IS Architecture Assessment for Requirements Engineering

Erling S. Andersen & Andreas L. Opdahl
Department of Information Science
University of Bergen, Norway

Abstract

An information system (IS) architecture *comprises the computerised ISs of an enterprise as well as their interconnection structure and intercommunication methods. Requirements engineering work must 1) capture the appropriate requirements for a projected computerised information system in isolation and 2) ensure that the projected IS will conform to an explicitly expressed enterprise-wide IS-architecture strategy.*

The RAISA ("Representation, Assessment and Improvement of Information Systems Architecture") project aims at providing methods and description techniques for determination of IS-architecture strategies and hence for evolution of IS architectures. This paper reports on the initial stages of the RAISA project from the viewpoint that contemporary requirements engineering methods will become inadequate in the future unless they show increasing concern for IS-architectural issues.

1 The need for IS architecture assessment

Focus on IS architectures as part of requirements engineering work has a long tradition: Davis [5] points to "two levels at which information requirements need to be established in order to design and implement computer-based information systems:

1. The organizational information requirements to define an overall information system structure and to specify a portfolio of applications and data bases.

2. The detailed information requirements for an application."

More recently, Flaatten [7] has pointed out that the traditional systems development life-cycle is not sufficient for the 1990s, but that additional architectural and strategic life-cycles are needed. The strategic life-cycle "covers the systems plans of an entire organization or at least a strategic business unit," and is therefore similar to Davis' concept of "overall information system structure" and to the concept of IS architecture used in this paper.[1]

[1] Hence our concept of IS architecture is *not* related to the term "software architecture" (e.g., Shaw et al. [8].) More specifically, software architecture research aims at identifying generic structures *within* individual information systems, while work on information system architectures deals with single ISs as black boxes, focussing instead on the connections and communication methods *between* them.

The importance of IS architecture work is currently growing because of two trends in the way in which contemporary enterprises are using information systems. First, a major part of the enterprises' information processing has become or is becoming computer-supported. As a consequence, individual ISs are no longer separated by as much manual information processing as before, and large productivity gains can result from a tighter integration between previously separate applications. Second, a new generations of ISs has been introduced to profit from coordinated utilisation of data from previously unrelated systems, prominent examples being integrated production control systems. Both trends serve to pinpoint the need for increased information exchange between the computerised information systems of an enterprise.

1.1 IS architecture problems

The prevailing situation today is therefore enterprises which process information both manually and with the assistance of a diverse set of computerised ISs of different generations. In many cases, their IS architectures have been allowed to evolve in a more or less unmanaged way. Local integration needs have resulted in the — often ad hoc — construction of "bridges" of automated data exchange between individual ISs, which have been modified on demand to facilitate message passing, use of blackboards, exchange of files, or database sharing. In many respects, the resulting random IS architectures resemble the early "spaghetti programs" developed before the days of structured programming. They also share many of the same weaknesses [3]:

- it is difficult to get an *overview* of the enterprise's IS architecture;
- it is difficult to *comprehend* the IS architecture's relation to the business objectives and technological infrastructure of the enterprise;
- it is difficult to *assess* and *improve* the IS architecture in an organisational and technological context;
- it is difficult to *modify* the IS architecture in response to changing business needs and new technological opportunities;
- it is difficult to *maintain* the IS architecture as the enterprise evolves more and more rapidly in response to changing markets.

1.2 IS-architecture philosophies

Other, mostly larger enterprises haven taken a more deliberate approach by adopting idealised approaches to IS architecture work — called *IS-architecture philosophies*, in this paper — from literature. Magoulas & Pessi [11] have described three philosophies: The "information resource management" philosophy (Nolan [13], as well as [10]) advocates centralised information storage, processing and responsibilities, in contrast to the decentralised "function-based management" philosophy [14, 15]. In addition, "inter-organisational systems" are considered. Allen & Boynton [1] discuss a decentralised "low road" versus a centralised "high road" philosophy. Goldkuhl et al. (e.g., Goldkuhl, Petterson & Eriksson [9]) have presented a series of case studies comparing the philosophies.

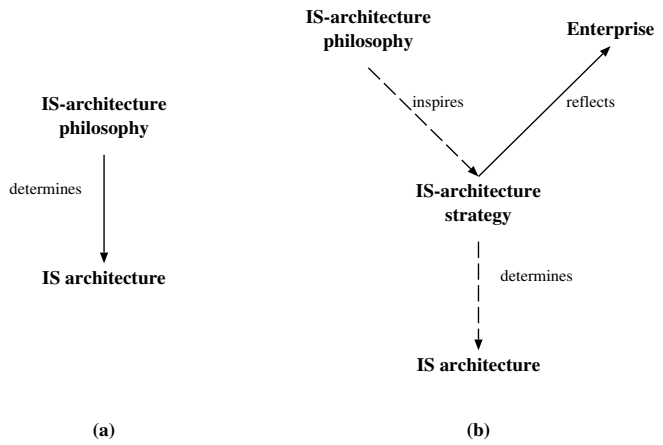

Figure 1: An enterprise's IS architecture should not be directly determined by an idealised IS-architecture philosophy (a), but indirectly through an IS-architecture strategy which reflects the enterprise's particular needs (b).

However, although success stories have been reported, the "off-the-shelf" philosophies adopted from literature or promoted by consulting firms are often not appropriate for an enterprise for several reasons:

- *Particularities of the enterprise:*
  Adoption of an "off-the-shelf" philosophy does not take particular characteristics and needs of the enterprise sufficiently into account: The small number of IS-architecture philosophies available is not likely to cover the needs of all enterprises. Important issues which may be ignored range from organisational culture and style of leadership, down to investments in existing information technology and information systems.

- *The need to combine philosophies:*
  The choice between a decentralised and a centralised organisation is often dependent on problem area and type of application. E.g., administrative systems for pay-rolling and personnel management may benefit from centralisation, while a management information system should be decentralised.

- *Problems of scale:*
  Most philosophies have been developed with large or very large enterprises in mind. It is not obvious that they are easily applicable to medium-sized or small businesses. This is particularly true of philosophies developed by consulting firms (e.g. [10, 6].)

A method for selection among and adaption of existing IS-architecture philosophies is therefore needed. The method should guide the enterprise in creating and implementing its own, particular IS-architecture strategy, and must be based on a broad view of the IS architecture problem, incorporating conventionally separate issues ranging from overall business objectives down to the technology level. Figure 1 depicts this revised view: Instead of letting an IS-architecture philosophy determine the IS architecture of an enterprise directly (a), IS-architecture strategies are needed as buffers

between the two levels to reflect particular enterprise needs (b). The IS-architecture strategies are in turn inspired by existing philosophies.

2 The RAISA project

The RAISA ("Representation, Assessment and Improvement of Information Systems Architecture") project was initiated at the University of Bergen in the spring of 1994. The project studies IS architectures and IS-architecture philosophies from a practical and operational viewpoint. It does not aim at selecting or introducing any new or "better" IS philosophy, but rather at aiding the enterprise in the *process* of tailoring an IS-architecture strategy to its own needs. Also, RAISA augments existing work on IS architectures by taking a broad view on the factors determining the appropriate IS-architecture strategy for a given enterprise. In particular, RAISA explicitly takes the enterprise's mission, objectives, activities and organisation structure, as well as its current IT infrastructure into account when reviewing an IS architecture. The RAISA project currently employs two staff members and six graduate students at the Department of Information Science.

2.1 The RAISA metamodel

At the heart of the RAISA project is a *metamodel* — i.e., a model of models — for IS architecture representation. The metamodel defines a set of basic concepts and relations that have been selected from literature, from discussions with IT managers, and from experiences with preliminary case studies. Apart from being a conceptual foundation for the project, the metamodel forms a hypothetical future, integrating description technique for IS architecture work. It is *not* intended as a presentation model in its own right. Instead it should be used as a repository for information which can be collected and displayed using existing, partial, visual description techniques, e.g., for organisation structures, IT infrastructures. As part of the project, four case studies are currently carried out within large and medium-sized enterprises in the private and public sectors. Hence, the metamodel also directs and coordinates the case-study efforts.

Figure 2 depicts the structure of the current RAISA metamodel. Key concepts have been selected at each of the four enterprise levels, and important relations identified. At the organisation level, the "mission" and "business objectives" describe the overall purposes of the enterprise, which are in turn reflected in "activities." The operational organisation structure is represented as "organisational units" and "subunits" which may exchange information through "communication paths". The organisation structure further determines which "functions" each "computerised information system" must support and which "communication protocols" and (shared) "databases" the IS architecture must provide. The IT level imposes restrictions on which IS architectures that are feasible by representing the interconnection structure of the enterprise's "computing equipment" and "networks". The metamodel has been elaborated on in [3].

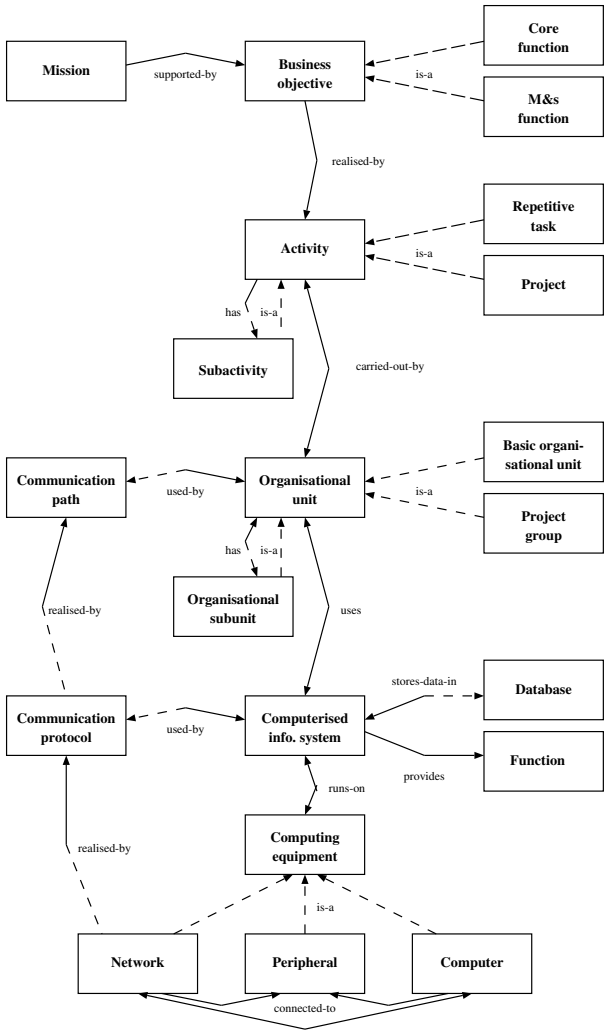

Figure 2: The current state of the overall RAISA metamodel. The model represents the main entities and relations to collect information about when reviewing an IS architecture.

2.2 The RAISA method

The project also outlines a RAISA method for IS architecture work. Its main stages are:

1. *Describe the current IS architecture.*
   Representation (the "R" in RAISA) using the RAISA metamodel is the first step of the RAISA method. The stage acknowledges that IS architecture work is (except when designing new enterprises from scratch) always carried out in the context of past investments in information systems and information technology equipment, as well as existing activity and organisation structures, which must be taken into account for IS architecture improvement to be feasible and fruitful. In addition, the IS architecture representation is needed in later stages for problem assessment and improvement formulation.

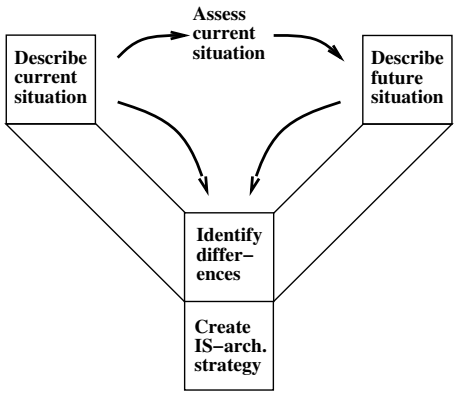

Figure 3: The overall structure of the RAISA method: Current and desirable future situations are described, compared and used for plan formulation.

2. *Assess the current IS architecture.*
   Assessment (the "A" in RAISA) is undertaken next to identify strengths, possibilities, and — in particular — weaknesses of the current situation. This is done partially by interviewing IT managers and users in the enterprise and partially by inspecting the IS architecture representation itself. An important objective for the ongoing case studies is to identify and validate "rules" about the well-formedness of IS architecture representations. An initial set of such rules has been taken from literature, e.g.,

   - "A decentralised operative pattern will ... put great demands on the central boundary conditions and IT-infrastructural standards." [4]
   - "Changing individuals and/or fixing technical features will have little effect on resistance" (towards introduction of management information systems.) [12]

   Currently, "rules" such as these act as hypotheses for the case studies. If validated, they may later serve as context-dependent *guidelines* for IS architecture assessment in other cases too. Only rarely can validated rules be regarded as universal and formal rules, however.

3. *Describe the future IS architecture.*
   Improvement (the "I" in RAISA) is based on the formulation of an improved future situation using the RAISA metamodel. From the differences between current and future situation, a plan to gradually improve the enterprise's IS architecture is formulated in terms of an IS-architecture strategy.

At each stage, more detailed method steps are provided. Figure 3 depicts this mode of work, inspired by the "Y-model" [2].

Coordination responsibilities: To what extent should the central IT department have the power to impose communication standards and data exchange conventions between organisational units within the enterprise?
Communication method: Should the ISs of the enterprise communicate through message passing, blackboards, shared files or common databases? Should the issue be standardised at all?
Communication structure: Should the IS architecture strive for a centralised or a network communication structure? Should the communication structure at all be regulated?
IT architecture: Should the IT infrastructure be as centralised as possible (i.e., a mainframe solution), as distributed as possible (i.e., a workstation solution), or should some intermediate solution be sought?
Data storage: To what degree should the enterprise accept data duplication? Should duplication be allowed in general, be allowed only for certain types of data, or be unacceptable as a rule?
Data localisation: Should the enterprise's data be kept all in a central database, or should each functional unit manage its own database or set of databases? Should centralisation be physical or only logical, i.e., implementing a (logically) centralised database in terms of a transparent, distributed database management system?
Schema integration: Should each of the enterprise's functional units develop their own schemas, or should there be a global, integrated schema for the whole enterprise? Should a multi-level strategy be adopted?
Development responsibilities: Should development of new information systems be the responsibility of a central IT function, or of the local units which will use the ISs? Should enterprise-wide ISs be centrally developed, or should development be delegated to the units with only a central integration effort?
Maintenance responsibilities: Should the responsibility for IS maintenance belong to the central IT department, to each functional unit, or to the body that first developed the system?
Operation responsibilities: Should the responsibility for IS operation belong to the central IT department, to each functional unit, or to the body that first developed the system? There is also the possibility of out-sourcing.
End-user support: Should end-user support be the responsibility of a central IT department or of each functional unit? Should support for some applications be centralised but for others not?
Target platform standardisation: Should software and hardware equipment be standardised throughout the enterprise, or should each functional unit be allowed to acquire resources of its own choice? Should some types of software and hardware resources be standardised and others not?
Development platform standardisation: Should IS development tools and methods be standardised throughout the enterprise or not?
Development project organisation: Should development projects be enterprise-wide, local, or cooperations among functional units? Should project guidelines be standardised across the enterprise?
Data entry: Should data entry be centralised, or should data always be entered at "the source" (i.e., by the functional unit receiving or providing the data), or should each functional unit be locally responsible for entering all the data it needs?

Table 1: IS-architecture strategy issues which act as boundary conditions for requirements engineering work.

3 Requirements engineering issues

In practice, a chosen IS-architecture strategy can not be realised immediately in terms of changes to all the enterprise's ISs. Instead, a gradual approach must be followed in which continuous organisational, IS and IT developments and redevelopments are undertaken to bring the IS architecture closer to a desirable future situation. Fundamental to the RAISA view of IS architecture work is therefore the understanding that a chosen IS-architecture strategy only can be realised through dissemination into the enterprise's operational plans and guidelines for IS and IT work. In particular, this applies to existing requirements engineering techniques, methods and procedures, both for development and redevelopment of information systems.

Table 1 lists the IS-architecture strategy issues that act as boundary conditions for requirements engineering, ranging from the management level (coordination and responsibilities) down to the technology level (communication method and IT architecture.) Although distinct, the issues listed are by no means independent. Indeed, it is likely that strong correlations exist between decisions about, e.g., data localisation, schema integration and communication method; the choice of a centralised database would necessarily call for a global schema-integration effort on one hand, and for storage-based communication on the other. The choice of storage-based communication will in turn strongly suggest a centralised communication structure, and so on. Identifying correlations such as these between major IS architecture issues is an interesting research task in itself.

Similar considerations apply to IS maintenance: To reduce overall IS architecture entropy, maintenance work must strive to streamline older ISs with the approved IS-architecture strategy as much as possible whenever possible.

With respect to software quality, the IS architecture considerations stress conformance and adaptability as important quality factors. *Conformance* relates to the degree to which the IS is built according to an explicitly expressed, overall IS-architecture strategy for the enterprise. *Adaptability*, on the other hand, emphasises that each individual information system must simplify future attempts to bring adjacent, already existing ISs more in line with the chosen strategy.

4 Conclusion and further work

The RAISA project augments existing work on IS architectures in several ways: 1) It has a *practical orientation* in its aim of developing a practical and operational description technique and assessment method. 2) It *focusses on the selection process* of finding and adapting the IS-architecture strategy that best suits the enterprise's needs. 3) It takes a *broad view of IS architectures* considering several enterprise levels when reviewing an IS architecture.

The bottom line of this paper is that requirements engineering can *not* be carried out in isolation. Instead, enterprise-wide IS-architectural issues must be settled prior to development in terms of an IS-architecture strategy. The IS-architecture strategy must limit the solution-space of the development project by explicitly resolving major IS architecture issues such as those listed in Section 3. Reversing this argument, the paper also stresses that an IS architecture study must be operationalised by disseminating it into existing strategies and plans for IS development and redevelopment.

At a more general level, the paper argues that information resources can *not* be managed independently of their associated information processes. Hence an IS-architecture strategy can be seen as a broad strategy for information resource management, in which the *information processes* are regarded as resources on an equal footing with the information itself.

The RAISA project is still in an early phase, and there are numerous paths for further work. The case studies are likely to suggest improvements and extensions to the metamodel. In particular, the current formulation is weak with respect to representation of users roles and responsibilities. The list of IS architecture issues in Table 1 should also be revised and extended based on case study experience. Determining correlations between them is a particularly interesting task. A further possibility is to realise the metamodel — together with appropriate visual description techniques — in a RAISA tool. The tool would implement as much of the knowledge and experience collected from the project as possible, including support for assessment and improvement of the represented architectures.

Contemporary enterprises will continue to become more and more dependent on information processing within a continuously changing business environment. The situation calls for enterprise-wide representations and strategies for managing the enterprises' information systems and communication structures and methods. Conventional requirements engineering methods must therefore become increasingly streamlined with IS architecture work to remain adequate in the future.

References

[1] Brandt R. Allen and Andrew C. Boynton. Information architecture: In search of efficient flexibility. *MIS Quarterly*, December 1991.

[2] Erling S. Andersen, Ingeborg Baustad, and Åge Sørsveen. *Ledelse på norsk*. ad Notam, Gyldendal, Oslo, 1994. In Norwegian: "Management the Norwegian Way".

[3] Erling S. Andersen and Andreas L. Opdahl. A Metamodel for IS-Architecture Representation. Reports in Information Science No. 42/95, ISSN 0803–6489, Department of Information Science, University of Bergen, 1995.

[4] Gunnar E. Christensen, Stein Erik Grønland, and Leif B. Methlie. *Informasjonsteknologi — strategi, organisasjon, styring*, chapter 7. Bedriftsøkonomens Forlag, Oslo/Norway, 2 edition, 1992. In Norwegian: "Information Technology — Strategy, Organisation, Management".

[5] Gordon B. Davis. Strategies for information requirements determination. *IBM Systems Journal*, 21(1):4–30, 1982.

[6] Per O. Flaatten et al. *Foundations of Business Systems*, chapter 5–6. Dryden, 1989.

[7] Per O. Flaatten. Requirements for a life-cycle methodology for the 1990s. In William W. Cotterman and James A. Senn, editors, *Challenges and Strategies for Research in Systems Development*, Series in Information Systems, pages 221–234. John Wiley & Sons, Chichester, U.K., 1992.

[8] David Garlan and Mary Shaw. An introduction to software architecture. In V. Ambriola and G. Tortora, editors, *Advances in Software Engineering and Knowledge Engineering*, volume 1. World Scientific Publishing Company, New Jersey, 1993.

[9] G. Goldkuhl, K. Petterson, and O. Eriksson. Hur studera realisering och konsekvenser av strategibaserade informationssystemarkitekturer. Technical report, Department of Computer Science, University of Linköping, 1993. In Swedish: "How to Study Realisation and Consequences of Strategy-Based Information System Architectures".

[10] IBM. *Business Systems Planning*, July 1981. IBM Manual #GE20–0527–3.

[11] Thanos Magoulas and Kalevi Pessi. En studie om informationssystemarkitekturer. Technical report, Department of Information Systems, Gothenburg University, 1991. In Swedish: "A Study on Information System Architectures".

[12] M. Lynne Markus. Power, politics and MIS implementation. *CACM*, 26(6), June 1983. Special issue on "Social Impacts of Computing," Rob Kling (ed.).

[13] R. Nolan. Managing the crisis in data processing. *Harvard Business Review*, March/April 1979.

[14] *VBS — Verksomhetsbaserade System, Principer och tillämpningar*, 1991. In Swedish: "Enterprise-Based Systems, Principles and Applications".

[15] *DEL II — Kommunikation mellan verksamhetsbaserade informationssystem, Realisering av Meddelandesamverkan*, 1991. In Swedish: "Part II — Communication Between Enterprise-Based Information Systems, Realisation of Message Passing".