation of delivered services with respect to their utility and warranty. An important aspect in the service implementation processes is the collecting and processing of knowledge of both the services and the processes of their implementation. Therefore, the knowledge management process has been distinguished. Its purpose is to support decision making, accumulating knowledge and making it available. A *Knowledge Management System* is created within the organisation, which in accordance with ITIL should be integrated with the previously discussed CMDB system.

Service Operation

After the implementation of services in the organisation, it is necessary to ensure the conditions for their operation. This is a key phase from the service client's point of view. It should ensure access to and coordination of all resources which are indispensable for using the services at the level agreed in the SLA. During the operation processes, bilateral requirements are important, relating to using services which generate conflicts. The sources of conflict may be: different points of viewing the IT by the client and the provider, the level of infrastructure stability at the moment of implementing the services and their resources, the quality of services provided and the service level. Awareness of the causes of conflict (knowledge obtained from the ITIL standard) may considerably reduce their occurrence.

Various events occur during service delivery related to registered changes in a service component which is indispensable in the implementation. They could be incidents, *i.e.* events of a negative nature whose effects can have an influence on the quality of services provided. They could also be problems which arise as a consequence of several incidents. In this situation, a change request may appear, which forms the basis for the change of a service component and in consequence the change management process is activated. As a result, a service request is made for performaning a standard operation, but not changing the existing service component.

For the operation of the implemented services the following processes are used: *Incident Management*, *Request Fulfilment*, *Access Management*, *Event Management*, and *Problem Management*. The service operation phase includes a definition of a set of functions which improve the discussed processes. The function in ITIL is defined as a set of activities supporting isolated processes in the organisation. Within the area of service operation, three functions are set apart: *Technical Management*, *Application Management* and *IT Operations Management*.

Incident management processes include procedures of incident effect removal and bringing the service to the level compliant with that established under the SLA. The procedures also include the recognition, register, coding and prioritising of incidents.

The purpose of the request fulfilment process is to handle requests for the services provided. The access management process uses security rules prepared by the IT. The event management process plays the monitoring, register and categorisation of events in the IT service environment.

The above-mentioned processes are supported with functions. The technical management function supports the planning and evaluation of changes in the IT infrastructure. The works are coordinated by a team which consists of administrators, consultants and IT specialists. The application management function refers to the use of knowledge of the system administrators and consultants with respect to providing services for applications. The function is supported by a relevant team. The IT operations management function is supported by an administrative unit, which normally provides support for the services. The unit performs two groups of functions: *IT Operation Control* and *Facilities Management*. Their purpose is to perform tasks related to monitoring systems and services supported by those systems. They also handle other administrative IT tasks, such as the administration of property necessary for IT operation.

Continual Service Improvement

In other phases, the support processes are similarly defined. Improvement of the performed services takes place at each phase of the service life cycle. From the ITIL point of view, the improvement includes the following processes:

- monitoring conformity with the processes and procedures of service preparation
- monitoring the effectiveness of services and processes
- monitoring the efficiency of services and processes.

In the case of service improvement processes, the Deming cycle is used, which presents a cyclic approach to service improvement. It includes the following stages:

- *Plan* – what should be prepared, how it needs to be prepared, who is to prepare it and when
- *Do* – performing the planned tasks
- *Check* – checking the effects of task performance
- *Act* – updating the action plan in accordance with the above-mentioned results.

Seven steps are defined in the cycle, which include:

- *define the process which should be measured*
- *specify what can be measured within the process*
- *collect data about the process*
- *process the data about the process*
- *analyse the collected data*
- *prepare and use information about the process*
- *perform the improvement procedure for the process*.

The cycle also includes two processes: *Service Measurement* and *Service Reporting*.

C.1.1 Service Support Processes

Before the start of architecture development within the ADM, the frameworks and basic design rules in the ADM need to be established, the purpose of which is:

- definition of limitations – primarily with regard to architecture development in accordance with the ADM
- preparation of a transition path and its monitoring (from the initial stage to the target enterprise architecture stage)
- specification of the evaluation criteria of architectures developed within the ADM and tools used in the development processes.

In order to achieve the above goals, a hierarchical and component-based approach to architecture development is assumed. The next step is the definition of enterprise architecture development principles and business rules (as the specification of principles of company activity is usually beyond the scope of the company's functionality and architecture).

As a rule, both areas blend with each other and that is why the business rules, according to which the company's strategic aims are fulfilled, are reflected in the enterprise architecture development principles. In the course of the management of such development within the ADM, the following need to be assumed:

1. the level of other architecture frameworks applied in the company (if they exist)
2. rules of building business strategies and knowledge of the company's business objectives
3. the management strategy for the IT which is used in the company
4. existing principles of enterprise architecture building or other IT development principles.

Familiarity with the level of existing solutions and IT strategies is the basis for defining the frameworks and adopting the principles for enterprise architecture building. To this end, universal methods of architecture building should be used, based on company experience and characteristics.

For a number of reasons, the scope of architecture development can be limited (possibility of implementation, availability of specialists, financial or time-related limitations and others). Four areas of limitations in the development process are defined (discussed in detail below):

- ✓ scope of enterprise
- ✓ scope of architecture
- ✓ granularity level of the used description
- ✓ time horizon for the ADM and its components.

- **Scope of enterprise** refers both to independent organisations gathered around the company and also to the company environment (with providers, clients and partners), and it includes areas which are included in the description (organisation activity sectors, system functions, activity areas, *etc*.). Taking into account the limitation in the scope of enterprise, two basic approaches are suggested for architecture building:
  - – vertical, when the scope of enterprise is divided "vertically", distinguishing 'segments': representing independent activity and own architecture (which includes all four above-mentioned assumptions and the idea of future integration) and separate implementation (to fulfil the company's particular objectives)
  - – horizontal, when the scope of enterprise is divided "horizontally" into four domains (in accordance with COBIT): PO (planning and organisation), AI (acquisition and implementation), DS (delivery), and ME (monitoring).

Organisations which implement architecture in accordance with the ADM pay attention to the hazard related to choosing the scope of enterprise which is too narrow (in both approaches: vertical and horizontal). They find that the development of enterprise architecture needs to be considered with a hierarchical strategy in mind (with the top-down approach), based on the hierarchical structure of four architectures (Section 2.5.1). Without this strategy, decisions related to the scope of architecture may be wrong (they may restrict the company's efficiency).

Experience gained so far shows that, considering time and cost limitations, it is not very realistic to create a complete enterprise architecture. This is why a recommended alternative for companies is to build a few different architectures.

Therefore, there is a need for the management and use of an 'aggregation' of architectures (a group of non-uniform architectures selected from the ADM, interconnected in different ways). A good starting point is to adopt rules of access to generic architectures. The architectural framework and the management framework are then considered, which include:

- **Scope of architecture:**
  - This can include all four architectural areas: business, data, applications and technology (or a part of those, with time and financial limitations). Since building a complete architecture in all four areas is not very realistic, architecture descriptions usually refer to specific objectives of the company or problems. The description provided helps to establish the role and place of a given area in the company. For example, if application building is important, a business architecture should be created. Due to continued changes this can be difficult, so it should be well thought-out, and building it without a description of the four architecture areas cannot be complete by definition. In this case, the implementation risk is high and there is little possibility for a deal between the architect and the company Management Board.
- **Granularity level of the applied description** (particularly in the context of work related to the design and manufacturing of the future IT system)**:**
  - Due care must be taken to ensure that architecture descriptions provide for future applications and are prepared at the similar granularity level: it is therefore important to include future applications of EA in the context of company objectives. If there are too many details, architecture building may exceed the allotted time and resources. A reasonably selected granularity level of the client's requirements specification should also be adjusted to the granularity level of the Continuum documents. There are other key factors in architecture building, such as the availability of resources and employees' competencies, and focus on the most important issues. Too wide a scope could limit the possibility of implementation of the ADM.

- **Time horizon** for the ADM and its components in relation to the adopted architecture vision:
  - If the adopted scope of enterprise is too wide, creating a target architecture vision for it could prove difficult. In this situation a more advantageous solution would be to perceive the architecture as a sequence of architectures developed in turn in a given time. The first of those is the main architecture of the company, and the following architectures (in the case of iterative EA development) present successive iterations of architecture visions. Aiming at a practical compromise, an intermediate solution is usually preferred, with a limited number of iterations.

The ADM method of enterprise architecture development proposed in TOGAF includes an iterative set of processes divided into cycles and phases. Previously prepared components-blocks from the company's repository are used. Before using the ADM, it is a good idea to check whether the available architectures which come from different sources can be considered in the integration or building of a generic architecture. When using it in the ADM processes, the time required for its building needs to be taken into account, including the number and scope of intermediate stages, as well as the use of the ADM (together with the generic architecture) for the purposes of the entire company. Consideration needs also to be given (within the main process) to the possibility of using other resources, previously manufactured with the use of the ADM method, and system frameworks and models, *e.g.* those which come from the Continuum. Fig. C.1 presents a typical ADM cycle for enterprise architecture development. In its manufacturing, three strategies are used:

- *top-down* including architecture development planning for an enterprise seen as a group of enterprises (one entity)
- *development* of the adopted general and dedicated architectures, typical for companies, groups of companies or branches
- *borrowed reference, i.e.* preparation and implementation of an architecture for a specific company and then copying it for the purposes of other companies.

The processes of adapting a universal architecture for the purposes of a company are called *Product Line Engineering*. It is recommended that the adaptation and pattern processes (in all phases of ADM) are performed in accordance with the ADM, which is of fundamental importance (through the availability control of resources, and planning the development processes on the basis of Governance processes) for IT design management (Fig. C.1).

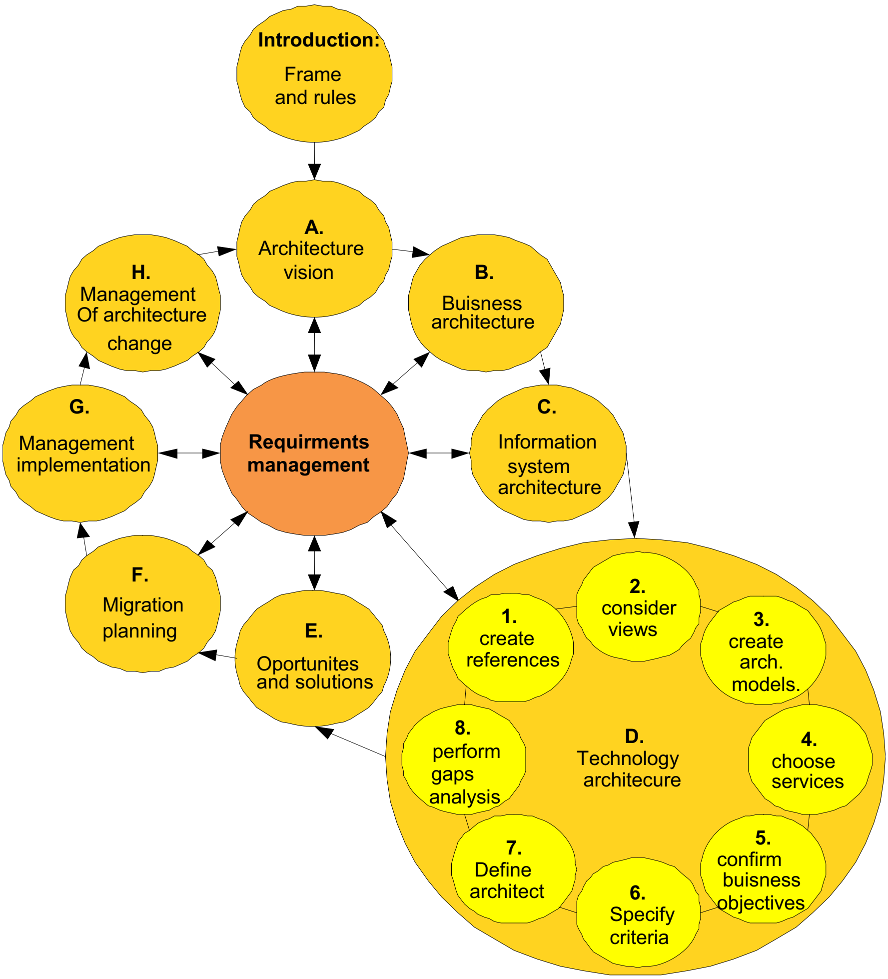


Fig. C.1 Enterprise architecture development cycle (TOGAF concept)

The solution presented should obviously be adapted to the specific functions and organisational conditions in the company. Special ADM processes are required for those purposes. It may happen that the sequence of the phases in the architecture development will be changed and (for the implemented IT systems) a business architecture will be created only after an application architecture has been prepared (in the case of improving business processes).

Adaptation of the ADM to the requirements of a company may be needed when:

- implementation of the ADM is one of the processes of creating enterprise architecture
- the ADM is a method of supporting the company's development

- enterprise architecture development has been contracted to an outsourcing company
- the enterprise is small or medium-sized (a hitch of adapting the ADM to its requirements)
- the enterprise is large and complex (dedicated architecture and manufacturing methods).

C.1.2 Service Support Tools

The implementation of enterprise architecture is possible with the use of supporting tools.

Using architecture frameworks, for instance, improves the architecture development process, creates a complete description of the future functioning of the system, and also ensures that the selected/prepared architectures will correspond to the business needs of companies. It is especially useful in the case of technically complex processes of a company.

An effective tool is the Zachman framework, which constitutes an external framework of the description of a company and its processes, and also of the existing IT systems. It categorises processes and systems by creating different levels of their description. It enables the recognition of IT resources (applications and data) which support individual processes in the company and their forecasts. However, the observation of transition processes or the assignment of persons responsible for them is not possible. To this end, the architect creates a communication network of people responsible for the manufacturing of interconnected artifacts: processes, data and applications. The construction of the Zachman framework can be supported (within the processes area) by the TOGAF standard, which organises the description of architecture development in the company by including its two main stages: initial level and target level. TOGAF enables the use of manufacturing methods (*e.g.* incremental-iterative), process management methods (hard or lean), and auditing methods for the level of IT systems usage (COBIT).

In companies, TOGAF [65] and the Zachman framework support EA development through a set of supporting methods and tools. Its first version was prepared by the US Department of Defence (DoD) in 1995. Its purpose was to set the framework for the manufacturing of technical architecture (technology architecture) for information management (TAFIM). Successive versions of TOGAF are prepared each year: they are published on the WWW site [65]. The current version 9.1 expands the descriptions of individual architectures of version 8.1.

When using the ADM, enterprise architecture is developed on four levels:

- *Business Architecture*, which includes the strategy for company activity and key business processes
- *Data Architecture*, *i.e.* its logical and physical models

- *Application Architecture*, which includes the applications functioning in the company that refer to business processes
- *Technology Architecture*, which includes the description of hardware and software on a logical level with reference to the above-mentioned architecture levels.

Evaluation and Selection of Target Enterprise Architecture

After discussing the issue of enterprise architecture development, we now present the method for the selection of target enterprise architecture, which consists of establishing the costs of changing the existing architectures or developing new architectures, and the costs of data exchange between the existing and newly created applications.

When selecting the target architecture, problems may be encountered which are related to:

- preparation of one user interface for the new and the existing applications
- access to data (it may happen that the new and old applications share data, for which a suitable sharing environment should be created)
- costs of software and hardware (which are caused by changing standards or the demand for increased computing power).

The evaluation and selection of EA are performed through the following processes:

- gap analysis between the existing functions and selected applications
- identification of main development processes and definition of the project environment
- classification of applications from the point of view of the possibility to purchase or develop or re-use applications which exist in the company.

Architecture development is related to the activation of management processes, such as:

- gap analysis evaluation in phase D (stage in Fig. C.1), *i.e.* differences between the created client views on enterprise technology (stage 2) and technology architectures defined (in a previous iteration) by the provider (stage 7)
- 'brainstorm' with regard to the technical requirements of applications and the cooperation and mutual use of the requirements
- analysis of the influence of the changes on project development
- planning and implementation of the migration strategy of applications.

As a result of the implementation of the development processes, relevant documents are created.

Manufacturing Documents

- target business, technology, data and application architecture
- viewpoints of technology architecture
- documentation of architecture blocks from the Continuum (if available)
- information on products available in the Continuum.

Management Documents

- client's declaration of the intention to manufacture IT systems together with the provider
- order for commencing work (received by the provider from the client).

C.1.3 Planning the Migration Processes

First, we shall define the priorities for development work, the main packets of work and the plans for migration to the new architecture and their development with respect to costs and income. We will do it on the basis of a detailed migration plan and a method of implementation.

Before the migration to the new architecture, the following questions need to answered:

- What are the implications of the current project for other projects and actions?
- What are the relationships between the projects?
- What products will be manufactured?
- Which components of those projects should be developed?
- Does the company have the resources required for the preparation of such components?
- What are the manufacturing standards for products or components?
- When will they be available?
- Are the products going to pass the test of time, both in terms of the technology used and the provider's profitability?
- What are the costs of additional training for users?
- What is the influence of IT culture on the user community; how can it be controlled?
- What are the total costs of and income from the migration, and its advantages for the organisation?

Answers to these questions in the context of the existing and future structures depend on the size of the organisation, its complexity, and also on the value of the new technology for the implementation of the basic functions in the organisation. When answering the questions, the advantages of using the existing systems and the risk related to changing the environment or the provider need to be considered.

Most companies are of the opinion that changing the architecture in one stage poses a high risk for the running of their organisation, one of the factors being the necessity to include many technical issues in implementing changes in IT systems. Other issues of migration are: the issue of the parallelism of the operations, the selection of approach (the entire system or selected functions), or the effect of the company's structure on the migration. Thus decisions taken in effect of those considerations should be included in the migration plan.

There are many migration strategies and plans for their implementation. It is best to focus on successfully completed long-term projects. An example of this approach are short-term projects where business functions are implemented – the manufacturing and activation of applications that process data for chosen business functions (so-called pilot applications).

The execution of the processes of migration between those systems (the application pilot and the full version) requires the following manufacturing and management processes to be defined:

(1) establishing the requirements for the IT system

(2) establishing the scheme for the migration of data to the target system (reading/update)

(3) improvement of data architecture development based on the use of the CRUD matrix (create – read – update – delete), prepared as part of data architecture

(4) specification of the company's income from project development

(5) assigning key project managers

(6) decision to remove the application pilot.

(1) Migration to the new system requires precise specification of the scope and structure of data.

(2) Migration to the target architecture is executed through the following management processes:

- assigning priorities to developed projects
- estimation of expenditure and implementation possibilities
- performing a calculation of costs/profits for individual projects of resource migration
- performing an evaluation of risk for migration to the target architecture
- preparation of a roadmap for plans of migration processes
- analysis of the impact of risk factors on project development.

(3) The CRUD matrix, which improves the design process, is described as a specification.

The term: CRUD matrix (*Create, Read, Update, Delete*) *is a table prepared for the purposes of the examination of the level of a company, where connections between processes and data and data sets and also between processes and their components are presented.*

(4) For establishing the income, a matrix is used which is built on the basis of project value indicators and descriptions of their risks. The indicators are based on the company's financial measures and they show its strategic position with respect to competition. Risk descriptions include such items as: the size and complexity of the project, technologies used, organisational possibilities, and their influence on the (lack of) success of the project. Each element is weighted separately. The indicators, criteria and weights are approved by the management.

(5) Which aspect is of higher implementation priority depends on the key managers: cost reduction, service consolidation or the ability to handle changes in temporary projects. The selection of implementation priorities is in the competences of the managers, and depends mainly on the size of the project and its suitability for the company's structure.

(6) The decision to remove the application pilot is made after the installation of the full version by business representatives (rather than the IT department).

The effect of the manufacturing and management processes are relevant documents:

- ✓ request for work commencement
- ✓ detailed migration plan and its implementation
- ✓ analysis of the impact of individual factors on project development
- ✓ target business, data, application and technology architecture.

Requirements Management for the Purposes of the Provider Organisation

Below we shall refer to the management of the requirements imposed on architecture development in the ADM processes. The ability to deal with changes in those requirements is of fundamental importance for the development of the ADM processes. The architecture development process should provide for the changeability of requirements as a result of the ongoing collaboration of project partners. Consideration should also be given to the limitations expressed in those requirements and processes, which result from the company functioning in a changing market reality, with new legal regulations, *etc*. (those which lead to changes in the requirements). Different standards can be used for the management of requirements processes. Even in TOGAF, there is not one specific approach imposed. However, scenarios are indicated there for acquiring business requirements. This way the process of acquiring and processing requirements is improved and changes can be incorporated smoothly.

The following processes are involved in the management of project requirements**:**

- preparation of output documents for each ADM stage
- compiling of high-level requirements, expressed by a part of the architecture vision, made up with the use of company functioning scenarios (or another way)
- preparation of specific requirements for a given domain and potentially also for other domains (where the architecture domain may require changes)
- mapping the ADM in current projects and new types of requirements
- analysis of structural differences between architectures in the individual phases of the development cycle and identification of differences between the initial and target architectures (certain gaps of this type may result in failure to fulfil the requirements)
- in the ADM, two types of gaps are described (with respect to requirements):
  - requirements which refer to initial architectures rather than target architectures
  - requirements which do not refer to initial architectures but to target architectures.

The principle of considering gaps in the requirements (although it only refers to the initial and target architectures) is related to the entire process of requirements acquisition (which is a continual process). It includes the identification and then documentation and registering of the requirements in the repository.

The acquisition and processing of requirements is performed in accordance with the manufacturing processes:

- identification/documentation of requirements (*e.g.* using business scenarios)
- preparation of basic requirements:
  - establishing the priorities for the current stage in the implementation of the ADM
  - confirmation by partners of the acquired requirements for the adopted priorities
  - entering the requirements and their priorities in the repository
- monitoring basic requirements
- specification of conditions for changes in the requirements
  - removing priorities or re-evaluating them
  - adding requirements and re-evaluating priorities
  - modification of the existing requirements
- specification of requirements and changing the assigned priorities:
  - specification of requirements by project partners and changes of their priorities by the architect responsible for the current phase
  - registering new priorities

  - ensuring the identification of conflicts and their management (in phase execution)
  - preparation of a statement, by the manager of the implementation team, of the effect of the requirements on the development process (* see comment below).
  - evaluation of the effect of changes in the requirements for the running stage
  - evaluation of the influence of changes in the requirements for the previous stages
  - proving whether changes are implemented or postponed till the next ADM cycle (before the implementation of change management, estimate the required time)
  - making a book of requirements
- implementation of requirements
- update of the repository of requirements related to changes (in the views of partners)
- implementation of changes in the current phase (performing an evaluation of the management level and a gap analysis in the final phases of the manufacturing cycle).

* **Note**: A change of requirements may occur in any phase. To ensure the correct assessment of them and the assigned priority, the process must be managed at the ADM phase levels and decisions referring to the requirements should be registered there. The establishment of the requirements management by the project partners makes development easier.

The following manufacturing documents are the effect of manufacturing processes:

- document of changes in the requirements
- statement of the impact of the changes on the development process compliant with ADM.

The repository of requirements includes requirements for the target architecture. When new requirements appear, it should be indicated which of them have been changed and the impact of the changes on the process of target architecture development must be shown. It is indispensable to prepare a statement where ADM development phases which are subject to change are listed. The preparation of change documents is iterative until the final version is obtained, which contains the full consequences (*e.g.* changes in costs, time schedule and business measures).

Target Architecture Implementation Management for the Purposes of the Provider

For an enterprise architecture adjusted to the purposes of a company, including the migration processes discussed above (Chapter 6 – Appendix 1), below we describe the method of its implementation, using the following management processes:

- formulation and execution of recommendations related to each project
- preparation of agreements for building an architecture which will regulate the processes
- using relevant management functions in the course of system implementation
- ensuring conformity with the architecture definition for the duration of implementation
- development of architectures in accordance with the adopted concept of EA.

According to TOGAF recommendations, information on the quality of architectures needs to be collected in order to ensure effective project management. This refers to information related to both the implementation of processes (planning of manufacturing) and their management. The target architecture implementation phase, executed as parallel to a company's functioning processes, cannot interrupt its organisation and development processes. Therefore, management connections are established which enable the implementation of the architecture in an operating company. The architecture manufacturing contract should assume full operation of the company.

An important aspect in EA implementation is ensuring consistent development (Fig. C.1). The effects of management processes are the following management documents:

- request for work commencement
- order document for EA manufacturing (with approval from all project partners)
- document describing the scope of the project and its impact on other projects
- document presenting the compliance of architecture development rules with the ADM
- proposal indicating the possibility for re-using blocks (from the Continuum)
- analysis of the influence of risk factors on project development
- transition plans and their implementation (including agreements for development)
- document of changes in the requirements, such as support for a standard interface
- review of the compliance of implementation management with the ADM processes
- document where recommendations for the project are formulated.

Change Management in the Course of Architecture Implementation

The scope of project changes can be wide or narrow. Below we consider the aspects of ongoing development related to major changes in newly-built architecture (as an effect of completing the previous phase). The aim of phase H

(Fig. C.1) is to organise the processes of change management, the purpose of which is to introduce and monitor changes (*e.g.* during the implementation of new technologies or changes in the business activity environment). In this regard, the expected changes refer to the project framework and the rules specified in the architecture vision document (phase A). The architect should ensure the dynamic execution of changes in ADM processes by using:

- monitoring of technological and business changes
- evaluation of changes in the company and adopting processes for architecture development
- discussion of architecture development by the Management Board in order to make decisions regarding the way to handle *major changes* with respect to technology and business processes in a situation when:
    - ✓ the architecture of the enterprise or its parts are changed
    - ✓ the enterprise architecture development cycle is restarted.

The management processes of architecture changes are closely related to the processes of requirements management in the company. In phase H (Fig. C.1) minor changes are made (often outside an agreement) and major changes, which require renegotiating an agreement. The execution method of the ADM processes confirms the organisation's maturity (its repeatable and defined processes). The factors taken into consideration for the evaluation of provider maturity are: implementation of new technologies of report generation, management of cost components, technology changes, and implementation of new IT standards. These factors are usually analysed with the use of existing procedures of change management. Other factors, such as the company's strategic changes, new repeatable business processes or technological innovations require the architecture development cycle to be restarted.

There are a number of approaches, methods and techniques for change management (*e.g.* the PRINCE method of project management, or ITIL as an evaluation standard for IT organisations). Below we describe two approaches to the management of minor changes:

- simplification change: can be performed with the use of change management techniques
- incremental change: the introduction of gradual changes can be managed with the use of change management techniques or lead to the partial involvement of the architect depending on the nature of the changes.

In order to establish the correct method for treating changes (simplification, incremental change, or major change), the following actions are taken:

- registering all requirements which may affect the architecture
- constant management of resources and task architecture
- assigning responsibility for the architecture when making changes
- evaluation of the effect of the changes made (on the development process).

So the scope of changes and the scope of implementation of the architecture development cycle are very important for a company. The following rules are recommended (as per TOGAF):

- If the change is related to considering the influence of two or more partners, the architecture should probably be designed once again and the ADM should be adopted.
- If the change is related to considering the influence of only one of the partners, it is more probable that change management procedures will be used.

A number of examples which demonstrate the use of the above rules can be given:

  - If the effect of the changes is significant with reference to the business strategy, it is suggested that enterprise architectures should be re-implemented and the architecture should be designed again with the use of the ADM.
  - When implementing new technologies or safety standards, the technology architecture should be refreshed.
  - If the change occurs in the infrastructure (*e.g.* there is a major change in the system or several systems are reduced to one system), it is recommended to limit the changes in the architecture to the physical layer and the basic description of the technology architecture (a change in management techniques could also be suggested).

A partial or complete change of the architecture can be implemented if:

- training is conducted on ways of adapting the architecture to the business strategy
- a substantial change of components is performed (in this case instructions need to be prepared and the components need to be deployed in the company's architecture)
- standards are used in the implementation of architecture changes which have a significant impact on the end user (*e.g.* in the case of regulatory changes).

If partial or complete changes are made, it is a good idea to publish them for the purposes of another cycle in the form of the following management documents:

- request for architecture change with the specification of the change technique:
  - *New Technology Reports*
  - initiatives related to the management of cost reduction
  - reports related to technology withdrawal
- request for architecture changes with the specification of the impact of the changes on:
  - business development

    - exceptions with regard to business processes
    - business innovations
    - technology innovations
- business strategy
- architecture update
- changes in development principles and architecture framework
- new request for architecture design with the use of another manufacturing cycle.

C.1.4 Governance Processes

When providing services for the purposes of the client organisation, Governance processes are used, which are described in the main part of the book. Their detailed specification has already been discussed. Below we present supplementary material, which may be used in extending the MITM model or in building an environment for the verification of project negentropy.

The term: SLA services (*Service Level Agreement*) *are a type of services performed for the purposes of an organisation – after signing an SLA document between the partners, which specifies the conditions for their performance, i.e.*:

- *ensuring that all significant information related to architecture implementation included in the contract is published and made available to authorised parties*
- *validation of the requested services, limitations adopted by project partners, costs of project development, etc.*
- *ensuring the mechanisms of formal acceptance and the approval of arrangements related to architecture development* (*publication in project documents*)
- *providing elementary methods of project development control for the effective implementation of the architecture in compliance with the ADM*
- *creating and maintaining relationships between the goals included in the architecture, the strategies to be used, the implementation and the company's strategic objectives*
- *identification of departures from the ADM cycle, and*
- *plans for remodelling the ADM cycle without including certain requirements (that were agreed by project partners) or ensuring a different update policy.*

Management of the Processes of Architecture Development Conformity with the ADM

Table C.1. shows the main roles in the process of EA development. The implementation of conformity processes is conducted in accordance with the steps presented in Table C.2.

Table C.1 Main roles in the process of architecture development

Number	Role	Responsibilities	Comments
1	Management Board/team	Ensures consistency of structures and supports business needs.	Sponsors the ADM and monitors the work performed.
2	Project manager	Responsible for the entire project.	
3	Architecture verification coordinator	Administers the architectures and their development as well as the review processes.	Activity-oriented rather than technology-oriented.
4	Lead Architect	Ensures consistency of architectures.	Architecture specialist.
5	Architect	One of the key functions, supports technical assistants.	
6	Client	Makes sure that business requirements are clearly expressed and comprehensible.	Manages the preparation of the company description within the architecture.
7	Business domain expert	Makes sure that the processes of business requirements fulfilment are justified and comprehensible.	Knows how business domains work. Can also be the client's representative.
8	Project decision makers	Ensure sufficiently precise access of architects to the client and data from company domains or experts.	Members of client organisation who are involved in the acquisition of business requirements.

Table C.2 Steps in the implementation of conformity processes

Number	Action	Comments	Who
1	Request for architecture evaluation.	In compliance with rules and procedures.	Any person or company familiar with the activity
2	Specification of the proper part or organisation and adoption of the main projects.		Architecture verification coordinator
3	Employing architects and defining EA problems.		Architecture verification coordinator
4	Defining the scope of the overview.	Identification of involved units/departments. System control within the enterprise architecture.	Architecture verification coordinator
5	Adopting control measures	For the inclusion of business requirements	Lead Architect
6	Review meeting and evaluation of time schedule execution.	Cooperation with requirements architect	Architecture verification coordinator
7	Interview with the principals of the project	In order to obtain technical information and performance conditions.	Lead Architect and/or architect, project manager, clients
8	Analysis of complete checklist.	Recognising and solving problems. Establishing recommendations.	Lead Architect
9	Preparation of a report on compliance with the architecture.	Using supporting personnel.	Lead Architect
10	Preparation of conclusions of this overview.	Intended for the client and Management Board	Lead Architect
11	Approval and signing of the review.		Management Board and client representative
12	Sending the evaluation report and summary to the architecture verification coordinator.		Lead Architect

Overview of Architecture Development Management Methods

The management of architecture development constitutes a set of *good practices*, which are used for the planning and control of the enterprise architecture development process, which include:

- implementation of a system of control over the manufacturing and monitoring of all architecture elements and actions, whose purpose is to ensure the effective introduction, implementation and evolution of architecture in the organisation
- manufacturing of a system in order to ensure compliance with internal and external standards and current regulations
- management processes, which effectively support manufacturing processes
- growth of practices related to the internal and external responsibility of project partners.

TOGAF Architecture Management

In the development of architectures within the ADM, apart from the Governance processes, classical management processes are also used. An example of such processes are the management processes for architecture manufacturing contained in the Continuum, which include: the planning, monitoring and control of architecture development. The processes should be supported by Governance procedures (which facilitate the identification of business processes and the clarification, transfer and effective management), which include:

- corporate governance principles
- applied management techniques
- architecture development management.

The management processes constitute a universal framework which can be adapted to the existing conditions of company management.

Each of those management areas can be found on various levels in a company – global, regional and local – within its general infrastructure. Governance, therefore, is a long-lasting process which extends beyond the scope of a company's activity and frameworks such as the TOGAF architecture, and it should be considered in the context of the entire organisation including the hierarchy of management structures. The analysis of the process leads to the explanation of the meaning of a few description problems related to it:

- management in the context of architecture development including IT technology
- Governance framework, which can be put to practical use.

The acquisition of requirements from the client, and IT implementation, and their support with management technologies are of strategic importance for the company but are encumbered with a major risk. In project development, organisations are more and more dependent not only on IT but also on the effectiveness of the implementation of project management mechanisms, manufacturing processes and supporting technologies. Thus a company's activity within its architectures results in an increase in its profitability (depending on the level of the use of management processes and their technological support), its reputation and its value.

Description of the Governance Framework

Governance processes (see Section 3.4.1), while ensuring the framework and structures of links between resources and information in order to fulfil the company's objectives and strategy, should refer primarily to the rules of the operation and control of environments and the supervision of company information (considering growing bureaucracy with regard to electronic reports and the risk of data loss). Governance processes are commonly accepted for company management. Together with suitable technologies, they constitute useful support, the purpose of which is also to identify the owner of the IT resources who should implement the development processes and bear the responsibility for the integration of architectures in the company.

Governance Framework for Architecture Development – Approach

The development and management of architecture is expressed by a number of processes realised as a part of organisation culture. It also forms a set of competences which guarantee the consistency and effectiveness of architecture implementation in a given organisation (Fig. C.2).

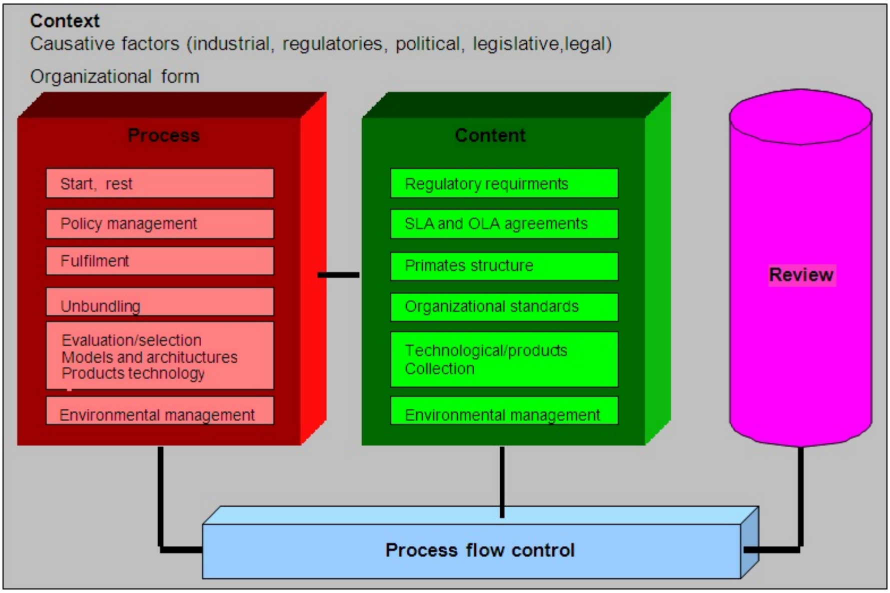


Fig. C.2 Governance framework for architecture development – approach diagram

The division of the Governance process, its contents and context are essential for the support of architecture management by enabling new (legal, regulatory and standard-based) actions which support management. Such an approach makes the TOGAF framework more flexible.

The Governance framework for architecture development is an integral part of the Continuum. It forms the basis for all processes significant for the architecture and the management of development processes. It is indispensable for the identification, control and distribution of information related to the management of architecture development, orders and their realisation. The Governance framework is used for monitoring architecture components, agreements and principles on the operation level (as well as the taking and verification of decisions on that basis). Any changes in architecture design, prepared agreements and additional data can be conducted in accordance with Governance procedures (register, verification, approval and publication). This makes integration with existing documents easier, and also simplifies current management and control of partners, documents, contracts, *etc*. Compliance of the performed tasks with SLAs and OLAs (*Operational Level Agreements*) is assumed.

The term: OLAs services (*Operational Level Agreements*) *are a type of support processes for the construction and use of services performed for the purposes of an organisation after the document which defines the conditions for their performance is signed by partners.*

It is also necessary to ensure compliance on the level of standards and regulatory requirements, which will be monitored on a current basis in order to achieve the monitoring of stability, conformity and efficiency.

The evaluations obtained are analysed and accepted or rejected, depending on the criteria adopted for the management. They are rejected if the service design processes or technology selection are not compliant with the assumptions. In this situation, the technology needs to be adjusted and services redesigned in order to fulfil the requirements. If the conformity evaluation is negative, an alternative action plan is to adopt a temporary conformity. This is granted for a given period and for a given set of services, with defined criteria that must be fulfilled in the specified time. The mechanism guarantees the quality of operation services and the flexibility of dates for their performance.

The Governance framework also defines the services for the functioning of a repository, which is the basis for the use of the management framework contained therein. It includes the use of physical and logical resources, ensures access and communication, and training for all participants of architecture development. The management environment adopted for EA development also includes administration processes used in service management, which include user management and internal SLAs (defined in order to control own processes).

Organisational Structure for the Governance Framework and Its Main Areas

The management of architecture development is a set of good practices, with which a company's architecture and other architectures are planned and controlled. In order to achieve effective control, relevant change monitoring mechanisms need to be activated, both at the level of manufacturing and management of architecture development. In the control process, the existing management processes and their suitability for an organisation's structures as well as the ability to develop architectures also need to be considered. The organisational structure presented in TOGAF and the Governance framework form an example of applying management processes to any company with different requirements (Fig. C.3).

The organisational structure of Governance processes includes the management of three key manufacturing processes: architecture development, the implementation of its processes and deployment. For each of them, one or more people responsible for their execution are assigned. To support them (component manufacturing) throughout the entire architecture life cycle, Continuum resources are used and recommendations, processes and structures related to the ADM contained therein (within the main process). The fact that Governance processes form an integral part of the Continuum brings the following advantages:

- IT processes, resources and organisational information are in line with the strategic goals
- manufacturing frameworks are ranked as (per COBIT): planning and organisation, acquisition and performance, delivery, support and monitoring of IT efficiency
- easier for a firm to use own information, infrastructure, hardware, software and resources
- integration of best practices, like requirements for verifiability, security and responsibility
- promoting risk management.

In order to execute a Governance approach to the management of architecture development, it needs to be ensured that:

- best practices are used with respect to submitting, receiving and re-using reports, generating procedures and tasks, using competences, organisational structure and its support in the form of services
- tasks and organisation structures are established for the support of architecture management processes and reporting instructions
- tools and processes are integrated for easier execution of processes (procedural, cultural)
- management is provided with regard to the selection of control processes for architecture development, requirements omission procedures and the evaluation of compliance with SLAs and OLAs
- meetings are held concerning the internal and external requirements for the processes of information, services and process management.

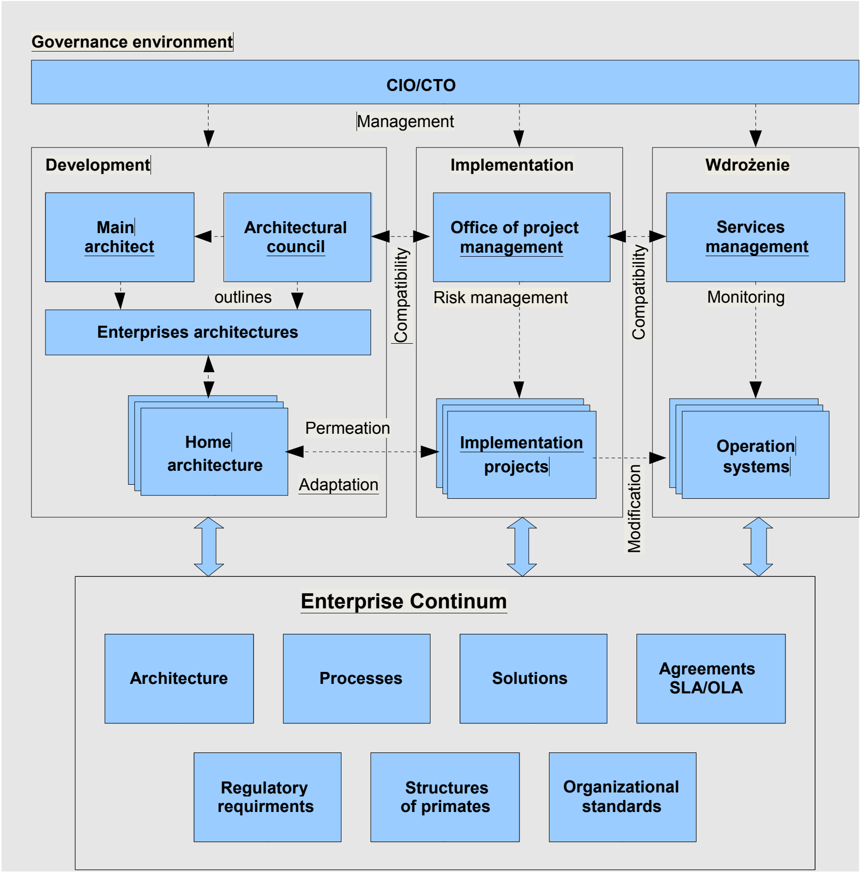


Fig. C.3 Governance framework – organisational structure

For effective enterprise architecture manufacturing, suitable strategic support needs to be prepared. A company's architects (if they are not engaged in the development of the business strategy) should show at least basic understanding of business problems faced by the company. There are two important elements in the management strategy of an architecture project which refer to architecture manufacturing within a company:

- creating a team gathering representatives of the whole cross-section of the company to support the management and supervise the implementation process of IT management strategy
- presenting a complex set of architecture development rules which should form a guidebook that shows how the organisation realises its mission through IT.

Data Management

In order to fulfil the development objective, which is architecture building, significant company data needs to be considered. To define it, the following questions need to be answered:

- Are there definitions of the data model and data structures, and have application installation options been purchased?
- What are the principles of defining and observing the requirements related to data and patterns for modules of the existing IT systems?
- Is the repository used for acquiring architectures and information on the processes of their development?
- What is the definition of the physical model of data (derived from logical models) that is used (in database design)?
- Have tools for software manufacturing and data management been chosen?
- What metadata has been identified (in the course of architecture development)?

In the course of architecture development (apart from defining the data) effective security and availability measures need to be demonstrated, with consideration to the following issues:

- What are the area and attributes of the data and the rules of access to it?
- What procedures are used to prevent unauthorised access?
- What are the mechanisms of data protection with respect to access control from external and internal sources?
- What are the management procedures and update principles for physical data deployed on different platforms?
- What are the replication management procedures for the data of operational activity?
- Which layers in the data server have been assigned to storing data contained in the designed data warehouse?
- Has a DBMS system been implemented?

The term: DBMS system (*Database Management System*) *is a system of database management (most DBMS systems operate within a client-server architecture*).

Good Practices Related to Manufacturing Processes

These rules form essential guidelines which provide support for the performance of tasks and strategy in the company (architecture development). They are an organised (in the form of a glossary) set of regulations whose object is to obtain the expected architectures, from requirements to development processes within the ADM.

Depending on the type of organisation, the rules may be established on three levels:

- company level: such rules are common in government and non-profit organisations; they also appear in companies as a source of processes which support decision making (*e.g.* they are a key element in the strategy of successful architecture management)
- IT technology: the rules form guidelines for using and implementing all IT resources and assets in a company (they are prepared in order to evaluate the effectiveness of the manufacturing environment for IT)
- architecture: as a subset of IT rules, which refers to the ADM, they reflect the level of cooperation of various groups in a company and present the principles which regulate:
  - the architecture design (the effect on its development, maintenance and application)
  - architecture implementation (which describes initial assumptions and guidelines for the design and development of IT systems).

First we will define the architecture development rules which refer to the implementation and use of IT resources and other assets of the company. They reflect the level of cooperation between the company's elements and constitute the basis for decision making. Each of the rules should be clearly related to key business objectives and be justified providing the consequences of its application. Therefore, in order to be able to define the rules precisely, recommendations are followed which refer to their format and they include:

- a name related to the essence of rule management rather than the technology platform
- a concise and clear statement of the defined rules
- a justification which emphasises the business advantages of using the rules
- effects of use, with emphasis on the requirements, both for the business and the IT.

C.1.5 Architecture Development Principles

Architecture development principles are prepared by the lead architect in cooperation with the CIO and the manufacturing team (and key partners). The principles of implementation should provide for general IT rules and principles on the company level (if such exist). They are selected in such a way that the IT strategy is adjusted to the strategy and vision of the company. To provide more detail, architecture development principles should include:

- the mission and mission plan of the company and its plans for organisational infrastructure development
- the current status of the company's strategic initiatives: a description of its weak and strong points, opportunities and risks

- external limitations: market-related factors (client expectations, *etc.*), existing and potential legal status
- current principles, procedures, systems and technology: the set of resources in the company, including the systems of documentation, equipment inventory, network configuration diagrams
- tendencies in IT: forecasts for its use, availability, costs of using the equipment and communication technologies.

The set of principles is expressed in a language understood by business applications. The principles should be few, future-oriented and approved by the senior management. They must present a solid basis for decision making with respect to the planning and design of architectures, creating behaviour, procedures and standards, and also support for actions in difficult situations. Unsuitable principles will not be used (and therefore they will not be reliable). There are four criteria which enable recognition of whether principles are useful:

- understandable – their basic assumptions are understood across the organisation: They are clear and straightforward, and their violation is minimal.
- precise – creating conditions for decision making related to architectures and their development plans, and also to realising the tasks and standards which are to be created: Each of the principles should be sufficiently precise to support consistent decision making in complex, potentially controversial situations.
- consistent – the set of principles has to be consistent and enable correct interpretation: The principles should not be mutually exclusive (strict observance of one principle may give ground for invalidating other principles, or their inconsistent interpretation).
- stable – the principles should be lasting, but modifiable: The process of making changes should provide for addition, removal or modification of the principles (after an approval).

Using Principles for Architecture Development

Architecture development principles take into account the knowledge of how the company will use its resources. This knowledge can be used in a number of ways:

- by providing a framework in which the company can make deliberate decisions on IT
- by establishing relevant criteria for the selection of products or product architecture in later stages of conformity management (with reference to IT architecture)
- by acting as a controller to specify the functional requirements for the architecture
- by contributing to the evaluation of both the existing systems and future strategies with regard to compliance with the architecture definition: the evaluation provides information required for taking actions to implement the architecture and business priorities

- by justifying the value of the architecture and its manufacturing
- by presenting an outline of the most important tasks, resources and potential costs
- by providing support for architecture management including the following actions:
  - adopting the project strategy for architecture development
  - supporting rejecting decisions: if changes cannot be included in a local procedure.

The principles can sometimes compete with each other; for instance, access and security principles can suggest contradictory decisions. Therefore, they should be treated as coordinate.

Example Architecture Development Principles: Business Processes of a Company

Adopting too many principles results in the reduced flexibility of the architecture development process. Many organisations limit their number to 10 - 20. The example below shows a typical set of architecture development principles as well as recommendations for defining their format.

Principle 4.1: Priority Principles

Thesis 4.1. Information management principles refer to all organisations in a given company.

Justification for use: In this way a consistent and measurable information system is obtained for decision makers. Lack of this principle undermines the procedures of information management.

Conditions:

- information and risk management initiatives are not activated until they are examined with regard to their compliance with the principles
- actions contradictory to the principles will be discontinued.

Principle 4.2: Maximisation of advantages for companies

Thesis 4.2. Decisions related to information management are made in order to maximise the advantages for the company as a whole.

Justification for use: This principle realises the idea of 'services beyond the scope of requirements of individual company departments'. Decisions made from the perspective of the whole company are more important than those which only take into account the points of view of individual departments. So in order to achieve a higher yield, information needs to be managed (due to the established priorities) at the company level. This way, partial requirements or the interests of small groups will not dominate the architecture development process.

Conditions:

- gaining advantages in the whole company requires change management
- technology itself will not force changes
- organisations often suggest their own solutions without analysing their effect on them
- the applied architecture development priorities should refer to the entire company
- architectures should cover the entire organisation
- information management initiatives should comply with the company development plan
- it is recommended that the initiatives are managed (due to the company plans/priorities)
- a change of priorities should refer to the complete representation of the company.

Principle 4.3: Analysis of company business processes requires information management

Thesis 4.3. The participation of all organisations in the company in acquiring and processing information and in decision making is indispensable for achieving business objectives.

Justification for use: Information obtained from users is needed by partners (clients) for the proper use of IT technologies in the company. Company experts and technical personnel in charge of the development and maintenance of the information environment, should be included in the team responsible for formulating IT objectives.

Conditions for use (that such a team can function properly) concern each partner, which:

- assumes responsibility for the development of the company's IT environment
- engages their resources.

Principle 4.4: Continuation of Business Activity

Thesis 4.4. Company activity should be sustained despite breaks in system operation.

Justification for use: In order for company activity to be comprehensive, the quality of the existing systems needs to be considered in the categories of structure and functionality. The company should obviously fulfil its business functions regardless of external events. Neither an equipment failure, nor a natural disaster should cause a disturbance or halting of company activity. In the event of problems, the company's business functions should have an action strategy. Continued operation of the company is ensured by using alternative activity options (*e.g.* additional data centre).

Conditions:

- operation risk has to be established in advance and properly managed.
- management processes need to include periodic audits, testing and evaluation of system vulnerability or critical processes referred to business services, *etc*.
- processes of restoring/maintaining the system should be included (in the design)
- applications should be evaluated with respect to their impact on company operation.

Principle 4.5: Shared use

Thesis 4.5. The manufacturing of applications in a company is aimed at its development.

Justification for use: Ability to manufacture repeatable applications are expensive and conducive to generating inconsistent data.

Conditions and effects:

- a suitable policy is needed for approaching the requirements which were not included in the manufactured application
- data and information supporting the entrepreneurial activity of the company should be standardised
- the standardised resources (data and information) should be processed together.

Principle 4.6: Compliance with legal acts

Thesis 4.6. Information management processes should be compliant with regulations and policies.

Justification for use: The adopted policy of a company should be compliant with legal requirements, external policies and other regulations. This does not rule out the possibility for the improvement of business processes, which implicate changes in company operation.

Conditions:

- an organisation's activity with respect to the collecting, storing and management of data needs to be adjusted to legal requirements and external policies
- rules of access to this activity need to be established.

Principle 4.7: IT responsibility

Thesis 4.7. IT organisations are responsible for the implementation of processes and for an IT infrastructure which enables the incorporation of solutions whose purpose is to fulfil the user-defined requirements at the level of functions, services, costs and dates.

Justification for use: It is expected that in consequence of using the IT responsibility principle clients' requirements will be adjusted to the possibilities and costs of project development, which will result in creating effective enterprise architectures.

Conditions:

- the process has to be updated in accordance with project priorities
- IT functions have to reflect business processes
- in the development process four integrated models are to be used: of data, business processes, applications and technology (see Section 2.4).

Principle 4.8: Protection of intellectual property

Thesis 4.8. The intellectual property of the company has to be protected and reflected in the IT architecture and in the implementation and management of processes.

Justification for use: Protection of intellectual property has a significant impact on the activity of the *host* in the IT domain of the company.

Conditions:

- protection of the intellectual property of company assets is actually provided in the IT area (regulations related to copyrights with respect to software use)
- IT security policy and relationships between IT employees and other employees of the company require the protection of intellectual property to be monitored.

Principle 4.9: Data is the company's asset

Thesis 4.9. This is one of three closely connected principles related to data. Data is shared and easily accessible. It constitutes a value (asset) of the company and needs management.

Justification for use: Data (especially its accuracy and timeliness) is the basis for effective management. It is an important asset in the company (of measurable value). Therefore, careful data management is necessary (it needs to be known where data is, how accurate it is, how it is obtained and where and when it is needed).

Effects of use:

- shared data is easily accessible
- metadata (data on data) ensures high quality of data

- metadata can be critical in architecture development – outdated, false or inconsistent data can stem from it.

Conditions:

- the main purpose of training during architecture building is to ensure that all organisations in the company know the relationship between the value of data and making the data available
- persons in charge need to assign who will be authorised to manage data
- a transition needs to be made from ‘data ownership’ to ‘data management’
- procedures for preventing and removing data errors need to be prepared and applied
- data quality has to be measured
- steps need to be taken to improve quality
- the provider team should independently make the decision concerning changes in own production processes suggested by the client team
- data of brokers responsible for data management must be submitted to the client team.

Principle 4.10: Company data is shared

Thesis 4.10. Users have access to data necessary for performing their duties, therefore data is made available in the entire company and its organisations.

Justification for use: Quick access to accurate data is essential for the improvement of the quality and effectiveness of decision making in a company. It is a less expensive approach to maintaining updated, accurate data in one application and its division than storing (or duplicating) data in several applications. The company uses data redundancy and it is stored in non-conformant databases. The rate of collecting, creating, transmitting and absorbing data by an organisation should go hand in hand with the ability to take effective action to make data available.

Making data available will result in better decisions, since we will use fewer sources of more accurate and updated data for the purposes of our decision process. Making it available will lead to an increase in the company’s efficiency.

Effects of use:

- data is available to all users to perform their duties
- only the most accurate and current data is used for decision making
- data available in the entire company constitutes one data environment
- the principle of data exchange is contradictory to the principle of data security (under no circumstances, however, can it result in making confidential data available).

Conditions:

- development and application of shared sets of rules of conduct, procedures and standards regulating access to data and its management in the short and long term
- investing – short-term – in software which enables data migration and the system of its legalisation (in order to support the existing systems)
- preparation of standard data models, data elements and other metadata which defines the shared environment
- building of a metadata storing system in the repository (to make data available)
- adopting and enforcing – long-term (replacement of existing systems) – a shared data access policy and requirements for new applications
- ensuring data availability in new (shared) environments and applications
- adopting common methods and tools for the manufacturing and maintenance of and access to data available in the company (short- and long-term)
- changes in the company's operating culture (making data available).

Principle 4.11: Data is always available

Thesis 4.11. Data is available for all users to perform their functions.

Justification for use: Wide access to data leads to increased efficiency and effectiveness of decision making processes and to better use of service delivery applications. The use of data and enabling access to it for a large number of users is a strategic aim from the point of view of the organisation.

Effects of use:

- data availability is connected with the ease of users obtaining information
- access to data does not affect the way it is understood.

Conditions:

- the means of accessing data and its display have to be sufficiently flexible, so the needs of various users can be met
- decisions should be made carefully (with respect to the correct interpretation)
- access to data should be distinguished from a user's right to its modification or disclosure
- training and changes in the organisation's culture, within which processes of intellectual property (data) protection by the provider/client organisation are important.

Principle 4.12: Responsibility for data quality

Thesis 4.12. Data is managed by a person responsible for the quality and integrity of data.

Justification for use: One of the advantages of a functioning architecture is the possibility to make data available (*e.g.* a text, video, audio, *etc.*) in the entire company. Since the availability scope grows with the increase of IT importance in a company, companies adopt the strategy of the shared use of data. A consequence of this could be the loss of data integrity, which happens when data is distributed and there is no access control. Therefore, persons planning and controlling access and implementing data have to bear full responsibility for its integrity. Thus the need to engage the entire team is eliminated and data resources are limited (to the size required). Integrated data is less vulnerable to redundancy.

Principle 4.13: Using a common dictionary and definitions of data

Thesis 4.13. All data is well defined for the entire company, and its definitions are comprehensible and available to all users.

Justification for use: Data used in the prepared application must have a common definition in the entire company in order to enable data exchange. A common dictionary makes communication easier and creates conditions for building a system interface and data exchange.

Effects:

- although the improvement of a data storage and processing environment is a separate task, creating a data definition facilitates its use across the company.

Conditions:

- significant additional resources are required to perform this task
- the company has to establish a common vocabulary used across the company
- data standardisation initiatives must be coordinated
- correction of data definitions must be disseminated across the company
- each new definition must be compliant with the corporate data dictionary
- a data administrator in the company is responsible for the above coordination
- data administration responsibilities must be assigned.

Ability to Design Architecture Frameworks

While describing the basic data related to details of architecture building, it is a good idea to provide the participants of the building process with a set of skills, functions, standards and experience of company employees. The presented description should include a set of skills as per the specified competency levels intended for the individual functions:

- roles performed in the company
- skills required for each of the roles
- level of knowledge required to perform the assigned roles.

Sets of competencies can include groups of common skills, required in the case of consulting or the management of project tasks or the development of a specific project or its packet. The sets are also widely used by recruitment agencies or search engines for finding and choosing suitable candidates for given functions. Having such sets makes it possible to quickly identify gaps and select candidates and roles (especially with a company's immaturity for architecture development).

Basic Practices in Architecture Building

Despite the lack of uniform terminology, architecture development skills are extremely important in the development of those architectures and they gain more and more significance in the IT field. Numerous companies have created environments which assemble practices in IT architecture development treating them as a means to stimulate the development of the required skills and experience of employees with respect to assuming the tasks of an enterprise architect. The practices are subject to a formal program of development and certification, on the basis of which companies formally recognise the skills of IT architects. Such a program is indispensable for adjusting the skills and experience of employees connected with IT architecture development to a company's needs. Skills and definitions related with individual roles should also ensure that recruitment organisations will provide employees suitable for working on enterprise architecture (architecture development is difficult and expensive). An IT architecture development team is assembled through mutual verification and requires time and skills from the company's management. Also needed is the mutual evaluation of the team, process documentation and internal certification, as well as time for candidates to prepare for the mutual verification by creating their own portfolios, and demonstrating their skills and professional experience.

The term: Work portfolio – *a set of studies, documents of a given member of the manufacturing team which enables the evaluation of his/her attainments and skills.*

In TOGAF, the skills of designing architecture frameworks include the necessity to define the role of the architect at different progress levels of architecture development. Due to the complexity, time and costs, companies often do not have their own internal program of IT architect certification and prefer to use interviews about their work or ad hoc hiring of employees. This approach carries a serious risk, because:

- communication between recruiting organisations, consulting companies and recruitment agencies is very difficult
- interviews with candidates for positions, conducted in good faith, do not show lacking skills and/or experience required
- employees who are capable of performing functions related to architecture development may remain unnoticed.

Despite acting in good faith, there is a risk that unsuitable personnel will be employed, which may lead to:

- increase in costs of labour, through the need to replace the personnel
- lower quality of provided IT systems and project development.

The certification of architects responsible for architecture development is needed. Therefore, a company creates an internal program of such certification in order to:

- formally recognise the skills concerning the experience of IT architects as part of the task to establish and maintain a professional trade organisation of IT architects
- adjust the skills and experience of the personnel, with respect to tasks related to IT architecture development, to the company's needs.

By using a certification program companies can reduce the time, costs and risk connected with training, hiring and managing IT architecture specialists (inside and outside the company). Skills related to designing architecture frameworks provide for levels of competencies for individual roles in the provider team which develops architectures. In TOGAF, the following roles are included:

- sponsors
- IT Managers
- architects designing enterprise architectures:
  - ➢ program or project managers
  - ➢ IT developers
  - ➢ team members.

In the Tables presented below (C.3, C.4, C.5, C.6, C.7, C.8, C.9, C.10) the required skills and their desired level are presented with the use of a scale (for each of the roles). The main task of an IT architect is detailed analysis of the needs and principles. It is often the case, however, that the roles of IT architect, IT developer and the person responsible for implementation are a little unclear. Some skills attributed to architects are also required for developers. In general they are complementary, since IT creators focus mostly on the techniques and architecture components used. The following categories of skills are distinguished:

- general, which include: leadership, teamwork, personal skills, *etc*.
- related to the application of methods: usually including usage analysis, business processes, strategic planning, *etc*.
- related to architecture development: usually including modelling, design, application building, systems integration, *etc*.
- project management: typically business-oriented, including change management, project method and tools management, *etc*.
- related to the use of general IT knowledge: usually including interoperability of applications, asset management, migration planning, SLAs, *etc*.
- technical: usually including software engineering, data security, data management, *etc*.
- uses of the legal environment: usually including protection of personal data, contract law, public procurement law, *etc*.

Tables C.3-10 presented below assign roles and the desired skill level to each skill. In TOGAF [64] the skills of designing architecture frameworks are defined on four levels indicated in Table C.3.

Table C.3 Roles in IT architecture development and skills definitions

Level	Accomplishment	Description
1	Background	This skill is not required – nevertheless, it is to be specified and managed as required.
2	Awareness	Sufficient understanding of a problem, context and implications to take further steps and advise the client.
3	Knowledge	Detailed knowledge of a given area and ability to provide expert advice and use it in an architecture development project.
4	Mastery	Extensive practical experience and use of knowledge of the area in question.

Table C.4 General skills

Roles of IT architect	Member of Architecture Council	Architecture sponsor	IT architecture manager	Technology architecture manager	Data architecture manager	Application architecture manager	Business architecture manager	Program or project manager	IT developer
Areas of expertise									
General skills									
Leadership	4	4	4	3	3	3	3	4	1
Teamwork	3	3	4	4	4	4	4	4	2
Interpersonal skills	4	4	4	4	4	4	4	4	2
Oral communication	3	3	4	4	4	4	4	4	2
Written communication	3	3	4	4	4	4	4	3	3
Logical analyses	2	2	4	4	4	4	4	3	3
Project partners management	4	3	4	3	3	3	3	4	2
Risk management	3	3	4	3	3	3	3	4	1

Table C.5 Methods used

Roles of IT architect	Member of Architecture Council	Architecture sponsor	IT architecture manager	Technology architecture manager	Data architecture manager	Applications architecture manager	Business architecture manager	Program or project manager	IT developer
Business skills and methods									
Business case	3	4	4	4	4	4	4	4	2
Business scenario	2	3	4	4	4	4	4	3	2
Organisation	3	3	4	3	3	3	4	3	2
Business process	3	3	4	4	4	4	4	3	2
Strategic planning	2	3	3	3	3	3	4	3	1
Budget management	3	3	3	3	3	3	3	4	3
Creating the vision	3	3	4	3	3	3	4	3	2
Business metrics	3	4	4	4	4	4	4	4	3
Corporate culture	4	4	4	3	3	3	3	3	1
Investment in existing systems	4	4	3	2	2	2	2	3	2
Business functions	3	3	3	3	4	4	4	3	2

Table C.6 Architecture development skills

Roles of IT architect	Member of Architecture Council	Architecture sponsor	IT architecture manager	Technology architecture manager	Data architecture manager	Applications architecture manager	Business architecture manager	Program or project manager	IT developer
Enterprise architecture skills									
Business process modelling	2	2	4	3	3	4	4	2	2
Business process design	1	1	4	3	3	4	4	2	2
Designing roles	2	2	4	3	3	4	4	2	2
Designing organisation	2	2	4	3	3	4	4	2	2
Designing data	1	1	3	3	4	3	3	2	3
Designing applications	1	1	3	3	3	4	3	2	3
System integration	1	1	4	4	3	3	3	2	2
IT business standards	1	1	4	4	4	4	3	2	3
Designing services	2	2	4	4	3	4	3	2	2
Designing architecture principles	2	2	4	4	4	4	4	2	2
Designing architecture views and points of view	2	2	4	4	4	4	4	2	2
Designing building blocks	1	1	4	4	4	4	4	2	3
Solution modelling	1	1	4	4	4	4	4	2	3
Analysis of advantages	2	2	4	4	4	4	4	4	2
Business cooperation	3	3	4	3	3	4	4	3	1
Retaining systems	1	1	4	4	4	3	3	3	2
Project management	1	1	3	3	3	3	3	4	2

Table C.7 Project management skills

Roles of IT architect	Member of Architecture Council	Architecture sponsor	IT architecture manager	Technology architecture manager	Data architecture manager	Applications architecture manager	Business architecture manager	Program or project manager	IT developer
Program or project management									
Program management	1	2	3	3	3	3	3	4	2
Project management	1	2	3	3	3	3	3	4	2
Business change management	3	3	4	3	3	3	4	4	2
Change management	3	3	4	3	3	3	4	3	2
Value management	4	4	4	3	3	3	4	3	2

Table C.8 Skills related to the application of general IT knowledge

Roles of IT architect	Member of Architecture Council	Architecture sponsor	IT architecture manager	Technology architecture manager	Data architecture manager	Applications architecture manager	Business architecture manager	Program or project manager	IT developer
IT knowledge skills									
Methodologies and IT application manufacturing tools	2	2	3	4	4	4	2	3	3
Programming languages	1	1	3	4	4	4	2	2	3
Intermediate applications	1	1	3	3	4	4	3	2	3
Information consumer applications	1	1	3	3	4	4	3	2	3
Information provider applications	1	1	3	3	4	4	3	2	3
Mass memory management	1	1	3	4	4	2	2	2	3
Networks	1	1	3	4	3	2	2	2	3
Web services	1	1	3	3	4	4	2	2	3
IT infrastructure	1	1	3	4	3	2	2	2	3
Asset management	1	1	4	4	3	3	3	2	3
Service level agreements (SLAs)	1	1	4	4	3	4	3	2	3
Systems	1	1	3	4	3	3	2	2	3
COTS	1	1	3	4	3	4	2	2	3
Enterprise Continuums	1	1	4	4	4	4	4	2	3
Migration planning	1	1	4	3	4	3	3	2	3
Management tools	1	1	3	2	4	4	2	2	3
Infrastructure	1	1	3	4	3	4	2	2	3

Table C.9 Technical skills.

Roles of IT architect	Member of Architecture Council	Architecture sponsor	IT architecture manager	Technology architecture manager	Data architecture manager	Applications architecture manager	Business architecture manager	Program or project manager	IT developer
Technical skills in IT									
Software engineering	1	1	3	3	4	4	3	2	3
Security	1	1	3	4	3	4	3	2	3
System and network management	1	1	3	4	3	3	3	2	3
Transaction processing	1	1	3	4	3	4	3	2	3
Deployment and catalogue	1	1	3	4	4	3	3	2	3
User interface	1	1	3	4	4	4	3	2	3
International operations	1	1	3	4	3	3	2	2	2
Data exchange	1	1	3	4	4	3	2	2	3
Data management	1	1	3	4	4	3	2	2	3
Graphics and pictures	1	1	3	4	3	3	2	2	3
Operating system services	1	1	3	4	3	3	2	2	3
Network services	1	1	3	4	3	3	2	2	3
Communication infrastructure	1	1	3	4	3	3	2	2	3

Table C.10 Skills of using the legal environment

Roles of IT architect	Member of Architecture Council	Architecture sponsor	IT architecture manager	Technology architecture manager	Data architecture manager	Applications architecture manager	Business architecture manager	Program or project manager	IT developer
Legal environment									
Contract	2	2	2	2	2	2	2	3	1
Protection of data	3	3	4	3	3	3	3	2	2
Orders	3	2	2	2	2	2	2	4	1
Unstable environment	3	3	3	3	3	3	3	3	1
Commercial law	3	3	2	2	2	2	3	3	1

Roles and Skills of IT Architects

It architects are visionaries, trainers, team leaders, links between business and technical solutions, IT engineers and experts in the given field. Below we present an example description of an effective IT architect's work [67].

"An architect is obliged to ensure completeness of architecture (suitability for purposes) with respect to all important issues and consider the concerns of partners, seeing the integrity of the developed architecture as an imperative." The selection of architecture building views, and in particular its development processes (ADM), is a key decision made by an IT architect. "An IT architect's role comes down to planning on the company level, rather than the branch level. The planning process is always connected with the company's business plans, and the project decisions made need to be monitored against the plan to fulfil business objectives." [67].

The IT architect's strategic plan concerns the management of the architecture development process, thus project decisions are considered on the tactical level. The IT architect prepares project documents in order to make decisions regarding the project, team development, or product applications. The architect is involved in the entire process, starting from working with the client in order to understand their actual needs, through the whole implementation process (fulfilment of those needs). The architect should be familiar with various models of communication with the client so the needs can be met (he/she becomes an important participant of the sales process). However, he/she is not the implementer, and should present general solutions. This is an opinion presented by [63]: "The architect's role is to focus on a few key details and interfaces which really matter, rather than on all architecture elements."

Therefore, the architect's role can be summed up as follows:

- understanding and interpreting requirements
- creating a useful presentation model for those requirements
- verification, improvement and development of the model for the requirements
- management of the support development of a generic architecture in the Continuum.

When faced with difficult problems, additional architects should be employed. Each member of such a team should have a specific role and responsibilities. The team should consist of:

- strategic architect, responsible for the architecture design and technical documents – he/she should manage the group of system and/or industry architects involved in the project, with consideration to business functions on the company level
- system architect, responsible for architecture design and documentation on the level of the system or subsystem, including management and security – he/she should analyse only technological solutions, such as data warehouses in the company, and have support from the organisational architect (so he/she doesn't need to focus on details of those systems, products and/or technologies)
- field architect, responsible for architecture design and documentation for a given field – he/she should cooperate with the system architect
- organisational architect, responsible for architecture design and documentation of specific organisations and for the acquisition of data from all other architects – his/her main goal is (on the company level) to prepare domain-based business solutions (finance, human resources, sales).

The most important qualities of an IT architect are:

- skills and experience of creating patterns:
  - should be familiar with the techniques of incorporating patterns in the manufacturing of complex systems, including requirements for the acquisition of requirements and analyses, the preparation of solutions in the context of other IT systems, identifying alternative solutions and their evaluation, the selection of technology, development methods and configurations

- extensive technical knowledge of one or several areas:
  - should have solid technical knowledge (supported by experience) in the IT field, related to the development and implementation of applications as well as creating and maintaining infrastructure in order to support applications in a complex IT environment (this includes heterogeneous IT environments, distributed systems and traditional mainframe systems)
- using the MDA approach to architecture development:
  - should have knowledge of many development methods, and use them appropriately – which should manifest in project development by repeatable effective use of various development methods
- extensive experience:
  - should be familiar with various aspects: design, development, examination, manufacturing and implementation – enabling the fulfilment of the objective which is the practical implementation of the system
- technical, communication, team-building and leadership skills:
  - should be seen by the IT organisation as a leader in the company
- personal and professional qualifications, negotiation and problem-solving skills:
  - should be able to communicate on complicated technical issues with project partners (including people who do not have such knowledge)
  - should control business and manufacturing processes
  - should be able to cooperate with companies.

C.2 Summary

Appendix 3 constitutes a compendium of knowledge on provider organisation maturity. Its main purpose was to compile information on enterprise architecture development processes (on the basis of the TOGAF standard) and on support processes for IT systems architecture development (in accordance with the ITIL standard). The enclosed material formed the basis for the study of initial processing procedures (described in Chapter 3) and work on the general MITM management model (Chapter 2). In addition, the material can be used by provider organisation managers. In order to increase its usefulness in those processes (initial processing, modelling and management), the contents of the appendix was divided into parts which describe maturity.

In part 1 (Sections C.2.1 and C.2.2), ITIL materials are enclosed which relate to delivery and support processes indispensable for operation and change in provider organisations. In Chapter 2, devoted to the IPP processes of organisation maturity, the focus was on those processes which are important from the point of view of a model. In this Appendix, data related to the remaining ITIL processes (versions 2 and 3) has been collected and organised. The material, processed for the purposes of the model described in the book, supports the project manager in making decisions concerning changes in his/her organisation. Tools have also been presented which support both the ITIL standard processes (CRUD matrix) and TOGAF (Zachman framework and the ADM), important for organisation managers.

Next (Sections C.2.3 and C.2.4), migration processes in architecture development were discussed, which form an example group of processes that require support from ITIL. Both the specification of migration processes and the support processes for the ITIL standard provide the organisation manager with knowledge on the method of using the ITIL/TOGAF standards. The migration and support processes presented in this Appendix are an introduction to Governance processes, whose importance for architecture development was discussed in Section 4.3. Since the place and role of those processes are not always clear for the managers of provider organisations, the examples presented in the appendix explain the role of Governance processes and demonstrate their significance at the current stage in IT development (not only for the purposes of the ITIL and TOGAF standards).

In the final part of this Appendix (Section C.2.5), we discussed the business principles of architecture development which provide organisation managers with business justification for the modelled development processes (just like the Governance processes). They also constitute a summary and generalisation of knowledge on the application of standards (ITIL and TOGAF).

D

Appendix
Examples of the Scalarisation of Vector Quantities

Consider the example of a two-dimensional vector. Let us consider the negentropy:

$$\mathbf{p}_t = \begin{bmatrix} hT_t \\ he_t \end{bmatrix} \tag{D.1}$$

$\mathbf{p}_t$ – two-dimensional evaluation of the negentropy of an IT project

hT_t – variable of the level of the applied technologies in the management of an IT project (in short, ‘technology’), ($hT_t \in <0, 5>$)

he_t – variable of the level of knowledge and understanding of the project domain (in short, ‘personnel’), $he_t \in <0, 5>$

t – project realisation time, $t \in <1, t_{end}>$

t_{end} – project completion time.

Where what is needed is the simple scalar evaluation p_t from the range <0, 5> , one of several possibilities of scalarisation can be used.

D.1 The Euclidean Metric

One of the simplest evaluations of the negentropy vector is its length, defined as its Euclidean norm. If we assume that the values of both of the components of the negentropy vector belong to a closed interval of real numbers <0, 5>, then this scalar measure of the negentropy vector will be located (approximately) in the interval (1.41, 7.08). Therefore, an additional normalisation (scaling) is necessary, which will provide the same <0, 5> range of the scalar measure of negentropy. As a result of such operations we obtain the quadratic operator, the function:

$$Q(\mathbf{p}_t) = sqrt\left((hT_t^2 + he_t^2)/2\right) \tag{D.2}$$

which allows the scalar evaluation of negentropy to be established as:

$$p_t = Q(\mathbf{p}_t) \tag{D.3}$$

This operator is expressed in a graphical form in the following Figure.

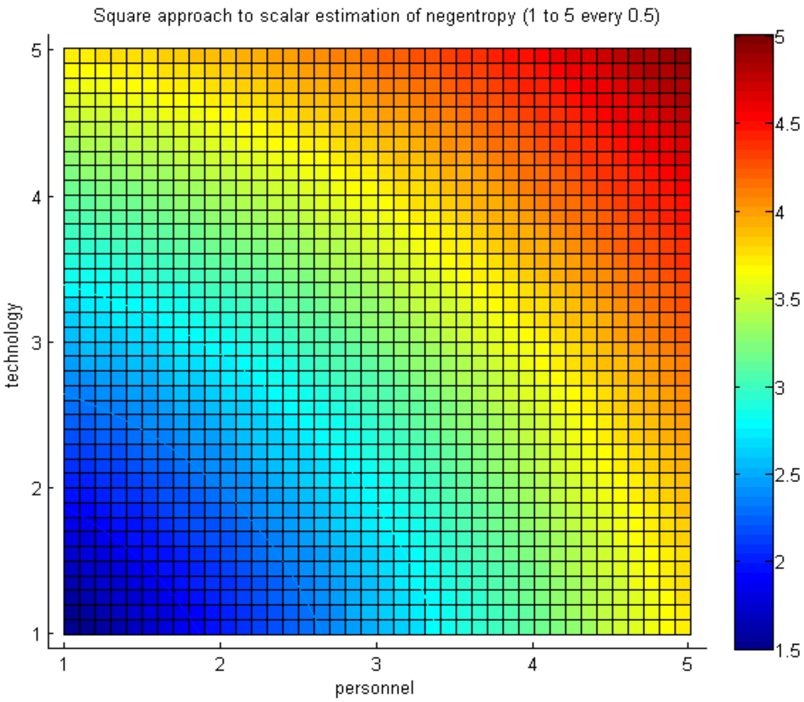


Fig. D.1 Euclidean two-dimensional conversion of negentropy

D.2 Equilibrium Evaluation

Another useful operator involves a simple equilibrium evaluation:

$$Q(\mathbf{p}_t) = (hT_t + he_t)/2 \tag{D.4}$$

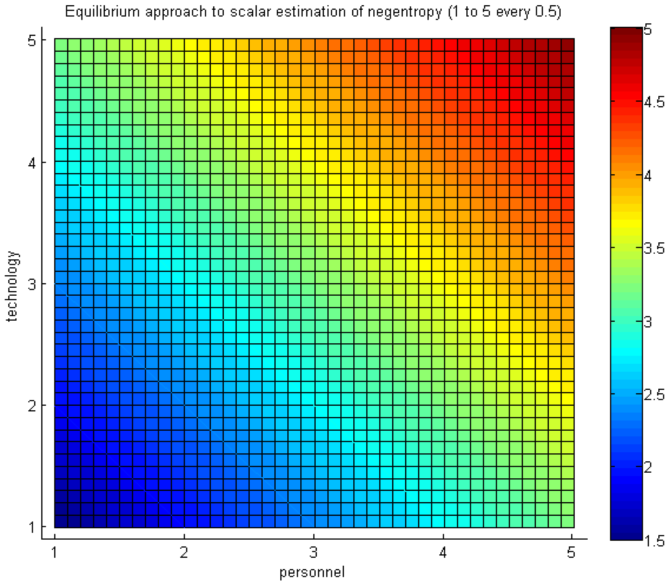


Fig. D.2 Equilibrium conversion of two-dimensional negentropy

D.3 Max Operator

In an optimistic case, the max operator can be used:

$$Q(\mathbf{p}_t) = \max\{hT_t, he_t\} \tag{D.5}$$

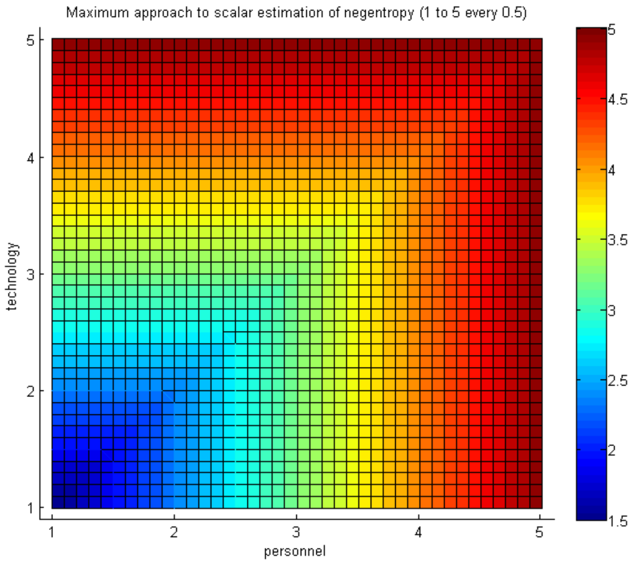


Fig. D.3 Maximum scalarisation of two-dimensional negentropy

D.4 Min Operator

A very cautious approach can be modelled with the use of the minimum operator:

$$Q(\mathbf{p}_t) = \min\{hT_t, he_t\} \tag{D.6}$$

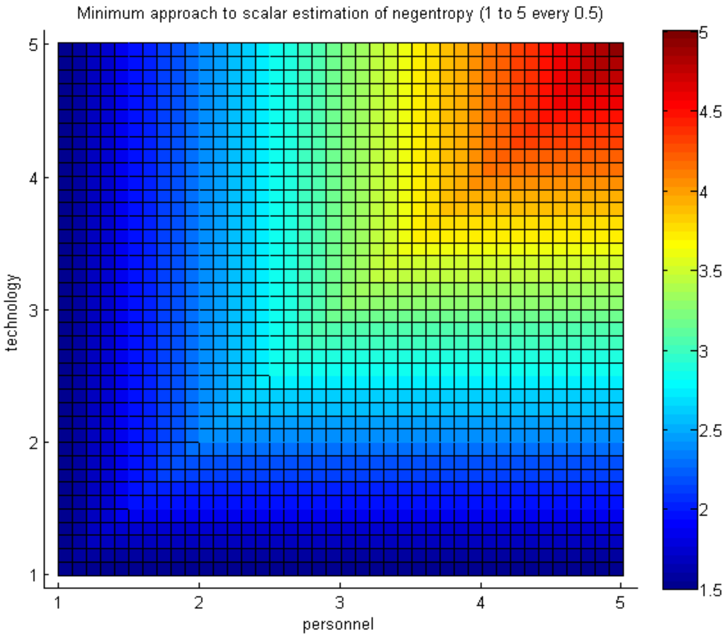


Fig. D.4 Minimum scalar evaluation of two-dimensional negentropy

D.5 Quadratic Weighted Scalarisation

An intermediate/weighted approach is possible referring to a quadratic operator:

$$Q(\mathbf{p}_t) = sqrt(\alpha \cdot hT_t^2 + \beta \cdot he_t^2) \tag{D.7}$$

With standardised weight factors α and β, so that $\alpha + \beta = 1$. An example of an output, shown in the Figure, is obtained when $(\alpha, \beta) = (0.3, 0.7)$.

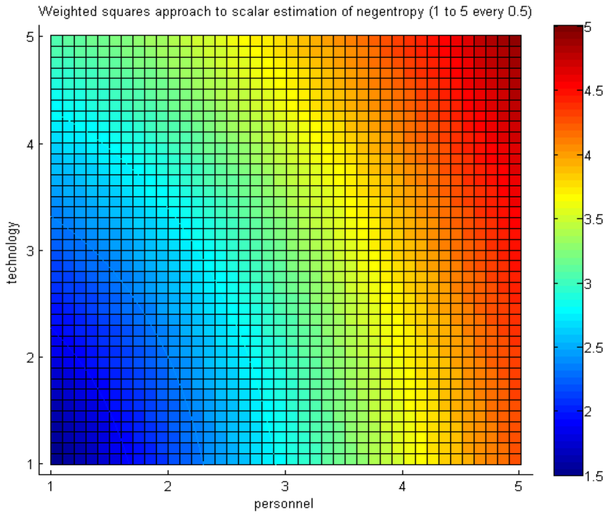


Fig. D.5 Quadratic weighted evaluation of two-dimensional negentropy

D.6 Weight-Linear Scalarisation

Or the linear approach:

$$Q(\mathbf{p}_t) = \alpha \cdot hT_t + \beta \cdot he_t \tag{D.8}$$

which, with identical weights (α, β) = (0.3, 0.7), leads to the following scalar evaluation.

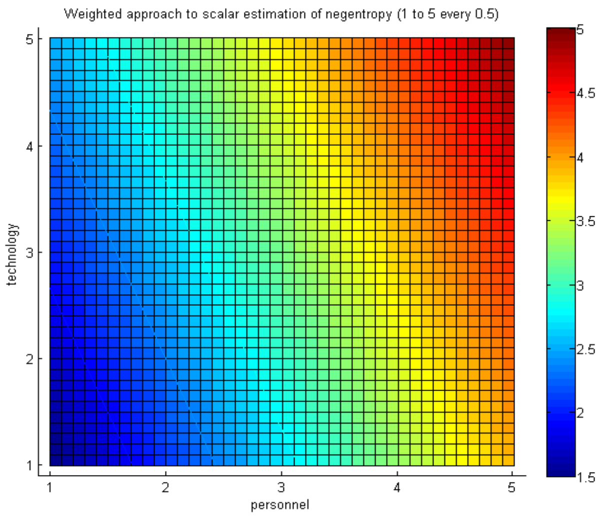


Fig. D.6 Weight-linear estimation of two-dimensional negentropy

Of course, each time, the selection of the weight factors is a sensitive issue, and in general – with no specific experience in this respect – such unequal weighting should be avoided, adopting variant 1 or 2 (or the limit ones of 3 or 4).

References

1. Barys, P.: The use of expert systems for modeling the evolution of services in IT support organizations, (MSc thesis in Polish). Gdańsk University of Technology, Gdańsk (2010)
2. Bass, L., Clements, P., Kazman, R.: Software Architecture in Practice. Addison-Wesley, Boston (2003)
3. Bińczycka, K.: Analysis and evaluation of an IT system in the company Energa SA (MSc thesis in Polish). Gdańsk University of Technology, Gdańsk (2010)
4. Boehm, B.W., Papaccio, P.N.: Understanding and controlling software costs. IEEE Transactions on Software Engineering 14(10), 1462–1477 (1988)
5. Clements, P., Bachmann, F., Bass, L., Garlan, D., Ivers, J., Reed, L., Merson, P., Nord, R., Stafford, J.: Documenting Software Architectures: Views and Beyond. Addison-Wesley, Boston (2010)
6. Cooper, K., Torczon, L.: Engineering a Compiler, 2nd edn. Morgan Kaufmann, San Francisco (2011)
7. Cyra, Ł., Górski, J.: SCF - A framework supporting achieving and assessing conformity with standards. Computer Standards and Interfaces 33(1), 80–95 (2011)
8. Grzech, A., Rygielski, P., Świątek, P.: Translations of service level agreements in systems based on service-oriented architectures. Cybernetics and Systems 41(8), 610–627 (2010)
9. Gunia, K.: Testing an IT system (MSc thesis in Polish). Gdańsk University of Technology, Gdańsk (2008)
10. Jerbic, M.: Enterprise Architecture & the Security Architect – Business imperatives on Governance. In: Proceedings of the Enterprise Architecture Practitioners Conference, pp. 47–56. The Open Group, San Diego (2007)
11. Kamiński, P., Frankowski, J.: Semantically-enabled SOA governance to improve BPM lifecycle abstract. In: Proceedings of Bird 2008 The Seventh International Conference on Perspectives in Business Informatics Research, pp. 36–47. University of Gdańsk, Gdańsk (2007)
12. Karembergen, W., De Haes, S.: IT alignment and value creation. Research Journey into Enterprise Governance of IT, Business 3, 1–13 (2010)
13. Kowalczuk, Z., Orłowski, C.: Model and fuzzy system management of project teams pursuing IT projects and the evaluation of the manufacturing processes of knowledgebased systems. Studia z Automatyki i Informatyki 27, 21–52 (2002) (in Polish)

14. Kowalczuk, Z., Orłowski, C.: Design of knowledge-based systems in environmental engineering. Cybernetics and Systems 35, 487–498 (2004)
15. Kowalczuk, Z., Orłowski, C.: Fuzzy decision support system for the control of budget and schedule in SPM. In: Cader, A. (ed.) Artificial Intelligence and Soft Computing, pp. 133–138. Academic Publishing House EXIT, Warszawa (2006)
16. Kowalczuk, Z., Orłowski, C.: Knowledge management based on formal models of software project management. Cybernetics and Systems 38(5-6), 611–626 (2007)
17. Łagowski, J.: ITIL v3 – The life cycle of IT services. In: XIV PLOUG Conference, Warsaw, pp. 138–150 (2004) (in Polish)
18. Manouvrier, B., Menard, L.: Application Integration: EAI B2B BPM and SOA. Wiley-ISTE, New York (2008)
19. Mędrak, P.: Testing and assessment of information technology in the company of the electrical industry (MSc thesis in Polish). Gdańsk University of Technology, Gdańsk (2008)
20. Nguyen, N.T., Szczerbicki, E.: Intelligent Systems for Knowledge Management. Springer, Berlin (2010)
21. Nguyen, N.T., Trawiński, B., Jung, J.J.: New Challenges for Intelligent Information and Database Systems. Springer, Berlin (2011)
22. Orłowski, C.: Fuzzy Model of IT Project Management. Publisher of Gdańsk University of Technology, Gdańsk (2004)
23. Orłowski, C.: Knowledge management and fuzzy modelling in software engineering. In: Kubiak, B., Korowicki, A. (eds.) Information Management, pp. 58–63. Publisher of Gdańsk University, Gdańsk (2005)
24. Orłowski, C.: Management of changes in software engineering. In: Grzech, A., Wilimowska, Z. (eds.) Proceedings of the 26th International Scientific School on Information Systems Concepts, Tools and Applications Information Systems Architecture and Technology (ISAT), pp. 128–135. Wrocław University of Technology, Wrocław (2005)
25. Orłowski, C.: Methods of building knowledge-based systems applied in software project management. In: Leondes, C. (ed.) Intelligent Knowledge-Based Systems: Business and Technology in the New Millennium, vol. 1, pp. 207–245. Kluwer Academic, Boston (2005)
26. Orłowski, C.: Change management and fuzzy modeling in software engineering. In: Scientific Papers of the Faculty of ETI. Information Technology, vol. 8, pp. 873–882. Gdańsk University of Technology, Gdańsk (2005)
27. Orłowski, C.: Information technology used to support the management of IT projects. In: Orłowski, C. (ed.) Managing Information Technology: IT Application Examples, pp. 85–94. PWNT, Gdańsk (2007) (in Polish)
28. Orłowski, C. (ed.): Managing Information Technology: IT Application Examples. PWNT – Pomeranian Science and Technology Publishers, Gdańsk (2007) (in Polish)
29. Orłowski, C., Karpiński, P.: Decision support and digital libraries in organisation transformation. In: Wilimowska, Z. (ed.) Information Systems Architecture and Technology: Application of Information Technologies in Management Systems, pp. 93–100. Publishing House of Wrocław University of Technology, Wrocław (2007)
30. Orłowski, C., Kowalczuk, Z.: Modelling of IT project management based on fuzzy sets. Zarządzanie Przedsiębiorstwem 7(2), 55–65 (2004) (in Polish)
31. Orłowski, C., Kowalczuk, Z.: Project management in enterprises: IT implementation based on fuzzy models. International Journal of Enterprise Information Systems 2(2), 1–12 (2006)

32. Orłowski, C., Kowalczuk, Z.: Knowledge management based on dynamic and self-adjusting fuzzy models. In: Gabrys, B., Howlett, R.J., Jain, L.C. (eds.) KES 2006. LNCS (LNAI), vol. 4251, pp. 671–678. Springer, Heidelberg (2006)
33. Orłowski, C., Orłowski, A.: Rational Portfolio Manager – Environment to support the management of information technology projects. In: Orłowski, C. (ed.) Managing Information Technology: IT Application Examples, pp. 95–101. PWNT – Pomeranian Science and Technology Publishers, Gdańsk (2007) (in Polish)
34. Orłowski, C., Sitek, T.: Decision-making in IT projects supported by systems based on knowledge. In: Grzech, A. (ed.) Knowledge Engineering and Expert Systems, vol. 2, pp. 373–382. Publishing House of Wrocław University of Technology, Wrocław (2006) (in Polish)
35. Orłowski, C., Szczerbicki, E.: Application of blackboard architecture in concurrent environment for IT project management. In: Knosala, R. (ed.) Computer Integrated Management, vol. 2, pp. 249–254. WNT – Publisher of Science and Technology, Warszawa (2004)
36. Orłowski, C., Szczerbicki, E.: The problems of design and evaluation of project organizations implementing IT projects. In: Knosala, R. (ed.) VIII Conf. on Computer Integrated Management (Zakopane, Poland), vol. 2, pp. 224–232. WNT –Publisher of Science and Technology, Warszawa (2005) (in Polish)
37. Orłowski, C., Szczerbicki, E., Kowalczuk, Z. (eds.): Management of Information Technology and Knowledge. PWNT –Pomeranian Science and Technology Publishers, Gdańsk (2008)
38. Orłowski, C., Ziółkowski, A.: Evolution of concept of agent system for information technology evaluation. In: Grzech, A. (ed.) Information Systems Architecture and Technology: Information Systems and Computer Communication Networks, pp. 25–36. Publishing House of Wrocław University of Technology, Wrocław (2008)
39. Patterson, D., Hennessy, J.: Computer Architecture: A Quantitative Approach, 4th edn. Morgan Kaufmann, San Francisco (2006)
40. Pravin, G.: SOA Case studies & semantic interoperability. In: Proceedings of the Enterprise Architecture Practitioners Conference, pp. 332–340. The Open Group, Bangalore (2008)
41. Radhakrishnan, R.: EA and ITIL – Implications of ITIL-ITSM framework & IT processes for EA development. In: Proceedings of the Enterprise Architecture Practitioners Conference, pp. 127–135. The Open Group, San Diego (2007)
42. Sitek, T., Orłowski, C.: Evaluation of information technologies – concept of using intelligent systems. In: Wilimowska, Z. (ed.) Information Systems Architecture and Technology: Application of Information Technologies in Management Systems, pp. 217–224. Publishing House of Wrocław University of Technology, Wrocław (2007)
43. Sitek, T., Orłowski, C.: Knowledge base in a multi-agent system for the assessment of information technology. Sharing of knowledge and knowledge bases. In: Knosala R (Ed.): Computer Integrated Management, vol. 2, pp. 409–415. Publishing Houseof Polish Society for Production Management, Opole (2008) (in Polish)
44. Sitek, T., Orłowski, C.: Model of management of knowledge bases in the information technology evaluation environment. In: Wilimowska, Z. (ed.) Information Systems Architecture and Technology: Models of the Organisation's Risk Management, pp. 221–231. Publishing House of Wrocław University of Technology, Wrocław (2008)

45. Sitek, T., Orłowski, C.: Management model of knowledge resources in an information technology assessment environment. In: Orłowski, C., Szczerbicki, E., Kowalczuk, Z. (eds.) Management of Information Technology and Knowledge, pp. 403–412. PWNT – Pomeranian Science and Technology Publishers, Gdańsk (2008) (in Polish)
46. Sommerville, I.: Software Engineering, 8th edn. Pearson Education, Harlow (2007)
47. Sobczak, A.: Ontology-based requirements representation in the process of building information systems. In: Proceedings of the Second AIS SIGSAND. European Symposium on Systems Analysis and Design, pp. 34–40. Department of Business Informatics, University of Gdańsk, Gdańsk (2007)
48. Szczerbicki, E., Orłowski, C.: Knowledge management: towards smart IT support system. In: Knosala, R. (ed.) Computer Integrated Management, vol. 2, pp. 521–527. WNT – Publisher of Science and Technology, Warszawa (2006)
49. Szczerbicki, E., Orłowski, C., Kowalczuk, Z. (eds.): Foundations of Control and Management Science, vol. (9). Poznań University of Technology, Poznań (2008)
50. Szczerbicki, E., Orłowski, C., Kowalczuk, Z. (eds.): Foundations of Control and Management Science, vol. (10). Poznań University of Technology, Poznań (2008)
51. Szczerbicki, E., Orłowski, C., Kowalczuk, Z.: Information technology and knowledge management: A predestined integration. In: Foundations of Control and Management Science, Part 1, vol. (9), pp. 4–5. Poznań University of Technology, Poznań (2008)
52. Szczerbicki, E., Orłowski, C., Kowalczuk, Z.: Information technology and knowledge management: a predestined integration. In: Foundations of Control and Management Science, Part 2, vol. (10), pp. 5–6. Poznań University of Technology, Poznań (2008)
53. Turek, K., Orłowski, C.: Proposal of a methodology for the research of clients' demand engaging information technologies. In: Borzemski, L. (ed.) Information Systems Architecture and Technology: Designing, Development and Implementation of Information Systems, pp. 213–218. Publishing House of Wrocław University of Technology, Wrocław (2008)
54. Wasiluk, K.: Testing and assessment of information technology in development along with a proposal of teaching the management of teams producing software – based on the tool of IBM Rational Team Concert – Jazz (MSc thesis in Polish). Gdańsk University of Technology, Gdańsk (2008)
55. Weber, I., Markovic, I., Drumm, C.: A conceptual framework for composition in business process management. In: Abramowicz, W. (ed.) BIS 2007. LNCS, vol. 4439, pp. 54–66. Springer, Heidelberg (2007)
56. Wiegers, K.E.: Software Requirements: Practical Techniques for Gathering and Managing Requirements Throughout the Product Development Cycle, 2nd edn. Microsoft Press (2003)
57. Yager, R., Filew, D.: Fundamentals of Fuzzy Modeling and Control. WNT, Warszawa (1995)
58. Yourdon, E.: Modern Software Analysis. Prentice Hall, New York (2001)
59. Zadeh, L.A.: Fuzzy sets as a basis for theory of possibility. Fuzzy Sets and Systems 1, 3–28 (1978)
60. Zakrzewski, D.: Service management in the process of the evolution of IT support organizations (report in Polish). Gdańsk (2009)
61. (2009), `http://www.architecting-the-enterprise.com/`
62. (2009), `http://www.opengroup.org/togaf9/cert/`
63. (2009), `http://www.opengroup.org/togaf/cert/`
64. TOGAF Ver. 9. The Open Group, San Diego (2009), `http://www4.opengroup.org/sandiego2009/`

65. TOGAF Version 8, Enterprise Edition Study Guide, (8.1.1 pdf edition). The Open Group, San Diego (2009), `http://pubs.opengroup.org/architecture/togaf8-doc/arch/`
66. (2008), `http://www-128.ibm.com/developerworks/rational/library/253.html`
67. (2009), `http://www.mcrososft.com/poland/developer/produkty/vs2008/`
68. (2009), `http://www.omg.org`
69. (2009), `http://www.tmforum.org`
70. (2009), `http://www.feapmo.gov/feaBrm.htm`
71. (2009), `http://www.msu.edu/user/mccarth4`
72. (2009), `http://www.jeffsutherland.org/oopsla2000/mccarthy/mccarthy.htm`
73. (2009), `http://www.openapplications.org`
74. (2009), `http://www.rosettanet.org`
75. (2009), `http://www.w3.org`
76. (2009), `http://www.opengroup.org/sib`

Subject Index