



Katholieke *Universiteit* Nijmegen



ORDINA INSTITUTE *for Research and Innovation*

Towards an Information Systems Engineering Body of Knowledge

White Paper

Towards an Information Systems Engineering Body of Knowledge

H.A. Proper*

Ordina Institute
Groningenweg 6
2803 PV Gouda
The Netherlands
E.Proper@acm.org

H. Bosma

Ordina Institute
Groningenweg 6
2803 PV Gouda
The Netherlands
H.Bosma@institute.nl

S.J.B.A. Hoppenbrouwers

University of Nijmegen
Sub-faculty of Informatics
IRIS Group
Toernooiveld 1
6525 ED Nijmegen
The Netherlands
StijnH@cs.kun.nl

R.D.T. Janssen

Ordina Institute
Groningenweg 6
2803 PV Gouda
The Netherlands
R.Janssen@institute.nl

April 17, 2001

PUBLISHED AS:

H.A. Proper, H. Bosma, S.J.B.A. Hoppenbrouwers, and R.D.T. Janssen. Towards an information systems engineering body of knowledge. In *Proceedings of the Second National Architecture Congress*, Amsterdam, The Netherlands, November 2000.
<http://www.architecture-forum.org>

Abstract

Though there may be millions of professionals worldwide acting as a designer, architect, or engineer in the design, realisation, and implementation of information systems, there is not yet a well established and clearly identified body of knowledge that can be said to define the profession.

In this article, we present the idea of developing an information systems engineering body of knowledge. Such a body of knowledge could play a pivotal role in the further professionalisation of our discipline. The approach we take is an organic approach in which we first aim to gather a library with significant knowledge, demarcating the field, and then move on to integrate this into a consistent body of knowledge.

We also realise that this effort can not be done in isolation. This article should therefore also be regarded as an invitation for additional participants.

1 Introduction

Even though there may be millions of professionals worldwide who act as a designer, architect, or engineer in the design, realisation, and implementation of information systems, the field of information systems engineering has not yet reached the full status of a recognised profession. With the term *information systems engineering* we actually refer to all activities involved in the design, realisation and implementation of information systems. This essentially ranges from the high level design of a portfolio of information systems (an information architecture) to the design and realisation of a specific information system.

The Software Engineering Body of Knowledge (SWEBOK) project [12] started out from a similar observation on the field of software engineering. In [12] the following motivation of the SWEBOK project is provided:

In other engineering disciplines, the accreditation of university curricula and the licensing and certification of practicing professionals are taken very seriously. These activities are seen as critical to the constant upgrading of professionals and, hence, the improvement of the level of

*Also: University of Nijmegen, Sub-faculty of Informatics, Information Retrieval and Information Systems Group, Toernooiveld 1, 6525 ED Nijmegen, The Netherlands,

professional practice. Recognizing a core body of knowledge is pivotal to the development and accreditation of university curricula and the licensing and certification of professionals.

It is our belief that this motivation not only applies to software engineering, but equally well applies to the field of information system engineering. The reason being that the field of information systems engineering is a field that is still under development, and is slowly gaining the appreciation it deserves as a distinct discipline.

Two key differences between information systems engineering and software engineering are:

- Information systems engineering exclusively focuses on systems handling information, while software engineering has a broader focus in this sense.
- Software engineering focuses on software, and as such, limits itself to computerised (information) systems only. Information systems engineering is therefore broader in the sense that it is not only focussed in computerised systems, but includes manual aspects of information processing as well.

For information systems engineering, the business, cultural, human and organisational context is just as important as the automated parts of an information system.

The latter difference makes it also much harder to clearly demarcate the information systems engineering field in comparison to software engineering.

We feel that the development of an explicit and accepted body of knowledge for the information systems engineering profession would, for example:

- Allow Universities and training institutes to tune their curricula to a well defined body of knowledge accepted by both industry and academia.
- Allow for the identification of distinct roles in the information system engineering process and related forms of certification.
- Allow managers of information systems engineering projects to constitute project teams with professionals who share a common terminology and understanding of the profession.
- Allow client organisations to organise second opinion reviews among providers of information system engineering.
- Allow for re-use of experiences and materials between practitioners, preventing them from having to re-invent the wheel over and over.

In line with [12], we define the body of knowledge for information systems engineering to be an all-inclusive term that describes the sum of knowledge within the profession of information systems engineering.

Three other examples of, ICT related, bodies of knowledge with ensuing standardisation, certification and training activities are:

1. The IT Infrastructure Library (ITIL) for IT infrastructure management [7].
2. The Project Management Body of Knowledge (PMBOK) for project management [10].
3. The Information Services Procurement Library (ISPL) for procurement of information services [13].

We mention these examples to illustrate the importance of having such a body of knowledge. ITIL, at least in the Dutch context, has evolved into a well-known standard work on IT infrastructure management. The project management body of knowledge may not be that well-known in the ICT communities, however, it appears to be a well-accepted standard for project management certification in the United States of America. ISPL has not yet reached the same status as ITIL already has, although the intention of the European Union (the main sponsor of the project leading to ISPL), is to make it into a standard body of knowledge for procurement of information services.

In our opinion, having an "Information Systems Engineering Body of Knowledge (ISEBOK)", would add to the maturity of our profession. In this paper we report on a project in progress, currently involving Ordina and the University of Nijmegen, that aims to develop an initial and modest form of such a body of knowledge in terms of a library.

2 Towards an information systems engineering body of knowledge

To develop an ISEBOK one first needs to demarcate the discipline. In other words, sooner or later we will have to identify clearly what is considered to be part of the discipline and what is part of neighbouring disciplines. It will, however, be extremely hard to demarcate this emerging discipline beforehand. It is well known that non ICT aspects, such as:

- business,
- stakeholder management,
- project management,
- procurement,
- human resource management,
- change management,
- ...

need to be taken into account when engineering information systems. In the present state of this field, it will therefore be nearly impossible to identify clearly what should be part of the core and what should be part of its neighbouring fields of study.

2.1 A library-based approach

Given the unclear boundaries of the information systems engineering field, we propose to use a more organic approach to the development of an ISEBOK. Organic in the sense that we will not put a strong restriction on the thematical scope of the contents of the body of knowledge beforehand, but rather let the scope emerge empirically from the contents as more is added. Added to this comes the fact that most of the available knowledge on information systems engineering is still in a case-based form. In other words, the knowledge is very much tied to experiences gleaned from specific cases from industrial practice.

The only deliberate and explicit demarcation, therefore, lies in the group of professionals responsible for providing the contents: those people who consider themselves to be information systems engineers. The bottom line, then, lies with the definition of *information system*, which, in line with [9], we take to be the following:

An information system is a sub-system of an organisational system, comprising the communication- and information-oriented aspects of the organisation system.

It is also not our initial goal to develop a fully integrated and completely consistent body of knowledge for information systems engineering. Our first aim is a more humble one. Initially, we ‘simply’ aim to compile and structure a library filled with a collection of significant, and loosely coupled, ‘knowledge items’ considered useful to project members of an information systems engineering project. These knowledge-items will consist of focussed denotations of knowledge pertaining to information systems engineering. More specifically, the library will contain:

- Knowledge that will aid project actors involved in an information systems engineering project in the *planning* and *execution* of this project.

Examples of this kind of knowledge¹ would be:

- Project strategies.
- Case studies (both successes and failures)
- Descriptions of methods, techniques and tools.

¹ Some of these items may actually be regarded information rather than knowledge. However, in this article we shall use knowledge as a collective term for any information or knowledge that is relevant to an information systems engineering project.

- Reference models and frameworks.
- Heuristics on the use of project strategies, methods, techniques, etc.
- Heuristics and guidelines with regards to design decisions, the use of reference models, architecture principles, etc.

The knowledge itself may be communicated in different forms. For example:

- Reference material.
 - (Self-)study material.
 - Teaching material.
- A classification of the available knowledge in terms of some suitable characterisation mechanism.
 - Guidelines to select relevant knowledge sources from the library, given a specific project situation and task at hand.

The intended audience of the library includes managers, architects, designers, engineers, etc. In the next section, we will see how we intend to dissect this knowledge into focussed knowledge items.

Once such a loosely coupled library of knowledge items has been gathered, we should be able to better demarcate the field of information systems engineering and start with the development of an integrated and consistent body of knowledge.

2.2 Purpose of the library

Even though the library serves the longer term purpose as a base to derive an information systems engineering body of knowledge, the library will in its initial form already be useful to project members of an information systems engineering project. Project members should be able to use the library to find the answer to such questions as:

- Which activities, techniques and tools are most appropriate given a specific project situation and task?
- What reference models are relevant in a situation where the business has selected "customer intimacy" [14] as their strategic focus?
- What are relevant design options in a given situation?
- What material is available to teach someone a given technique or method?

2.3 Related work

Some related work may be found in e.g. [2] and [9]. In [2], a model curriculum is described for undergraduate degree programs in information systems and, as such, also contains a definition of a body of knowledge for information systems engineering. The Framework for Information Systems Concepts (FRISCO) as reported in [9] aims to give the field of information systems a conceptual underpinning by introducing a unified framework of concepts. What both of these approaches lack is the practical side in terms of concrete work practices, techniques and tools to be used, etc.

Another field that is of relevance is the field of method engineering [1, 6]. The approach taken in this field, however, tends to dissect methods to the level of distinct concepts and their relationships. For the moment, we consider this to be far too detailed for our effort of gathering and structuring a representative library of relevant knowledge items. However, the theories provided by method engineering are useful in dissecting methodological knowledge into more elementary items (see the next section) and will be of even more use when developing an actual consistent body of methodological knowledge once we have gathered a significant collection of knowledge items into our library.

3 Contents of the library

The contents of the library will mainly be of a methodological nature. This makes it worthwhile to take a closer look at the structure of a method, as the methodological knowledge as it will be stored as knowledge items in the library may be dissected according to this structure.

3.1 The structure of a method

In literature on method engineering, frameworks may be found that basically provide an anatomy of a method [3, 15]. For our purposes of building a library of methodological knowledge for information systems engineering, we have made a combination of the two frameworks as reported in [3, 15].

This has led to the following structure of a method:

- An *execution plan* focuses on the *what* and *when* issues of a method. It mainly deals with the managerial aspects of information systems engineering and includes such aspects as human resource management, quality and progress control, resource planning, contract management, evaluation of plans, etcetera (see [8] and [11]).

Typical elements in an execution plan are:

- Identification of activities/steps needed.
 - Actor types and related skills/disciplines.
 - Deliverable specifications and associated planning.
- The *how* and *what with* questions of a method are the focus of the set of *tools & techniques* to be used to produce the deliverables as specified in the execution plan.

Typical sub-components are:

- Notation specification.
- Task descriptions.
- Design artefacts.
- Tools supporting the different notations and execution of tasks.

The notation specification basically defines an abstract language in which to express models and is usually provided as an abstract description of the underlying modelling concepts together with their interrelationships and properties. It structures the models which can be used in information systems engineering, i.e. it provides an abstract language in which to express the models.

The task descriptions structure the way in which an information system is to be engineered. They define the possible tasks, including sub-tasks, and ordering of tasks, to be performed as part of the engineering process. They furthermore provide guidelines and suggestions (heuristics) on how these tasks should be performed.

Design artefacts refer to ready-to-use design elements, such as reference models, templates, etc.

3.2 Specifying methodological knowledge

As the intention is to fill the library with methodological knowledge, this knowledge needs to be made explicit. At the moment we will mainly rely on natural language to express this knowledge. We do not (yet) have the ambition to model this knowledge in terms of some formal knowledge representation language.

When specifying methodological knowledge, one may actually do this at different levels of concreteness. For example, the following heuristic:

If the complexity of the information to be stored in an information system is high, then it is advisable to use a natural language based approach for information modelling.

is of a higher level of abstraction than the precise description of the tasks involved in doing a natural language based information analysis such as may be found in e.g. [5]:

- **Step 1:** From examples to elementary facts.
- **Step 2:** Draw fact types, and populate.
- **Step 3:** Trim schema; note basic derivations.
- ..

In our view, there are two *extreme* levels of concreteness at which methodological knowledge may be described:

- **Strategic**
High-level, strategic heuristics, guidelines, architecture principles, etc. concerning the execution plan. For example: linear versus evolutionary approach, level of user participation.
- **Operational**
Specific, precise task descriptions, meta-models of notations, etc. concerning the operational side of the project. For example: modelling steps, completeness and quality checks.

We now have two axes along which we could roughly position methodological knowledge in the library:

1. the methodological aspects it pertains to,
2. the concreteness level at which it is described.

These axes have been depicted in figure 1. Note that *strategic* and *operational* are used here as perspectives on the information systems engineering *project* and not on the (portfolio of) *information systems*.

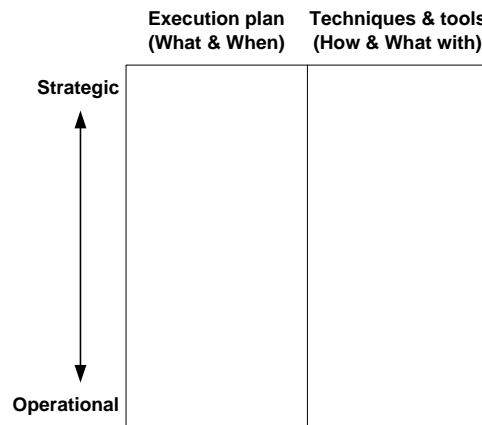


Figure 1: Two axes along which to position methodological knowledge

4 Characterising methodological knowledge

A first way to characterize methodological knowledge is provided by the different methodological aspects as discussed in the previous section. However, methodological knowledge may be characterised along other axes as well. In this section we discuss some of the possible additional axes. At present we are not yet certain which axes are the most suitable ones.

- *Communication style.*
What is the communication style used in describing the methodological knowledge? Some examples are: reference material, study material and teaching material.
- *Language used.*
The language used to express the methodological knowledge. Examples would be: Dutch, English, American, German, Frisian, Gaelic, French, etc.
- *Level of abstraction.*
What level of abstraction, with respect to the systems under consideration, is the methodological knowledge focussed at? Examples, derived from [4], would be: conceptual level, logical level and physical level.
- *Systems scope.*
What is the scope of the methodological knowledge in terms of the system(s) considered? Some examples would be: portfolio of systems, family of systems, specific system and specific use-case.
- *Systemic focus.*
What system is the methodological knowledge focussed at? Some possible systems are: organisational system, information system, computerised information system, infrastructural system.
- *Systemic aspect.*
In [16], Zachman identifies a number of different aspects of a system, based on the interrogatives of the English language. This leads to aspects such as: what, how, when, why and where.

5 Discussion

The aim of this paper was to present the idea of developing an information systems engineering body of knowledge. We have discussed our motivations, our initial goals in terms of a library, as well as a preliminary structure for the library.

As this project constitutes a collaboration between Ordina and the University of Nijmegen, we are currently in the process of gathering information systems engineering knowledge within the context of Ordina. However, we would like to stress the fact that this article should also be regarded as an invitation for additional participants to join our effort in gathering and structuring an information systems engineering body of knowledge. An effort that can only be done in collaboration with many colleagues from industry and academia.

References

- [1] S. Brinkkemper, K. Lyytinen, and R.J. Welke, editors. *Proceedings of the IFIP TC8 WG8.1/8.2 Working Conference on Method Engineering*. Chapman & Hall, Atlanta, Georgia, August 1996.
- [2] G.B. Davis, J.T. Gorgone, J.D. Couger, D.L. Feinstein, and H.E. Longenecker, Jr., editors. *IS'97 – Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems*. ACM, AIS and AITP, 1997.
- [3] L. Fokkinga, M.H. Glastra, and H. Huizinga. *LAD - Het lineair ontwikkelen van informatiesystemen (In Dutch)*. Academic Service, Schoonhoven, The Netherlands, 1996.
- [4] J.J. van Griethuysen, editor. *Concepts and Terminology for the Conceptual Schema and the Information Base*. Publ. nr. ISO/TC97/SC5/WG3-N695, ANSI, 11 West 42nd Street, New York, NY 10036, 1982.

- [5] T.A. Halpin. *Conceptual Schema and Relational Database Design*. Prentice-Hall, Sydney, Australia, 2nd edition, 1995.
- [6] A.F. Harmsen. *Situational Method Engineering*. PhD thesis, Twente University, Enschede, The Netherlands, 1997.
- [7] A code of practice for it service managers, 1998.
- [8] F. Kensing. Towards Evaluation of Methods for Property Determination: A Framework and a Critique of the Yourdon-DeMarco Approach. In Th.M.A. Bemelmans, editor, *Beyond Productivity: Information Systems Development for Organizational Effectiveness*, pages 325–338. North-Holland, Amsterdam, The Netherlands, 1984.
- [9] P. Lindgreen, editor. *A Framework of Information Systems Concepts*. Interim Report of the IFIP WG 8.1 Task Group FRISCO, 1990.
- [10] Project Management Body of Knowledge, Last verified on: 29th of August, 2000.
<http://www.pmi.org/publicctn/pmboktoc.htm>
- [11] H.G. Sol. Information Systems Development: A Problem Solving Approach. In *Proceedings of 1988 INTEC Symposium Systems Analysis and Design: A Research Strategy*, Atlanta, Georgia, 1988.
- [12] Software Engineering Body of Knowledge, Last verified on: 29th of August, 2000.
<http://www.swebok.org/>
- [13] The ISPL Consortium, editor. *Information Services Procurement Library – The Introduction*. ten Hagen & Stam, Den Haag, The Netherlands, 1999.
- [14] M. Treacy and F. Wiersema. *The Discipline of Market Leaders - Choose your customers, narrow your focus, dominate your market*. Addison-Wesley, Reading, Massachusetts, 1997.
- [15] G.M. Wijers and H. Heijes. Automated Support of the Modelling Process: A view based on experiments with expert information engineers. In B. Steinholz, A. Sølvberg, and L. Bergman, editors, *Proceedings of the Second Nordic Conference CAiSE'90 on Advanced Information Systems Engineering*, volume 436 of *Lecture Notes in Computer Science*, pages 88–108, Stockholm, Sweden, 1990. Springer-Verlag.
- [16] J.A. Zachman. A framework for information systems architecture. *IBM Systems Journal*, 26(3), 1987.