Praxeme Institute

Component

SLB-02e “An initiative for an open method”

Introducing Praxeme

Objective Both private and public organizations are crying out for a comprehensive and shared method to guide their business transformation projects. Given that organizations are increasingly complex, interest in and the use of enterprise modeling techniques are greater than ever. For a method to succeed, it must therefore draw on industry best practices as well as modern modeling techniques and tools. This whitepaper presents an initiative that is working on such a method.

Content

- Synthesis
- Today’s situation
- The Initiative
- The Method

Authors Dominique VAUQUIER, Philippe DESFRAY

Translators Anthony JERVIS, Reviewer: Nigel STRANG

Version 2.2, January 29, 2010

Configuration elements

Availability

This document is available on the Praxeme website and can be used if the conditions defined on the next page are respected. The sources (documents and figures) are available in the private space of the Praxeme Institute site.

Revision History

The first version of this document was published in September 2004. It is quoted, as a reference document, in the statutes of the "Praxeme Institute" association.

Version	Date	Author	Comment
1.	2004		The first French version of this document was published in 2004.
1.99	2006		The translation used the original version in French: v2.0, 14/06/2006.
2.1	March 2009	DVAU, AJER	Changes of both texts (English and French) take into account the progress made in the initiative, specially the creation of the Praxeme Institute.
2.2	18/03/2010	DVAU	Comment in the properties
2.2			Current version of the document

License

Conditions for using and distributing this material

Rights and responsibilities

This document is protected by a « Creative Commons » license, as described below. The term « creation » is applied to the document itself. The original author is:

- Dominique VAUQUIER, for the document ;
- The association *Praxeme Institute*, for the entire Praxeme methodology.

We ask you to name one or the other, when you use a direct quotation or when you refer to the general principles of the Praxeme methodology.

This page is also available in the following languages :
български Català Dansk Deutsch English English (CA) English (GB) Castellano Castellano (AR) Español (CL) Castellano (MX) Euskara Suomeksi français français (CA) Galego עברית hrvatski Magyar Italiano 日本語 한국어 Melayu Nederlands polski Português svenska slovenski jezik 简体中文 華語 (台灣)

creative commons

COMMONS DEED

Attribution-ShareAlike 2.0 France

You are free:

- to copy, distribute, display, and perform the work
- to make derivative works
- to make commercial use of the work

Under the following conditions:

Attribution. You must attribute the work in the manner specified by the author or licensor.

.

Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under a license identical to this one.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the Legal Code (the full license).

Content

Configuration elements ii
Conditions for using and distributing this material iii
Synthesis 5
The need for a reference method 5
A global methodology that aims to coordinate areas of expertise 5
An open initiative, on a stable core 6
Involved parties 6
For more information 6
Today's situation 7
Lack of a method of reference 7
Impacts of under-modeling 8
Costs of under-modeling 9
Exploiting new contributions 10
The need for a common framework 11
The Initiative 12
An open method 12
The Organization of the Initiative 13
The Method 14
The theoretical basis 14
A multi-aspect approach 15
Definition of the aspects 16
Methodology driving action 17
Other structural notions 18
Modeling techniques: tools 19
Modeling techniques: philosophy 20
Modeling techniques: in practice 21
Structuring the methodological repository 22
The "Product" dimension of the methodology 23
The content of the methodology on the "Process" dimension 24
The content of the methodology on the "Procedure" dimension 25
Modeling for analysis and design 26
Positioning 27
Conclusion: synthesis of what has been covered in this whitepaper 30
Appendices 31
The name of the method 31
The definition of Praxeme 31
For more information 32
Short Bibliography 33
The coverage of concerns 34
Glossary 35
Index 38

Epigraph

"*Theory without practice is useless; Practice without theory is blind.*"
Emmanuel Kant

"*And the secret of the whole method is such: find out carefully that which is the most absolute in all things*", Rene Descartes, Rules for the direction of the spirit (rule VI)

Synthesis

The need for a reference method

Complex challenges

Organizations constantly face challenges of variable dimensions. Globalization, for example, impacts modern organizations. In order to adapt, organizations need to operate in a permanent cycle of change whilst preserving the synergy that already exists between its domains of expertise. Additionally the information system is at the heart of all organizations and information and telecommunication technologies should, therefore, be a mandatory feature of any business strategy.

Restoring legibility

This organizational & technological complexity is, however, very difficult to come to terms with. Managing such complexity and inventiveness needs collaboration between several domains of expertise, and yet, even within a given single domain cooperation remains limited due to the lack of common references or frameworks.

Traditional points of reference have fast become obsolete in light of the speed in which modern technologies have evolved over the past decade, generating a disconcerting unease for many on the way. In order to facilitate an organization's transformation, it is therefore necessary to have a modern reference method that is shared by a wide community.

A global methodology that aims to coordinate areas of expertise

Theoretical basis

It is no longer sufficient to rely on any one of the specialized methodologies which cover only one aspect (e.g.: business process modeling, information technology, strategy consulting…) Overall control requires coordination and synergy between areas of expertise in order to align them with the objectives of the organization.

The "system topology" supplies such a reference framework. This framework positions the different aspects in the way that they should logically appear. This multi-aspect approach allows us to:

- Isolate and coordinate informational and decisional elements, both of which are distributed across the transformation chain (from strategy to technology);
- Define "who does what" (roles & responsibilities).

Techniques

With such an ambition, the main concern is communication. How can the knowledge and its implied representations be integrated into an organization or its projects? What must be done to ensure that the participants understand each other and work together harmoniously? The importance of communication justifies the attention that is brought to modeling techniques and the value they represent in the methodology.

The proposed method is based upon standards for representing business and systems which have been fine-tuned over recent years. However, it does not neglect the methodological heritage of past generations. The objective is to bring together the contributions & benefits of each and to provide a road map which guides us, in an intelligible way, through the different aspects.

Synthesis (cont.)

An open initiative, on a stable core

Community approach

The ambition is to develop an open method that is widely known and recognized. Starting from a defined structure, "Praxeme" must take into account the community's aspirations and integrate best practices where available. It is with this open mind that the initiative is instigated.

The rule

In order to integrate the different contributions from the community in the most efficient possible way, it is necessary to have an initial methodological core. This core contains the general philosophy and governs how new elements are integrated.

The objective of this whitepaper is to establish the initial core which all Praxeme contributors agree to.

Project organization

"Praxime" is the name of the initiative, that is to say the project. The method is called "Praxeme"[1].

A non-profit association that complies with the 1901 French law was created and is called "Praxeme Institute". The Praxeme Institute is the trustee for all of the methodological assets, the guarantor of openness and the coordinator for any activity surrounding the conception and publication of the Praxeme methodology.

Existing methodological documentation is progressively being reviewed and published.

Involved parties

Field of study

The method covers traditional analysis and development type projects, strategic road maps and also permanent activities such as governance, infrastructure optimization and enterprise architecture. It addresses all project populations (business and IT) from those in steering committees to each individual project team member. The method, in particular, looks into helping the different populations work together, without inducing any loss of energy.

Audience

The current whitepaper addresses all who are confronted with the job of rethinking the way in which organizations function. It does not presuppose any specific technical expertise.

For more information

Please refer to page 34 for the glossary of terms used in this whitepaper.

Please refer to page 32 for the bibliography.

[1] The meaning behind the word is given on page 31.

Today’s situation

Lack of a method of reference

France in the 80s & 90s

The method applied by several generations of Modelers, for whom it was a reference, was called “Merise”; It was a method that began from an initiative of the French administration. Merise became obsolete with the appearance of new standards and practices for the modern technologies of the last decade.

Concurrently, the SDM/S methodology was also considered an important framework for many French organizations. However, it was considered inflexible & too complex and eventually suffered the same fate.

Methodological weaknesses

Since the beginning of the IT-era, processes and methodologies have changed depending on the techniques in vogue: from a structured to a spontaneous approach, from Merise to Rapid Application Development (RAD) type approaches, from the waterfall model to the V-model, from the Y-model to the spiral model, from a full-method approach to anti-method movements, from safe “ISO” or “CMM” approaches to their outright criticism for being counter-productive. The result of this ebb and flow is that many doctrines exist but their accumulation has lead many practitioners to lose their bearings.

In addition to this, specialized perspectives have appeared that address specific concerns: business optimization and processes, strategic and economic models, infrastructure, etc. These perspectives cover wide scopes, but their separatist tendencies make them difficult to assimilate.

Persistent issues

Current methodological documentation remains scarce. Even with respect to the UML notational standard. Isolated patches of documentation exist but each centers upon different subjects without providing an integrated view of the big picture. Often, it deals with a particular domain of expertise in which the concerned community has become highly specialized whilst becoming less able to communicate with others.

The absence of reference material for the wider community brings about:

- Difficulties in communication within projects and organizations
- The loss of large amounts of energy & creativity and the under-exploitation of possibilities that are available at hand
- Waste of produced documentation and models that prove to be unexploitable as they are not placed in the correct context within a consistent approach
- Imprecise estimates of required expertise and deficiency in the training of personnel

Today's situation (cont.)

Impacts of under-modeling

The shortfalls of modeling

The activity that suffers as a result of this phenomenon is, without a doubt, modeling (in all its forms.)

Symptoms A few examples of current difficulties:

- The lack of a global framework provokes a mix-up in the level of detail in the models: the Practitioner also does neither know what must be expressed in a model nor the aspects he should represent. He suffers from a lack of predefined expectations in terms of quality: Is the model complete? Is it too detailed or too vague?
- It has always been difficult to produce true conceptual models which are free of technical assumptions. It has, also, always been difficult to choose between what is essential and what is contingent, and between what is conceptual and what is organizational.
- It is also not uncommon to find a model of a system's architecture represented on one or two pages. The aspects of the problem being addressed are not always dealt with exhaustively, the choices not always justified and the associations with prior models are ignored.
- In addition, the description process and links between the different models are not always defined. In consequence projects tend to aim straight for the solution, skipping the design stage without exploring different alternatives and innovative solutions.
- Encouraged by ISO 9000 norms and driven by strategic ideas, modeling of business processes requires significant amounts of energy. Several factors amplify the consequences of these investments: the choice of standardized notations, links with other modeling activities, and the influence of new management styles and new organizational theories.

Diagnostic Before becoming involved in the details of models and the methods of developing them, the Modeler must answer two vital questions:

- What should the diagram contain?
- What is a good diagram?

Today’s situation (cont.)

Costs of under-modeling

Costs

Without a set methodology, coordination of the different activities proves difficult and the lack of objectives results in an unclear vision of the big picture. Both of these consequences are costly to an organization. The direct costs impact the project’s deliverables, including the ‘customary’ slip-ups (delays and overspending). The indirect costs are measured on the organization itself, which is directly impacted by a maladapted or late solution.

Direct costs

Direct costs increase exponentially with the amount of resources being used. For example the absence of UML on an IT project has consequences on the whole project: the time wasted producing useless or unusable diagrams, or those parts of the design that are not covered by the model and as such remain unclear. Typically, for a business process reengineering project, sometimes involving several hundred participants, the absence of a rigorous methodology leads to considerable waste to the point of ruining the project: discussions, or even conflicts, caused by a confusion in terminology, going into too much detail with a loss of vision of the objectives, personal points of view that prevail over the common good, the cost of off-subject considerations compared to the cost of producing pertinent materials, etc.

Indirect costs

Indirect costs, albeit more complex to estimate, are of a larger scale[2]. They concern the impacts on the organization and on its way of functioning. Insufficient analysis & design increases the risk of implementing an incomplete solution, not being able to achieve the operational benefits, not exploiting possible business, organizational or technological opportunities, or introducing an inapt business process…

[2] Please refer to the enlightening Standish Group report for more information.

Today's situation (cont.)

Exploiting new contributions

Techniques & abstraction

1970-80s: Discovery of modeling techniques

Programming techniques (symbolic, functional, structured, modular, object) and domain issues (databases, real-time, network, parallelism, and human-computer interfaces) gave birth to modeling technologies in the 70s & 80s. For a large part, the principal modeling techniques had been identified at this time (entity-relation, business process modeling, state diagrams, etc.) Each methodology, however, had its own variation for each, some were customized for the domain for which it was intended (service, data-oriented, process-oriented, etc.), by country or by the target technical domain (programming language, infrastructures, etc.), hereby producing numerous different methodologies, each focused on a specific point of view.

The lifespan of each methodology was conditioned upon the underlying techniques or languages, or by the targeted domain or country.

1990s: unification of modeling techniques

Metamorphosed by the constant renewal of techniques and the opening up of organizations, methodologies in the 1990's progressively united the modeling techniques that were considered stable and commonly utilized. Object-oriented modeling was a main federator. Component-based & improved business processing modeling completed the transition.

Conclusion

Now, at the dawn of 2000's, we can take advantage of our knowledge of modeling techniques. The level of maturity and stability provides a solid starting point from which a global methodological approach can develop.

The advent of standards

Amongst recent contributions, two standards arose to complement modern modeling techniques: UML[3] & MDA®. These standards maintained by the OMG[4], were the result of international consensus and were rapidly supported by modeling tools that facilitate their application.

UML

UML (Unified Modeling Language) federates different modeling techniques. It is a toolbox providing a complete range of diagram types. It is a language and a representation technique but it is not a methodology. This is the source of much confusion in contemporary projects. Defined by (amongst others) a consortium of software editors, UML is driven from the bottom-up so it is understandable why IT departments have adopted it rather than other areas of the enterprise. Consequently, many consider the use of UML to be restricted to technical profiles. However, UML offers strong modeling capabilities which can be of value to other activities, elsewhere in the software development lifecycle.

MDA

MDA (Model-Driven Architecture) puts the emphasis upon:

[3] Unified Modeling Language. The modeling standard from the OMG, published in 1999 (UML v1.0) underwent a major upgrade in 2002 (UML 1.4). The latest version (UML 2.0) was published in 2004.

[4] Object Management Group. The OMG is an international organization that is known for having defined the CORBA standard as well as other interoperability norms & standards

- The notion of models that are independent from constraints tied to technical platforms or that represent a business domain or activity (PIM[5]).
- The notion of models that are specific to a technical platform and which aim specifically at describing software implementation issues (PSM[6]).
- The automatic transformation from one model to another and the generation of programming code.

An organization's collection of technology independent models is both an important knowledge capital asset and a key element for speeding up the design and production of new software solutions. This approach to development, thanks to perspectives in model & code synchronization, has the potential to revolutionize current practices in the design and development of information systems. Industry experts and consultants already proclaim the virtues of MDA (consult bibliography for further, related reading, including the Meta Group's whitepaper).

The need for a common framework

The aim of the reference methodology

The previous sections have shown us that the methodology must provide ways to ensure the successful collaboration of different areas of expertise and increase consistency in terms of the contribution from each business area. Whether it is on the scale of a single project or a single organization, there is an immediate need for a common framework. This is qualified as being a "vertical" requirement.

The reference method is also designed to be applied on a "horizontal" level, meaning that it contributes to the definition of the necessary activities and skills that organizations must recognize and adhere to. Once in place, this will make it easier for people in the organization to move between projects and business contexts.

In conclusion

It is clearly urgent to build a methodological framework that is based upon proven experience and which benefits from previous methodologies. There is a huge amount of knowledge and experience available on best practices and standards that have now matured. It is our objective to collect and to reorganize this knowledge and to constitute a methodological referential[7].

The new methodology must be free from the assumptions that restricted the use of previous methodologies and must adopt the new "paradigm" that looks at how we (re)think our organizations. Rapidly this paradigm is characterized by the following orientations:

- Extensive use of object-oriented modeling techniques[8].
- Recognition of semantic modeling, that is to say the representation of the truth as it is understood, independently of organizational and technical considerations.
- Acceptance of the importance of the human dimension and of communication.
- Recognition of the complexity of systems.
- Acknowledgement of the economic dimension and its influences on all of the other aspects.

[5] Platform Independent Model

[6] Platform Specific Model

[7] "Referential" is used, here, in the strongest of sense of the word: a collection of objects or information that is shared by a community. As such, the community that the Initiative is addressing encompasses the crafts which relate to rethinking organizations, from the definition of their activities and deliverables to the building of the entire value chain (organization, process, automated solutions…)

[8] The object-oriented approach – monopolized by the computer industry –, itself, is the extreme outcome of a profound conceptual revolution.

The Initiative

An open method

The project

The Praxeme project is elaborating a reference methodology – Praxeme – which can:

- Fill the void that was left by the demise of previous methods (as SA/SD, Merise & SDM/S), whilst taking into account evolutions in technologies, practices and needs.
- Deliver a common framework in which the knowledge of systems (organizational, functional and physical) and know-how are clearly organized.
- Lead the thought-process and work upon these systems, from analyzing their objectives to their implementation.
- By organizing these systems, take advantage of contributions originating from different practices by using standards throughout.

The objectives of the methodology

A method is always driven by motives towards action. The following sentences outline concrete objectives of the method that is being built:

- Control system complexity: insofar as the extent of their interactions are concerned and in solutions that result from the combination of sociological and technical influences.
- Reduce costs and stimulate synergies: clarification of responsibilities and the collaboration between the different expertises.
- Restore clarity in organizations (companies, public bodies, organizations and human & technical systems): definition of the necessary modeling techniques, guidance in the utilization of techniques and standards such as UML, MDA, BPM, information system mapping…
- Elaborate precise techniques, which are recognized and shared by the wider community: system engineers, functional analysts, information system architects, software developers and industry experts & consultants.
- Automatically orient and generate a part of the deliverables.

The means behind the ambition

The priority is on mapping out the expertises, conciliating the different points of view and the definition of the chain of activities.

This program demands, first of all, an overview of the complete picture. It has to be sufficiently understandable for it to be communicated, yet sufficiently rigorous for it to be effective.

The theoretical core fulfills this ambition. Having this to begin with, constitutes the basis on which to consolidate the different contributions and to bring about the collective development of the entire method. The "system topology" defines the general framework and the order in which the different contributions are integrated.

The Initiative (cont.)

The Organization of the Initiative

Openness

The open nature of the future method is not only to be read at face value. It is the fundamental response to the issues that were brought up in the first part of this paper. Only a method that is widely shared and recognized will be able to take on the role of “reference” and to help achieve the designated objectives.

The initiative, therefore, is open to all volunteers and is actively seeking the widest consensus. The project’s organization, laid out below, is defined in such a way so as to ensure that the core is respected and the integration of each contribution is consistent. This organization is specified in the statutes of the “Praxeme Institute” association.

Institutional aspects

This initiative for an open method positions itself, quite naturally, under the patronage of the powers that be. It is seeking support from major industrial and service sector companies. The proposed organization has two levels of commitment:

Sponsors Sponsorship is the strongest level of commitment and as such Sponsors can benefit from the ability to have a personalized analysis done on their methodological needs for their particular context. Any resulting demands will weigh heavily on how the methodology is oriented. Close relations with Sponsors can accelerate the delivery and the assimilation of the methodology’s components, as they are produced.

Observers Other than the above, previous level of commitment, the Initiative is open more widely to all who show their interest. The results of the work on the methodology will be released to them on a regular basis.

Operational aspects

The Contributors college In order to ensure a good level of consistency – at least at the methodological core –, the drafting of the methodology is restricted to a small group of authors whom are renowned for their practical, theoretical and pedagogical experiences. They will be supported, where necessary, by experts in a given aspect or technique.

The Reviewer’s club On a regular basis, the Reviewer’s club meet together in order to review the current orientations as well as the delivered components, to verify the transmissibility of each component and to vouch for the respect of the tradition.

Financial aspects

The financing of the Initiative is ensured by the Sponsors. It covers the payment of authors and expenses associated with logistics, assemblies and marketing & communication.

The following pages provide an insight to the content of the method.

The Method

The theoretical basis

The starting point

The concerns that were expressed in the "Today's Situation" chapter lead us to ask a fundamental question:

"In an organization, what must be represented in order for it to have a positive impact and to better control it?"

Indeed, if the objective of the methodology is to explain how to operate, what to produce and how to decide… its first job is to clarify the general understanding of the subject that is under study. What is it? How must it be described and communicated?

The "What" precedes the "How." The description of that of which the System is composed, is part of the foundations of the methodology and is a precondition to the definition of the activities to be performed on it (i.e.: the transformation process).

The System

We remain voluntarily evasive on the definition of the System; in some cases, we talk of an Enterprise System in order to emphasize upon its global dimension, whereas in others we talk of an Action System that takes into account other types of issues. The latter could possibly be Human, Organizational or Physical, all of which have an IT component playing a more or less prominent role[9].

What is trying to be achieved is not only a description of the IT system but that which is actually occurring in real life: a description of the actors and their activities – those things that we know are tangible. This choice clearly expresses the desire to start at the beginning: that is to say, to not only consider the IT system as the be all and end all but simply as one possible way, amongst others, of dealing with the issues at large.

The first models that we recommend, therefore, are not ones that describe the software solution but ones that deal with this "reality."

The Enterprise System Topology

The principles of our methodology are set forth below:

- Reality must be approached from several angles: these angles correspond to aspects in the topology, which relate to the different skills and pre-established perimeters.
- Reality must be described by several models, each one specialized according to the particular aspect.
- There is set number of aspects: they are defined by the methodology and articulate well with each other.

The "Enterprise System Topology" defines the different aspects and their articulation. It constitutes the readers guide, the framework that interprets how the topology should be applied to a reality. Each aspect has an ordered list of questions to ask and decisions to make.

[9] There have been many cases that demonstrate its large range of application; in sectors as varied as insurance, army, defense or industrial systems…

The Method (cont.)

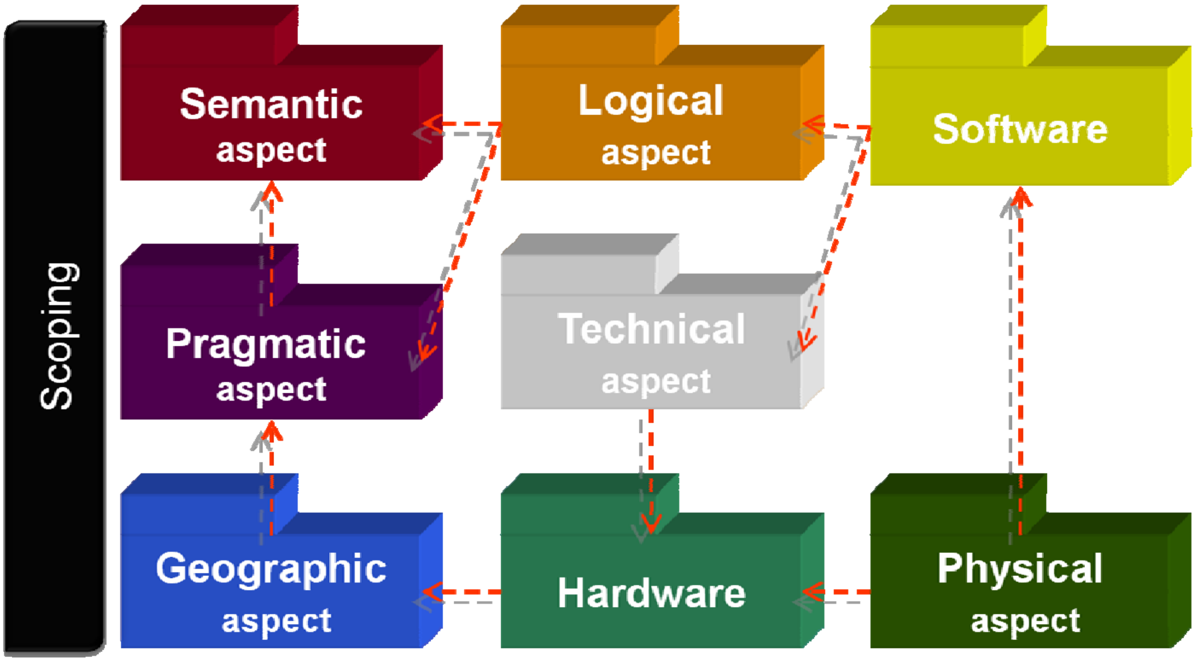
A multi-aspect approach

The aspects

The figure, below, synthesizes the aspects defined by the topology of the enterprise system.

The aspects stem from the questions we ask in order to analyze a system. They define the viewpoints from which we perceive this reality.

Figure SLB-02e.1. The diagram of the Enterprise System Topology



Comment

A Pragmatic model describes the actors involved in the system and their activity, by means of roles, processes, use cases… Such a model conveys organizational choices and habits which have piled up over time. Upstream, the method recommends isolating the core “business” knowledge. This core is much more stable and sharable than the pragmatic aspect. It is the role of the Semantic model to formulate this basic knowledge in a formal way.

The Geographic aspect is about constraints and decisions regarding the location of activities. These three “upstream” aspects constitute the external view of the system, as the actors can perceive it. The topology then identifies aspects related with tooling and automation. These “downstream” aspects help build an operational and technical solution.

The following table provides the definition of each aspect.

The Method (cont.)

Definition of the aspects

Separation of aspects

The systems that we deal with are certainly complex ones. They embody a large amount of information and questions that concern different disciplines in the business area. In order to control this complexity, it is preferable to separate the concerns into harmonious subsets, each one applying to a limited discipline or specialty.

The separation of aspects helps to orient the description and to ease the system's evolution.

Aspects An aspect is a view on the system. The system is seen from the point of view of a particular concern. A cohesive set of concerns constitutes an aspect. An aspect, as well as being a part of the system, is dependent on a point of view, a type of concern or a specialization.

The table, below, defines the eight aspects that make up the topology (see the "General Guide" – ref. PxM-02 – for explanations as to how these aspects were defined).

Figure SLB-02e.2 The definition of the aspects in the Topology of the Enterprise System

Aspect	Equivalent Terms	Definitions
Semantic	Conceptual, business essentials, basics	The semantic aspect is only interested in the objects that concern a business reality. Here, the fundamental elements are described independently to the way in which it is implemented – it represents the "knowledge".
Pragmatic	Organizational	The pragmatic aspect reunites decisions & choices relating to the way in which activities are performed: organizational units, roles, responsibilities, lifecycles, business processes, work contexts.
Geographic	Contextual, communication(al)	The geographic aspect concerns the positioning of elements and activities. Elements such as sites and telecommunication network requirements are described.
Logical	Functional	The logical aspect is an intermediary one that allows certain structural decisions to be made with relative independence in terms of technical constraints or implementations.
Technical	Technological	The technical aspect is where technological decisions & choices are made as well as including how to implement them.
Hardware	Logistics	The hardware aspect concerns the physical elements (machines, nodes, etc.) and their characteristics of which the system is composed.
Software	Application, IT	The software aspect covers the software components that automate parts of the system.
Physical	Deployment	The physical aspect describes the relationships between software elements (including databases) and the identified hardware.

The Method (cont.)

Methodology driving action

Derivation rules

In figure SLB-02_1, above, we can see that the different aspects are interlinked. These links are designed in order to increase independency between aspects, to organize the activities effectively and to control the repositories (models, descriptions, documentation, etc.) associated with each.

Models are produced by aspect. In MDA terms, certain models can be categorized as being independent with respect to the expected target platform, whereas others are dependent upon a given platform.

The dependencies that have been defined between aspects synthesize the derivation rules that transform elements from an aspect on a higher level to other more "technical" or internal equivalents of the next aspect. In some instances, the UML profile mechanism can be utilized to automate certain derivation rules. Using the combination of modeling and development tools, these transformations guide our understanding of the system, from its abstract, conceptual form to its runtime equivalent.

Impacts on activity planning

Before even considering such a matter as the management or the lifecycle of the system, the multi-aspect approach, as defined in the Topology, brings about the following advantages:

- **Specialization by discipline**: Modeling techniques[10] are specific to an aspect and as a consequence, the knowledge & skills necessary to work on a particular aspect are already pre-established.
- **Organization of information and knowledge**: The Topology provides the structure for the documentation repository, at the project-level as well as at the whole organization level. It defines the framework in which each piece of information and each decision on the system is mapped and becomes a part of the knowledge capital.
- **Categorization of decisions**: By applying the Topology, decisions to make are sequenced. The processes to build and manage the system are based on the framework.
- **Linking of different aspects' models**: In some ways, this also suggests linking disciplines to each other. An element that is represented in the Semantic model will be manipulated by an organizational element in the Pragmatic model, before being transformed, first of all, into a Logical element and then a Software component. In the Physical aspect, it will be associated with a Geographical site and Hardware. All the above entail micro-decisions that are made within a specific transformation chain, but which are part of a bigger chain that controls production at the scale of the system and not only of projects.

[10] Mainly based on the UML standard.

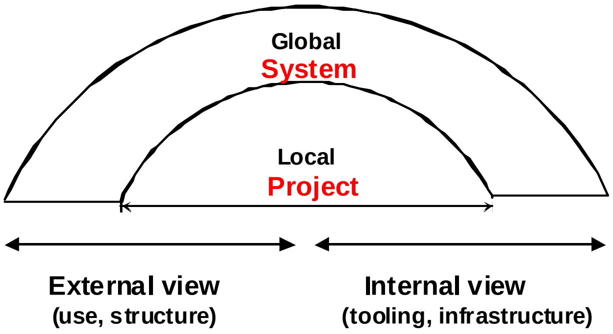
The Method (cont.)

Other structural notions

Scale

In this sense, scale refers to the notion in which the action can be measured in terms of extent and time. Equivalent terms: scope, range.

Figure SLB-02e.3. Scale – local or global



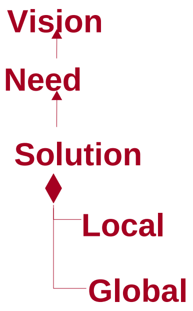
Local scale Investments are, almost always, made on a project basis, along with its clear objectives and short term effort. The methodology must provide clear answers at this level of organization, which respects the "local scale". It applies to easily identifiable elements (a localized business process, an activity or an application).

Global scale However, our aim is not only limited to helping projects succeed. It is also aimed, if not more importantly, on ensuring overall consistency, controlling the knowledge capital, management and consolidation of the system across all aspects, including its external context. The methodology must, therefore, take into account the "global scale" –the system (enterprise, etc.) as a whole and over time. It is only by doing this that it can enter into a qualitative optimization cycle.

Levels of targets

In addition to the notion of scale, the "positioning" system also contains the notion of target against which activities are defined. The target level allows to situate the "depth" and to define the appropriate attitudes to summon within the organization in order to ensure correct transformations.

Software development methodologies are exclusively concerned with the local scale and a "local solution" (moreover, this solution is generally limited to its IT perspective).



This local solution is integrated into a global one that corresponds to a business requirement. The latter, itself, falls within a long-term perspective that is expressed as a vision. Organizing the concerns into levels allows the organization to reconcile with its different rhythms: long-term road maps (strategy, doctrine), reactions to the environment (market, politics), current projects and general management.

The "positioning" system constitutes, hereby, a tool for identifying and sequencing the activities within the process that accompanies the Enterprise System.

Figure SLB-02e.4. The levels of targets

The Method (cont.)

Modeling techniques: tools

The approach

The Enterprise System Topology can be thought of as a checklist or a methodological framework. For each item (aspect) in the list, it remains for us to identify how we represent its contents. To do this, we recommend, as a preference, the use of an object-oriented approach along with the underlying UML language standard[11].

Utilization of UML

UML is not a method! UML supplies the tools for modeling (the diversity of which has increased with UML2.0) but not the instructions on how to use them. The Praxeme method relies on this ready-to-use toolbox and aims at providing the necessary instructions. Notably, it will describe how to model in UML for each of the different aspects.

The advantages

UML can be used for many things. Its first advantage is that it can be used by profiles in all areas of the organization. It can also be used for all types of modeling. Not content in being a standard language, it rises to the level of being a common language. The second advantage follows on from the first one as all Modelers, Organization designers, Architects and other Designers will be able to use the same modeling tool. This facilitates the constitution and management of an Enterprise Repository.

Other forms

Praxeme remains open to other contributions. Each time that UML covers a modeling requirement, its notation will be preferred, but this does not prohibit the utilization of other, more specific, techniques (e.g.: Ishikawa diagrams, breakdown structures, objectives hierarchies, glossaries, requirements…) It is quite reasonable for us to not recommend the proliferation of different modeling notations, variants and dialects with the methodology … this diversity is the exact contrary to the intended outcome of a method of reference.

The utilization of formal languages, such as OCL[12], is also recommended in order to complete the graphical UML diagrams.

[11] This is not an absolute decision. Sometimes, for an aspect or another, other approaches can offer better alternatives, more appropriate to a specific need. It may also happen that we have to use traditional tools in specific contexts or for tactical reasons.

[12] Object Constraint Language: an OMG standard that falls within the MDA category of standards

The Method (cont.)

Modeling techniques: philosophy

The quality of a method

The quality of a method lies in two fundamental criteria: pertinence and consistency.

Pertinence It is a common mistake to build a theory (a model) that is not adapted to reality. The model will not be pertinent and will not provide an understanding of the system nor provide the ability to react appropriately on it in order to optimize it. Pertinence can only be guaranteed by following an approach that relies on carefully elevating information regarding problems, on steps dedicated to reviewing with the right people and on permanently justifying modeling decisions. Thus, there is a need to build a common understanding of the system as well as to justify a detailed model based upon this shared understanding.

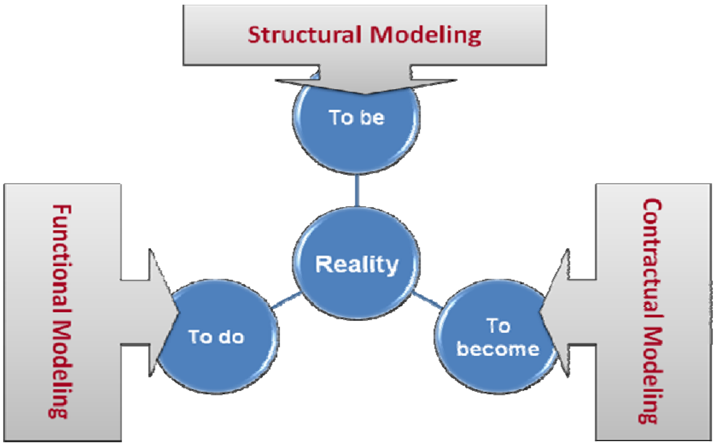
Consistency Consistency requires defining a set of views on the model, according to the rules which ensure completeness and guarantee its overall consistency. Thus, after having ensured the pertinence of the theory, we set out to define a consistent one.

Triangulation

Coordinating the different UML diagrams helps us to achieve this qualitative objective. Beyond the traditional "static" and "dynamic" representations, we emphasize on a "by contract" approach to modeling. It provides a way of coordinating the different types of diagrams defined in UML.

This ternary approach prompts us to consider the reality that is to be represented in a more systematic manner and it produces a set of models that are based on facts. This increases the quality of the models. Generalizing the use of states diagrams[13] facilitates the modeling of contracts by guiding the definition of "pertinent" and "complete" processes.

Figure SLB-02e.5 The three axis of modeling



[13] UML2.0 reinforces the definition of "protocol state machines." This modeling technique allows for the specification of a lifecycle for the entities of a system. According to the services provided by an entity, it describes in a clear & synthetic manner the authorized states and transitions.

The Method (cont.)

Modeling techniques: in practice

The level of detail

Even if the approach can be adapted on a case by case basis, the method must be broad and deep, covering all of the aspects and going into all levels of detail. It must allow us to avoid the usual dilemma where we are divided between:

- On one hand, high-level approaches that only apply for making decisions that are general and which have a small validity criteria (architecture road maps, planning…)
- On the other hand, low-level, detailed approaches, with precise procedures but which are incapable of maintaining the inspiration and motivation from above.

As such, we try to not mix-up the conceptual level with the detailed. The "high-level" models are detailed – they express all that is known at the abstract level at which they are situated. All definitions and decisions are placed at the right level, hereby avoiding any inappropriate decisions at the time of implementation.

The "high-level" model contains real value for the organization and is a source of knowledge and know-how relating to the business.

Text & graphics

We do not fully comply with the latest "visual modeling" tendency in which the model is reduced to a collection of diagrams. Admittedly, diagrams constitute a necessary tool for communication, but by no means can they replace text. Textual descriptions of elements and the justification of certain modeling decisions are key to the comprehension of the diagrams that make up the model.

Complementary textual descriptions bring important explanation and further enrich the model by completing undefined parts of the model or those that are considered to be implicit by the modeler. The methodology and its supporting tools must provide the means to define the right balance between graphical & textual descriptions and to ensure consistency with each other.

Repositories & models

The collection of diagrams must be maintained in a common repository. This enterprise-wide repository, itself, should be structured by aspect, respecting the Praxeme Topology. Each aspect's models are combined with the others to make an exhaustive "global" model[14].

[14] Models, specific to each aspect and produced by one project after another, are literally dumped into the repository. However, in order for the models to work together and produce the benefits that are expected, the repository must have been appropriately structured to allow projects to comply with the modeling principles and rules for integrating their models into the repository.

The Method (cont.)

Structuring the methodological repository

The Transformation Model

Concepts that are mandatory to the methodology and to its functioning are presented in a Transformation Model. It is the methodology's "metamodel" (its terminology, its different categories of representation) The Transformation Model, where possible, should be based upon the SPEM[15] standard and other standard metamodels. It plays a vital role in the engineering, as well as in building consensus, of the methodology. It is a guide as to how the chain of activities can be tooled and is also very educational.

PRO^3

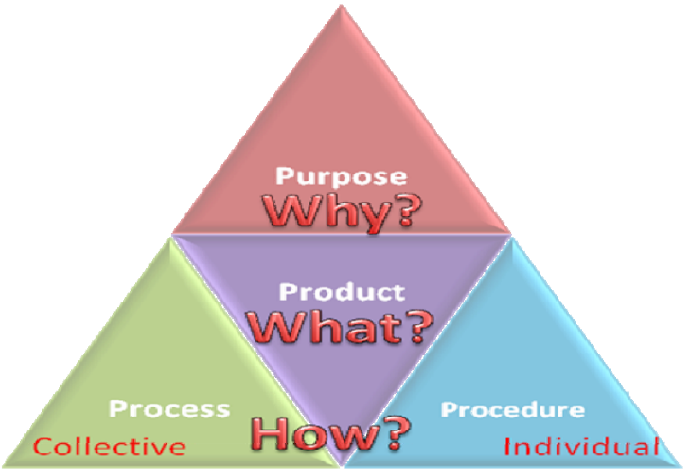
The metamodel provides the method's syntax. It is important to note that the methodology is deployed in the three dimensions – Product, Process, and Procedures – (assembled together in the "PRO cubed" formula) that are defined below and illustrated in the following pages.

Product Elements relating to objectives, to objects that are to be produced or modified, to the system itself and to its different facets, are all grouped under the "Product" dimension, independently of how they will be produced. It is the answer to the question "*What?*" It is here that the projects deliverables are found.

Process In a methodology, the "Process" is the definition and the organization of the activities and resources. It answers to the question "*How?*" at a group level (a project or a whole organization).

Procedure The Process provides guidance at a collective level; it remains to define how to guide the worker and to answer to the question "*How?*", but this time at the level of the individual. A Procedure is a way of doing something, an operating mode, a technique or method to execute a task, possibly with the help of a tool.

Figure SLB-02e.6. The methodological repository adopts the PRO^3 structure



[15] SPEM: Software Process Engineering Model, an OMG standard that provides the outline for how methodological corpus should be structured in order to ease interoperability.

The Method (cont.)

The “Product” dimension of the methodology

Petitio principii

The most visible, stable and most exploited element of a methodology is *what* it produces: the deliverables and intermediary products. Processes come and go and are applied at a variable degree, but deliverables remain the most tangible elements of all. From this viewpoint, the methodology can easily deduce the most optimal way for producing its deliverables. The process is derived from the knowledge of the expected deliverables and the requirements which constrain them.

Notions

The System and its aspects (Topology), the relations between the aspects, scale and target levels, all exposed in the previous pages, are all salient notions that have allowed us to understand the Product dimension. We’ll also find more traditional notions such as the model and architecture in this dimension. The methodology provides details on the terminology in which these deliverables will be expressed. The value of a deliverable resides in its ability to communicate information to an audience at a given level. Therefore, it is reasonable to adapt the terminology to the aspect in question as well as the intended target level.

Models & architectures

UML defines the types of diagrams but remains mute as to what constitutes a model[16]. The first contribution of the methodology is the rigorous definition of the necessary models and their acceptance criteria. The choice of diagrams is secondary[17].What is more, each type of diagram can be used in several ways. The role of the methodology is to put these varied utilizations into a consistent framework.

Models can apply on the scale of a specific solution or on a global one. In the latter, we often speak of Architecture. For example, there are logical, technical or physical architectures. Each aspect is, indeed, examined on both of these scales. When an organization applies a logic of governance or Enterprise Architecture, these specific models are integrated into the repository according to precise guidelines.

Ambition

Our ambition is to obtain an exhaustive inventory of modeling elements and decisions, as well as how they fit into the overall transformation chain, beginning at the management level and extending down through to IT development teams or beyond. The articulation of all these elements, at a fine-grain level, restricts loss and opens the path to automated transformations[18].

[16] Far from being a loss, this limitation is constitutional of UML: without it, there wouldn’t even be a standard for modeling.

[17] It is our basic principle to first of all be clear on the *what* and the *why* for our actions before studying how they will be performed. Many others propose, first, to define which UML diagrams to use without identifying the reasons why, avoiding the notion of a *model* and how it will be communicated. These situations generally fail as they are unable to bring sufficient intelligence behind the actions in order for them to adapt to the situations that they will have to face.

[18] Following the definition of models and architectures – definitions that will be a constituent of the first deliverable – the method will propose a wide range of forms and document templates along with their associated comments. These tools do not aim to replace knowledge of the methodology, but to guide it in its application.

The Method (cont.)

The content of the methodology on the “Process” dimension

The meta-process

This dimension has been largely documented in methodological literature over the past years. Rather than proposing yet another phasing or rather than laying out what could be a universal approach to project management, our concern is to demonstrate how to elaborate an approach in a given context and with enough rigor to ensure its success. It is therefore a question of:

- Verifying that activities function smoothly, especially with respect to the requirements established for the production of deliverables;
- Integrating available processes;
- Completing the missing blocks in the methodological repository.

Parameters

A real process (as applied to a project or to an organization) is very dependent on its context. Managers are careful when adapting the organization’s reference processes, very often in order to make them simpler. Simplifying a process may have impacts on the conditions in which it functions, and must understandably be done carefully. Therefore, the methodology must clarify the formal conditions to which the process must abide, as well as mention how the process should be developed. To do this, it must outline the process’ parameters that Managers can adjust, thus giving them a certain amount of liberty to customize the process yet containing any possible risk of dysfunction.

SPEM

SPEM[19] is a standard that formalizes and authorizes exchanges between variants of processes. As such, processes are manipulated as if they were components and are adaptable to different contexts. As a result, projects can set themselves up with “à la carte” processes, exploiting varied methodological resources. Respecting SPEM helps to manage flexibility in processes and their functioning.

Integrable blocks

The Process dimension is not limited to a single process (such as a software development process.) It incorporates several processes due to the need to cover different target levels and the twin scales – processes at the organization level (global scale) and those at the project level (local scale) which evolve at different rhythms.

The task is to integrate the numerous contributions that are available today. For example: ISO 12207, RUP, RM-ODP, ITIL (see page 27).

Proposed blocks

Sometimes it is necessary to complete the picture with new, freshly defined activities in order to cover all the process requirements. These concern, in particular, high level activities: activities that are generally found at the global scale, activities that are associated with evaluation or optimization, and activities that are considered “horizontal” or administrative activities.

Organization

The Process dimension includes the definition of roles & responsibilities, as well as the instructions and procedures that are required to move from intention to action.

[19] See page 22 for the definition of the SPEM term.

The Method (cont.)

The content of the methodology on the “Procedure” dimension

Associated notions

A procedure details process “know-how” and, as such, is closely related to operating modes. The discipline, here, is to aggregate the different steps into a consistent whole, which, itself, must be compatible with the intellectual and operational abilities of a given individual. The notion of professional expertise is joined with its art (the art of engineering, the art of Architecture, etc.).

The methodology is expected to explain how to operate with a level of precision that is detailed enough to guide operations on the ground. For example, having already defined what a semantic model or a technical architecture is and having identified the expectations of them, it remains to explain, in detail, how to fabricate them and how to make the right decisions. This level of operational detail is not treated by processes; it is the role of the procedure.

A canonical typology of activities

Exhaustive research into activities considerably helps project managers and directors. It complements the analysis and management of expertise and is integrated with human resource management, training and career road maps. The typology of activities – or discipline tree – includes:

- Management (steering, administration and management),
- Development (research and design),
- Environment (administration and consolidation, applied to all aspects),
- Support (assistance, methodology and quality, audit).

This typology facilitates the organization of procedures and ensures the full coverage of requirements in operational and organizational skills.

Examples

The initiative is developing, as a priority, procedures and methods in:

- Modeling
- Semantic modeling
- Process design
- Logical Architecture design
- Enterprise Architecture (target & transition phases)
- Technical Architecture design
- Software development
- External design (ergonomics)
- Testing, reviewing, metrics analysis
- Internal design (detailed designs)
- Economic evaluation
- Dashboards
- Infrastructure consolidation

The Method (cont.)

Modeling for analysis and design

Analysis vs. Design

The “analysis/design” dichotomy is very familiar in software engineering (and to engineering on the whole). However, recent evolutions (notably in the widespread use of iterative approaches) have led to a blurring of this simple contrast.

Traditional development approaches identified activities (analysis or design) with phases (separation of tasks into units). They respected the waterfall model which arranges activities into specific sequences. Now, it is necessary to distinguish between the two notions, as the same phase can contain both types of activities. Iterative development cycles, often, include both analysis & design in a single iteration. The RUP methodology distinguishes the notions of phases from types of activities, or “disciplines.”

Having said this, it is necessary to eliminate the troublesome connotations that these terms drag along with them. Analysis is perceived to be work that is considered generic or functional, whereas design dives into the details. This is not our approach. Before being steps in a lifecycle, analysis & design reflect alternative behaviors of a same reality.

Definitions

Analyze To analyze is to observe. The term evokes breaking the subject down into its finer grained elements and paying attention to details. The posture of a Modeler during analysis is characterized by:

- Passivity (she/he does not take initiatives; she/he is content with documenting what she/he observes).
- Attention to detail and exhaustiveness.
- Ability to trace dysfunctions with the aim of providing input for reviews (as-is analysis, auditing).

Design To design is to invent. The posture of this Modeler is very different to the previous one. It is expected of her/him to imagine one or several solutions to a given problem or a requirement. The characteristics of this activity are:

- Taking into account requirements (which have, beforehand, been formalized and analyzed) or the coverage of a need.
- Inventiveness, applied to experience in the state of the art and its potential.
- Qualitative and quantitative evaluations of the proposed solutions.

Messages

1. Analysis & Design activities apply to all aspects of the System. The “business” or upstream aspects (semantic, organizational and geographical) are just as likely to be candidates for design proposals as are the more technical aspects.

2. Be it analysis or design, a model is not complete until it has been described in detail[20].

[20] This does not forbid the production of generic or incomplete models which correspond to a particular intention (sketches, architecture decisions, etc.).

The Method (cont.)

Positioning

On the principle of things

The following pages describe how the initiative positions itself with respect to a few methodologies that are common today.

Our methodology principally seeks to integrate methodological content that is proposed by others. In general this third-party content, often specializing in one or two aspects of our Topology, concerns a narrower scope than our method, whose ambition is to cover all aspects of a given system. The integration of their content into the methodological corpus is simplified due to their limited specialized coverage.

These specialized methodologies contain best practices which it would be foolish and unproductive to ignore. However, integrating third party content must not upset the equilibrium that is already established. There are rules to be respected, beginning with SPEM, as discussed earlier, as well as our Topology of the Enterprise System.

Legacy of traditional methods

Praxeme fits in the software engineering tradition and recognizes its inheritance from traditional methods such as structured analysis and design, functional and systematic approaches and, in France, the Merise methodology.

Merise

Amongst the things we retain from Merise is the notion of "levels of abstraction", in accordance with the "separation of concern" principle. This powerful principle has considerable effects when it is applied correctly. It has been recently rediscovered in the MDA standard, which presents the principle but not the content. The "Topology of the Enterprise System" revisits and updates the notions of level of abstraction and level of concern. Praxeme also retains the desire to "normalize" models.

SDM/S

Widely applied in France during the 90's, SDM/S was a reference methodology for the Public Sector and large enterprises. From it, we essentially retain the methodological conduct dimension (development process, lifecycle). It has slowly become obsolete, without being replaced by a worthy substitute. SDM/S remains an exemplary model for new methodologies.

Zachman's Framework

The Enterprise System Topology is inspired from Zachman's "framework." Here, two criteria are cross referenced: the first corresponds, more or less, to levels of concern, the second to generic questions. As a result, 30 representations are required to describe an entire system. This hefty number is due to the matrix-like format, which is quite characteristic of American methodologies of the 60s & 70s. The Topology has largely reduced this number and clarifies the objectives of the models. The Zachman Framework helps to justify the Topology's empirical nature of its models.

The fundamental idea that was 'borrowed' from the Zachman Framework is the importance of looking at reality from different angles in order to obtain an exhaustive description. This idea is also found in the RM-ODP approach, with its notion of points of view, and also in RUP with its 4+1 views (see after).

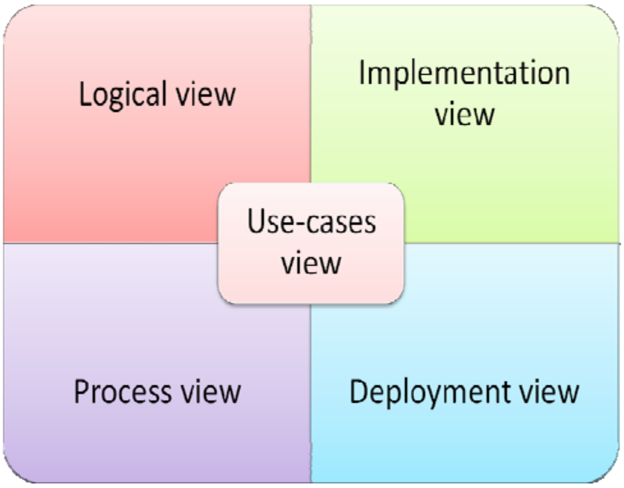
UP and RUP

UP (Unified Process) and RUP (Rational Unified Process) constitute a methodological answer to software development. They are widely known in the IT community. Their principal contribution resides in their original approach to project phasing. Praxeme refers to them especially since they both advocate UML as well as the openness and flexibility expressed by SPEM.

Views and Aspects

Philippe Kruchten proposes a 4+1 approach: logical, implementation, process and deployment, upon which views on use-cases are placed.

Figure SLB-02e.7 Views in Philippe Kruchten's 4+1 approach



If we were to compare this structure with that of the Topology, we would highlight the following points:

1. The logical view includes the Semantic (the "core business" knowledge), Pragmatic (action, organization) and Geographic aspects, to which is also added the Logical aspect, which is the intermediary between a system's reality and the ability to automate it. Such condensation is explained by the nature and scope of RUP – limited to software development, it excludes activities that precede IT development projects.
2. On the other hand, the implementation view corresponds to the Software aspect and the deployment view resembles the description of the Physical aspect.
3. As far as the process[21] view is concerned, its merit lies in the raising of intricate questions associated with the performance of IT systems. In Praxeme, these questions are dealt with progressively across the technical, software and physical aspects.

eXtreme Programming (XP)

eXtreme Programming qualifies itself to be "anti-modeling." This choice of method further limits the methodological reach to only software programming, and this, in the scope of small sized applications and on condition that specific disciplines all cooperate closely with one another[22].

eXtreme Programming introduces very useful methodological elements such as techniques for unit testing, continued integration of developments and even "peer programming" which ensures a relatively good quality of programming. Highly Developer-centric and relying on the availability and continual communication between the Developer and the Business owner, eXtreme Programming is not adapted to large teams or to a subcontractor model.

MDA

The Model Driven Architecture standard constitutes a tool within the method. It lays out the principles behind the separation of aspects. The reference to MDA ensures that our initiative is able to be in line with an extremely important software engineering trend. The methodology and its application will benefit from the techniques and tools adapted for MDA for the transformation of models from one aspect to another (thanks, in part, to UML profile techniques.)

[21] Process, here, is understood in the IT sense of the word, that is to say the execution process of an application.

[22] If we wanted to face the challenges of today's modern organizations, to help them draw up their strategy and put them on the track to industrializing their software development process, it is *eXtreme Modeling* that would guide us along the way.

RM-ODP

"Reference Model – Open Distributed Processing" is the result of a joint effort between the ISO and ITU-T organizations. It has strongly influenced OMG work and is known, internationally, as the ISO 10746 and IYU-T W.900 standards.

This referential, oriented towards IT systems, defines points of views: enterprise, information, computational, engineering and technology.

The RM-ODP approach is used, amongst others, in the telecom industry.

Component-based approach

If we were to quote Peter Herzum[23], logical architecture and design practices are being added to work published under the component-oriented design and development trend. In addition, this approach, which blossomed from real experience, brings numerous details and recommendations to the technical and software aspects. Praxeme completes this approach by demonstrating how to derive components from "conceptual" models.

2TUP

The Two Track Unified Process, from Valtech, is based upon the development lifecycle known as the "Y-model." This model is compatible with how the aspects of the Topology have been organized, which pushes even further the concept of parallelism.

Numerous practices

Whenever it comes down to analyzing, modeling, evaluating or designing an aspect, and where renowned best practices exist and are efficient, Praxeme adopts one of the following attitudes:

- Either the new method draws on existing ones,
- Or, it keeps them as they are and finds them a place among the other types of activities.

The objective is to provide an exhaustive catalogue and a consistent vision, thereby allowing the comparison of elements contributed from different traditions or practices.

A few examples: Balanced Scorecard, TCO, ITIL, QOS, ROC…

The reason for this syncretism is two-fold: it is a question of, on one hand, producing a complete methodological repository as quickly as possible, and on the other hand, to welcome existing practices and help them to increase their usefulness by associating them with others.

Quality

Praxeme intends to merge Quality with Methodology. This unification occurs sometimes in practice, but rarely in theory. The limit of cultural traditions is often rapidly met[24]. Praxeme aims at remediating this phenomenon by consolidating the different contributions, whilst retaining the radical question of Quality.

This subject will be the theme for future developments.

CMM

CMM is not a method. CMM does not only apply to software development.

All the same, Praxeme carefully positions itself with respect to the maturity levels

[23] Peter Herzum, Oliver Sims, Business Component Factory, A comprehensive overview of Component-Based Development for the Enterprise, Ed. OMG Press 2000.

[24] All literature on UML, for example, is blind to the culture of Quality.

defined by this referential and the associated maturity models. It is possible to measure just how much the method can ensure improvements in maturity.

CMM and its competitors drive to enrich the Process component of the methodology.

Conclusion: synthesis of what has been covered in this whitepaper

Praxeme aims at building a complete communication chain from the Board of Directors to employees, via the Organization designer, Designer, IT developer and other profiles. Relying on a global vision of the System (Enterprise system, action system…) the methodology mobilizes experts who, today, simply don't communicate enough with each other. Provoking thought-processes and innovation in organizations and systems is conditioned on establishing a vision of the different aspects. From here, modeling techniques and methods can be revisited and integrated with one another. Another condition is having detailed work instructions to hand to guide the user in its application.

Praxeme covers both strategic and tactical levels.

Appendices

The name of the method

Etymology

- Ancient Greek, “praxis”: action.
- Ancient Greek, “semeïon”: meaning, significance.

Hence, the motto: “Praxeme, meaning in action” (French version: “*Praxeme, le sens*[25] *de l’action*”).

The definition of Praxeme

Scope and intent

Praxeme is an enterprise methodology. That is to say that Praxeme is an attempt to encompass all the aspects of an enterprise[26] or any kind of system. Praxeme emphasizes the communication issues in the context of an enterprise. It aims at linking together the various disciplines we need to mobilize in order to think and transform the enterprise. Numerous opportunities for change and innovation are lost due to lack of synergy between disciplines which are isolated and unable to merge in a united endeavor. The first contribution of Praxeme lies in the comprehensive framework the Enterprise System Topology provides. It fulfills the first condition for a harmonious transformation chain, from strategy to implementation.

Nature

The Praxeme corpus proposes several elements of the Praxeme method; the kind of things the operational staff is seeking at a grassroots level. But, in the first step of its development, Praxeme lays down the foundation with the theoretical basis of the methodological guides. The methodology is not the method; it is the words describing and preparing the method. We need to clearly establish the foundations for the following reasons:

- It constitutes the prerequisite for ensuring the validity of the method.
- It helps to gain support and confidence during the transition phase, when an organization is adopting the new method.
- It positions Praxeme against the available methods such as UP, TOGAF, ITIL…
- It lays the groundwork for scientific contribution.

[25] The French word “sens” has two meanings: a) meaning, b) direction.

[26] We use the “enterprise” term in both senses: an organizational unit of our socio-economical system; a project or venture. Indeed, the methodology applies to various contexts in any kind of sector: business, industry, military, physical…

Appendices (cont.)

For more information

On the Internet

Official site of the Praxeme Institute: http://www.praxeme.org.

For a broader audience: http://www.enterprisetransformationmanifesto.org.

Blogs

- From the creator of the methodology: http://dvau.praxeme.org (French) and http://dvau-en.praxeme.org (English).
- From the Praxeme community: http://friends.praxeme.org.

Available documents

The methodological guides are available for free and can be downloaded on the site.

The structure of the corpus reflects the Enterprise System Topology:

- The General Guide gives an overview of the methodology and justifies the topology.
- Each aspect identified in the topology is the subject of a dedicated guide. For instance, the Guide to the Logical aspect explains the method for SOA, since Praxeme situates SOA as a style for logical architecture.

Pedagogical material is also to be found on the website. Presently, the following courses are available:

- “Praxorama, an overview of Praxeme methodology”.
- “Semantic modeling”.
- “Logical modeling”.
- “Praxeme for SOA” (Extraordinary session, 2007).
- “Extreme Architecture” (Extraordinary session, 2008).

Besides these documents and articles, the Praxeme Institute, along with the actors involved in the initiative, provides operational components such as UML profiles and pre-built models.

An illustration of the use of the method is given in the book “Sustainable Information System” (Wiley, 2009), based on a real Information System overhaul project based on SOA.

Appendices (cont.)

Short Bibliography

Selection

The table below lists the main references which are directly linked with the topics of the white paper. The last columns indicate whether the text is aimed at Managers (Strategic) or operational (Technical) people.

Theme	Title	Reference / URL	Strategic	Technical
Analysis	Club des Maîtres d'ouvrage		✓	
	Chaos Report, Standish Group, 2001	Standish Group	✓	
UML		http://www.omg.org		✓
MDA	*Aligning enterprise architecture and IT investments with corporate goals*, Richard Buchanan (MetaGroup), Richard Mark (OMG)		✓	
	Executive overview, MDA, The Architecture of Choice for a Changing World	http://www.omg.org/mda/executive_overview.htm	✓	
	MDA Guide Version 1.0.1	http://www.omg.org/mda		✓
RM-ODP	« *Reference Model of Open Distributed Processing (RM-ODP): Introduction* », Kerry Raymond			✓
	A Guide for Using RM-ODP and UML Profile for EDOC, INTAP			✓
RUP	Philippe Kruchten, The Rational Unified Process, an introduction	Ed. Addison-Wesley, 1998		✓
Zachman	« The Zachman Framework and the OMG's Model Driven Architecture », Whitepaper	In "Business Process Trends"	✓	
		http://www.zifa.com http://www.zachmaninternational.com		✓
View	*IEEE Recommended Practice for Architectural Description of Software-Intensive Systems*	IEEE Std 1471-2000		✓

Appendices (cont.)

The coverage of concerns

Introduction

The chapter entitled “The Initiative” introduced the purpose of the methodology. The table below indicates responses that Praxeme’s contributions bring to address the concerns. These elements are dispatched across the three dimensions of the PRO^3 scheme. (see page 22).

Figure SLB-02e.8 Coverage of concerns (table)

Concern / Objective	Response on the “Product” dimension	Response on the “Process” dimension	Response on the “Procedure” dimension
Control system complexity	Isolated aspects. Checklists of topics to be examined, decisions to be made. Putting together various disciplines all along the transformation chain. The scale (local/global).	Processes should be grounded upon a rigorous description of the “Product”. Constraints guide the development of processes in the methodology.	Specific modeling techniques, taking into account the intent of the communication.
Reduce costs and stimulate synergies	Precise definition of repositories, allowing a high level of reuse. Clear links between local models and the repositories.	Clarifying and dispatching responsibilities along several axes. Adapting the organization towards excellence and innovation.	Identifying the required disciplines and specifying the deliverables. Automation (see below).
Restore clarity in organizations	Enterprise System Topology. Incorporation of various viewpoints. A means to better integrate and link together the many disciplines in the organization.	A unified transformation chain, ensuring consistency and vision, from strategy to implementation.	The focus on communication issues.
Elaborate precise techniques	The metamodel provides the rigorous analysis of all scoping and modeling elements. In addition, salient notions such as the scale and the analysis/design dichotomy, as well as measurement, make it possible to elaborate detailed procedures and methods[27].	Definition of the disciplines which are involved in the transformation chain. Specification of skills. Introduction of variants (multi-scenarios approach).	Many proposed procedures and methods, some of them are generic (e.g. modeling), others are specific (e.g. business process modeling). Structuring the disciplines is part of skill management.
Automate	Derivation rules from an aspect to another, in accordance with the Enterprise System Topology.	Metamodel and Transformation model.	MDA standard; UML profiles associated to specific activities.

[27] Every aspect is subjected to quantitative evaluation, either economical or technical. Among the evaluation techniques, we find: audit, metrology, cost evaluation, optimization…

Appendices (cont.)

Glossary

Architecture

The “architecture” term relates either to something to be built or to the art of designing.

Product A specific architecture is a collection of well-argued choices and building rules regarding an aspect of a system.

Some features of an architecture according to Praxeme methodology:

- A specific architecture is a model, the scale of which is global. It deals with the system as a whole.
- With respect to the limit of each discipline, a specific architecture considers only one aspect, as defined in the Enterprise System Topology. Therefore, we use the term to designate an “enterprise architecture” (upstream aspects), “logical architecture”, “hardware architecture”, “software architecture”, “technical architecture”, “physical architecture”.
- There is a continuum between the architecture of a system (global scale) and a model (local scale).

Art Architecture is the art of designing a construction. In the scope of the enterprise methodology, the construction is the enterprise itself or any of its inner systems. The main concern of architecture is structure, implying the qualities and behaviors of the system on the long term.

Aspect

Roughly speaking, there are two ways to cope with complexity: the first one is to break down the object into smaller parts, until finding elements which are easy to understand and to manipulate; the second one is to distinguish between sets of features which can be explained separately. The first way is known as the Cartesian approach and can apply as far as the system displays a unique and homogeneous nature. The second approach appears in system theory. Regarding the enterprise, the complexity stems from the merging of various concerns, the conflict between divergent interests and the cohabitation of extraneous cognitive universes. Therefore, we have to recognize the plurality of this reality.

Reality must be approached from several angles: these angles correspond to aspects in the Topology of the Enterprise System. An aspect is part of the System. This part reveals itself as a set of elements, closely related, focused on a specific concern. The elements differ from one aspect to another so that the aspects don’t overlap.

Unlike the View (see below), the aspect is a notion which doesn’t refer to an actor considering the system (viewpoint), but which entirely belongs to the “instance”. This metaphysical standpoint can be easily challenged but we stick to it because it brings efficiency in representation and action.

Component (of the method)

The method is made up components that respect the salient issues that determine its structure. The architecture of the methodology takes into account requirements from theoretical, practical and didactical points of view. The Praxeme website gradually reflects this structure.

Discipline

A discipline aggregates know-how, procedures and methods, into a unit of skills based upon a generic knowledge. This notion is very close to the notions of craft and art (art of engineering, art of architecture).

Quantitative limits in human intellectual and operational abilities determine the potential discipline.

Examples of disciplines: modeling, project management, strategic design, documentation management…

Enterprise System

"Enterprise System" is the generic designation of the field in which the methodology applies.

"Enterprise" can be understood in both meanings: a) an organization or business; b) a project, a venture.

What is the difference between enterprise and Enterprise System? The latter expresses the effort of rationality. It refers to the enterprise insofar as it endeavors to analyze all its dimensions and to think itself in the full light of reason.

Facet

Equivalent to "aspect."

Metamodel

A metamodel is a model of models, i.e. a formal representation of how ordinary models should be built. It clarifies the categories we use when modeling. "Category" is used here in his Kantian meaning.

Establishing the metamodel is a healthy exercise which allows the setting of the groundwork for the methodology. Moreover, the metamodel can be used as a model for the tools we need to equip the transformation chain. For instance, Praxeme metamodel specifies the UML profiles dedicated to the different modeling disciplines.

Method

A prescription of a how-to-do.

Methodology

Methodology is the words describing or the study of the method, i.e. the cultural background and the way of thinking that ensures the efficiency of methods.

PRO^3

"PRO3" (Pro-cubed) is the formula which sums up the three chapters of methodology – as the science and study of the method –, i.e. Product, Process, Procedures. Here, "Product" designates all kind of objects we want to examine, design, build or transform, from small units as software component through to the enterprise itself, across all its aspects. Methodology obviously deals with processes which are a way for organizing collective activities, while procedures are the way to optimize individual activities.

Scale

The notion applies to the Enterprise System and can also qualify the scope of an activity. We oppose local and global scales but the notion can be extended as a continuum of various units and aggregates: operational unit, department, company, group, federation, activity chain including partners, society…

System

We use "system" in its usual sense but bear in mind the findings of the theory of systems, if not only to benefit from its rigor. We compensate the pretence of the theory by referring to Henri Poincaré's quotation: "Systems only exist in the human mind." System and the notions around it are intellectual tools we use when coping with reality and its complexity.

Transformation

The purpose of the enterprise methodology is to help transform the enterprise. Transformation is opposed to execution. The notion covers every activity which examines and analyses, describes or prescribes, designs and builds the enterprise or part of it.

Transformation chain

The transformation chain is the unified concept of the collection the activities link together and whose purpose is the enterprise transformation.

Transformation chain

Also: “activity chain”, “transformation”

The scope of Praxeme is the transformation chain which aims at building or transforming the Enterprise System. This chain includes:

- Formulating the strategy, which defines the goals and expresses values and purposes.
- Capturing the basic knowledge (semantic aspect).
- Analyzing and designing the business processes and the organization.
- Software development and maintenance.
- Managing, consolidating and optimizing the information system.
- Integration, audit and governance.

View

A partial set of information or knowledge about a reality, a view always refers to an actor or a given kind of actors. This notion differs from the one of aspect. Some views are made up of information, decisions, elements which come from various aspects.

Index

A

activity chain · 37
ambition · 23
analysis · 26
Architects · 19
architecture · 35
Architecture · 25
aspect · 16, 35
aspects · 15

B

Balanced Scorecard · 29

C

CMM · 7, 29
component · 35
costs · 9

D

Descartes · iv
design · 26
Designers · 19
Directors · 30
discipline · 35
disciplines · 16, 17

E

Economic evaluation · 25
engineering · 25
Enterprise
    Enterprise Architecture · 6, 23
Enterprise Architecture · 25
enterprise methodology · 31
enterprise system · 15
Enterprise System · 14, 18, 36
eXtreme Programming · 28

F

framework · 11, 17

H

Herzum · 29

I

IEEE · 33
Infrastructure · 25
ISO · 7, 8, 24
ITIL · 24, 29, 31

K

Kant · iv
knowledge · 12
Kruchten · 27, 33

L

licence · iii

M

MDA · 10, 17, 28, 33
Merise · 7, 27
metamodel · 22, 34, 36
Modelers · 19

O

Object-oriented · 10
organization · 5, 9, 13
Organization designs · 19

P

Poincaré · 36
Praxeme · 31
Praxeme Institute · iii, 6
Pro cubed
    PRO3 · 36
PRO cubed
    PRO3 · 22
process · 9

Q

QOS · 29
quality · 20
Quality · 29

R

RAD · 7

reengineering · 9
repositories · 17
repository · 21, 23
Repository · 19
RM-ODP · 24, 29, 33
ROC · 29
RUP · 24, 27, 28, 33

S

SA/SD · 12
scale · 18
SDM/S · 12, 27
SPEM · 22, 24, 27
strategy · 5, 18, 31, 34, 37
System · 14

T

target · 18
TCO · 29
TOGAF · 31
topology · 12, 14, 15, 16
Topology · 21
transformation · 37
Transformation · 22, 36
transformation chain · 37
typology of activities · 25

U

UML · 7, 10, 19, 20, 23, 33
UP · 27, 31

V

Valtech · 29
view · 37
View · 33

Z

Zachman · 27, 33