Strategic Information Systems Planning with Box Structures

Alan R. Hevner
Donald J. Berndt
College of Business Administration
University of South Florida
Tampa, FL 33620
{ahevner, dberndt}@coba.usf.edu

James Studnicki
College of Public Health
University of South Florida
Tampa, FL 33620
jstudnic@hsc.usf.edu

Abstract
Strategic Information Systems Planning (SISP) is the process of aligning an organization’s business strategy with effective computer-based information systems to achieve critical business objectives. SISP is a top concern of major executives and considerable resources (time and money) are spent in SISP activities. Many SISP initiatives are not successful due to the difficulty of implementing the recommendations. A significant problem is the Specification Gap between the description of the recommended systems and the detail needed for actual system implementation. Existing SISP methods do not provide sufficiently rigorous representations to specify detailed system recommendations. Box structures are proposed as a solution to this problem and a SISP process with embedded box structure methods is presented. We have used this innovative process in two SISP projects with large organizations. Partial results from one of the projects are presented as a case study to illustrate the use of box structures and their benefits.

1. Strategic information systems planning (SISP)

Strategic information systems planning (SISP) is defined as “The process of identifying a portfolio of computer-based applications that will assist an organization in executing its business plans and realizing its business goals” [9]. The key objective of SISP is to align the organization’s business strategy with its information technology strategy [5].

Research on SISP has focused primarily on developing conceptual frameworks for understanding the SISP process. Seminal proposals by Zani [25], King [8], Rockart [21], and Porter and Millar [17] led to the understanding of important business planning concepts such as top-down strategy, critical success factors, and value chain analysis. More recent research by Galliers [2], Henderson and Venkatraman [5], Lederer and

Salmela [9], Raghunathan and Raghunathan [20], and Sambamurthy et al. [22] have enhanced our understanding of how best to execute SISP and how to measure its ultimate success in the organization.

Transitioning this research into effective practice is difficult. SISP has been identified as a top concern of corporate executives and managers [16]. As new business strategies and information technologies are both rapidly moving targets, it is a very challenging task to produce an effective plan that achieves business objectives with efficient information systems support. The success, and even survival, of an organization in today’s markets is largely dependent upon the development and implementation of a coherent and innovative strategic information systems plan.

Organizations invest vast amounts of time and money in SISP projects. In a typical SISP project, teams of key managers, users, selected clients, and IS specialists are formed and a planning methodology is chosen. There are a number of well-defined and documented planning methodologies available [13] that can be customized. Or the organization can hire an IS consulting company to train and guide the teams through its proprietary methodology. While the benefits of effective SISP are obvious and have been clearly demonstrated in many organizations, most industrial surveys show considerable dissatisfaction with SISP projects [1, 4, 10, 18, 19]. In particular, the time investment, the inability to implement the IS recommendations, and the cost of the projects are viewed as significant problems.

The remainder of this paper is structured as follows. In Section 2, we identify a major problem with current SISP methodologies, the Specification Gap. A proposed solution using box structure methods is presented in Section 3. The use of box structure concepts is demonstrated in a case study of an actual SISP project in Section 4. The paper concludes with future research directions.
2. The specification gap

Empirical research on SISP projects [1, 10, 18] has identified a number of critical factors that determine the success of the planning project and the recommended IS plan. A major factor that leads to the perceived failure of many SISP projects is the inability to effectively implement the recommended IS solutions. Premkumar and King [18] present several cited reasons for implementation failure, including resource shortages, substantial changes required in existing systems, significant investment in new systems, and friction between IS personnel and users. We propose that an overarching reason for implementation failure is a vast Specification Gap between the recommended IS solutions and the detail required to actually implement the desired information systems.

SISP methodologies require the participants to devote large amounts of time and effort to requirements engineering. Typically, existing systems are described via context diagrams, data flow diagrams (DFDs), and entity-relationship diagrams (ERDs). Then, guided by the business strategy and business objectives, enhanced systems are reengineered or originated. The result of the SISP process is a strategic IS plan accompanied by descriptions of the new, recommended systems that implement the strategic plan. Such descriptions are often at a fairly high-level of abstraction in DFD and ERD forms. These descriptions become the de facto specifications for the subsequent implementation of the new IS strategy.

A recent study by Lederer and Sethi [10] sheds a critical light on the effectiveness of the SISP project teams in applying system description techniques. They gathered responses from 105 firms who had completed SISP projects. The firms were asked to state whether they followed 71 prescriptions hypothesized to be associated with SISP success. Based on the survey data, the authors ranked the prescriptions by the extent followed. All the prescriptions relating to system descriptions ranked in the bottom 25% of the list. The system description prescriptions included:

- Use of entity-relationship diagrams
- Use of data flow diagrams
- Use of process diagrams
- Use of process-data matrices

Thus, the authors conclude that SISP teams are not very concerned with process and data architecture issues. This finding corresponds closely with research by Goodhue et al. [4] who found that the system (process and data) recommendations of SISP projects are highly ambiguous.

A clear analogy with the specification gap in SISP can be observed in the software development of large systems. Frequently, the systems design team, using techniques such as DFDs and ERDs, develop system specifications that are incomplete and ambiguous in key areas [15]. The specification is passed on to programmers who must fill in the needed details to complete the implementation. More often than not it is just these ‘to be completed’ details that determine the success or failure of the system. Effective design methods must establish a discipline in which rigor and completeness are enforced at all levels of system abstraction.

As a solution to the SISP specification gap, we propose the use of box structure methods. They provide the rigor, completeness, and ease of use needed to specify recommended IS solutions that bridge the specification gap to effective system implementation. The key asset of box structure representations and methods is that they are scale-free. Box structures can handle the full range from high-level system abstractions to low-level abstractions with the same amount of rigor and precision.

3. SISP process with box structure methods

SISP project teams face many challenges as they attempt to develop an information systems strategy that supports the organization’s business strategy. The representations of the existing (i.e., ‘as is’) systems and the desired (i.e., ‘utopian’) systems must be produced as natural outcomes of the SISP process. The majority of project participants, however, are not IS professionals and they are not familiar with IS representational techniques. Thus, the work of describing the systems is either done by internal/external IS experts or it is not done at all (as indicated by the research in [10]). Effective system representation methods should be an integral part of any SISP process and all participants must understand them and their use.

An important issue is -What level of system detail is needed in the desired system specifications found in the strategic IS plan? The system representation method must be able to support increasingly more detailed levels of abstraction in a top-down system hierarchy. While extensive design detail is not needed (and not desirable) in the IS plan, it is important that the system description be rigorous and complete at an appropriate level of abstraction. In this form, the IS specification can serve as a well-defined and unambiguous starting point for the implementation of the desired systems.

Box structure methods of system representation satisfy both the above requirements for ease of use and rigor. A review of box structure foundations is provided in the
Appendix. Box structure methods are proposed as a complement to existing SISP approaches. An outline of an enhanced SISP process with embedded box structure methods is presented below. We assume that the current portfolio of organizational systems is described in sufficient detail to be used as input to the SISP process, providing the foundation for new system specifications. Otherwise, box structures can also be used to describe the existing systems as well.

The newly proposed SISP process provides a number of innovative enhancements to current SISP methodologies. Overall, the box structure methods support a rigorous framework for the project teams to structure the systems planning information. Once the teams identify the Critical Business Processes (CBPs) via well-established and effective planning methods, the CBPs are individually described in detailed Black Box, State Box, and Clear Box forms. Note that teams can work independently in parallel on groupings of the CBPs. This increases the efficiency of the SISP process.

The following information is gathered for each CBP:

- **Black Box Definition** – Information on CBP inputs and outputs are elicited from the team. Functional transactions are identified that transform sets of inputs into sets of outputs (see Figure 1). This provides a behavioral view of the CBP.
- **State Box Description** – The state information that is required to be persistent in the CBP is elicited from the team. Rigorous expansion and closure operations are performed to help identify such persistent data. Entity-relationship diagrams can be used to describe the state. State box descriptions (see Figure 2) are useful for planning information repositories in the planned systems.
- **Clear Box Description** – This step expands the CBP description by eliciting information on the roles and activities to be performed. Process flow diagrams can be used to describe the sequence dependencies among essential process activities (see Figure 3). There is significant potential for optimizing the workflow of the CBP in this step.

---

**Strategic Information Systems Planning (SISP) Process with Box Structure Methods**

1. Collect and analyze all pertinent SISP information, to include:
   - Business strategies and business objectives for the organization
   - Current information systems portfolio (‘as is’ descriptions of existing systems)

2. Apply strategic planning methods (e.g., Critical Success Factors [21], Value Chains [17], Strategic Alignment [5]) to identify Critical Business Processes (CBPs) in the organization.

3. For each CBP, describe the CBP in box structure forms:
   3.1. Produce a Black Box Definition of the CBP. (Figure 1)
   3.2. Produce a State Box Description of the CBP. (Figure 2)
   3.3. Produce a Clear Box Description of the CBP. (Figure 3)
   3.4. Verify the completeness, consistency, and closure of the box structures [14].
   3.5. Refine the box structure descriptions as needed to accurately represent the ‘utopian’ CBP.

4. Integrate the CBPs into a strategic enterprise information systems plan.
   4.1. Apply traditional integration techniques (e.g., Process/Data Matrices) as well as innovative ‘Bright Ideas’ for CBP integration.
   4.2. Use box structure composition and decomposition techniques [6] to produce an integrated system hierarchy.
   4.3. Verify the completeness, consistency, and closure of the box structure hierarchy.
   4.4. Refine the box structure hierarchy as needed to accurately represent the ‘Utopian’ enterprise information systems plan.

5. Develop a high-level Incremental Development Plan for implementation of the desired strategic information systems.

6. Produce a Strategic Information Systems Plan that includes the information systems specification and the incremental development plan for implementation.
Figure 1: Black box description

Figure 2: State box description
The underlying mathematical basis of the box structure methods allows the verification of the important properties of completeness, consistency, and closure [14,15]. The teams refine the box structure descriptions until they are satisfied that the critical business processes are represented accurately and completely at the desired level of abstraction.

In our experience guiding SISP teams through box structure methods, we found the teams very open and responsive to the box structure descriptions and methods. After a half-day training session on box structure concepts, the teams were able to participate effectively in CBP description meetings led by an expert facilitator. After some initial struggles to reach the right levels of abstraction in the descriptions, the teams responded readily to the structured elicitation of CBP inputs, outputs, state, and process roles/activities.

The integration stage of planning is a critical step to achieve an effective IS plan. Given the box structure descriptions of the CBPs, there are a number of standard techniques used to analyze integration options. Examples of integration techniques include Process/Data Matrices and Trade-off Analysis [13]. However, we believe that this stage of the process is ripe for the discovery of ‘Bright Ideas’ for CBP integration. We encourage the project team to brainstorm on ideas that break current business molds. These innovative ideas are used to guide the integration of the CBPs into an enterprise IS architecture. Box structure methods of composition and decomposition are used to form a rigorous Box Structure Usage Hierarchy. The principles of state migration and common services also provide guidance for the development of the integrated system hierarchy. The system structure is verified and refined by the team members until they are satisfied with the accuracy and completeness of the IS specification.

An additional step of strategic planning is strongly recommended. This step involves the development of a strategic incremental development plan [24] for implementation of the IS specification. In building a first-cut incremental development plan, the SISP project is forced to consider issues of implementation feasibility; for example, resource availability (systems and skilled personnel), schedule requirements, and costs of revitalizing old systems and buying new systems. Such considerations may impact decisions on the recommended IS architecture. The inclusion of an initial development plan in the SISP recommendations should alleviate many of the reasons for implementation failure found by Premkumar and King [18].

The completed Strategic Information Systems Plan would include a rigorous information systems specification as a recommended IS plan and an incremental development plan for implementing the IS specification.

4. A case study of a SISP project

We have applied the proposed SISP process with box structure representations in two projects with large healthcare organizations in Florida. In this section, we
present selected results from one of the projects in order to demonstrate the feasibility and effectiveness of box structure methods in an actual SISP project.

Over the period of three months, we facilitated a SISP project for a rapidly growing healthcare organization. The SISP task force was made up of 40 thought leaders in the organization. Both IS users and technicians were included on the task force. At the start of the project, the participants were given a half-day of training on box structure specification and analysis techniques. A set of exercises was performed both in class and at home to familiarize the participants with concepts of box structures, critical success factors, and integrated system architectures. Due to the limited timeframe of the project, only an abbreviated version of the full SISP Process, as described in Section 3, was executed. The following sections provide examples of the SISP outcomes.

4.1 Identification of critical business processes

The full project team was divided into three subgroups to begin the identification of organizational Critical Business Processes (CBPs). The three major areas of planning were:

- Patient Access
- Patient Care
- Managed Care

The goals of each subgroup were to identify a ‘utopian’ approach to satisfy the needs of the focus area. The outcomes of the subgroup meetings were a well-defined set of CBPs represented in box structure formats. A total of seventeen CBPs were described in the three focus areas. To illustrate, we present two CBPs from the area of Patient Care – the Request for Service CBP and the Provision of Service CBP.

Patient Care – Request for Service

The request for service process is central to the efficient functioning of a healthcare organization. The request for service must be consistent with the patient’s treatment plan and must be authorized by a physician. The outputs of the process would be one or more orders for patient service sent to the responsible departments either within or outside of the organization. Appropriate physicians and caregivers update the patient record. Figure 4 presents the box-structured representation of this CBP.

Patient Care – Provision of Service

The request for service initiates the provision of service to the patient. The actual performance of the service will vary. However, the outputs of the service are consistent across all types of service. The outcomes of the service must be recorded in the patient record and transmitted to the appropriate physicians and caregivers. The patient must also be educated as to the implications of the outcomes. In addition, financial charges must be accurately posted to the patient’s account. Figure 5 presents the box-structured representation of this CBP.

These box-structured representations identify the black box inputs and outputs, the state box state, and the roles and activities found in the clear box. The groups found these levels of abstraction to be appropriate for the strategic planning goals of the organization.

4.2 An integrated information systems architecture

Simply identifying a set of Critical Business Processes for the healthcare organization is not sufficient. The CBPs must be integrated into an information systems architecture that supports top quality patient care at an efficient cost. The SISP task force developed an innovative set of strategies to integrate the CBPs into a comprehensive healthcare delivery system. Healthcare Integrated Delivery Systems (IDSs) [23] must be composed of affordable, efficient, interoperable information systems. These systems must operate seamlessly across a wide variety of institutions – hospitals, clinics, pharmacies, laboratories, medical centers, etc. Leading healthcare user groups, such as HL7, CORBAmed, and MS HUGS, are proposing IDS architectures based on distributed object technology.

Based on the box structure representations, we developed a top-level model of an IDS information systems architecture in terms of high-level business objects. These business objects are drawn from the critical business processes (CBPs). The innovative integration ideas mentioned above contribute the framework for enhancing quality of patient care and improving cost efficiencies in the architecture. We identified ten fundamental objects in the IS architecture. Each of these complex objects will encapsulate the essential information for the domain and the required functional interfaces for interaction with other objects and the external environment. The ten fundamental business objects are:

- Electronic Patient Record (EPR)
- Personnel Administration
- IDS Institutions
- Patient Access Management
- Patient Case Management
- Request for Services (Ordering)
- Provision of Services (Delivery)
- Clinical Decision Support
- Contract Management
- Financial Services
Box structure representations of each of these objects were developed at an appropriate level of abstraction [6].

Under the proposed open systems architecture, we illustrate a high-level process flow for an important healthcare scenario - In-Patient Process Flow. We use an Activity Diagram to provide a basic understanding of the workflow through this scenario. Activity diagrams are a behavioral modeling technique found in the Unified Modeling Language (UML). Additional details on the development and use of activity diagrams are located in the UML literature (e.g., [3]).

The In-Patient Process Flow is shown in Figure 6. The primary role players, located in the swimlanes, are:

- Access Team Members
- Primary Caregivers - Physicians and Nurses
- Case Manager
- Ancillary Units – Surgery, Pharmacy, Labs, etc.
- Account Team Members

The Patient is shown as an object with an encapsulated EPR that is constantly updated with new information throughout the workflow. The activities and workflows in the diagram are largely self-explanatory. It provides an illustration of how business processes can be represented and reengineered based on the proposed IS architecture.
Critical Business Process: Patient Care - Request for Service: Physicians and/or Caregivers produce a request for patient services.

**Inputs:**
- Care Giver Orders
  - Testing
  - Surgery
  - Therapy
  - Medication
  - Nutrition
  - Social Services
  - Pastoral
  - Clerical
  - Consults
  - Nursing Treatments
  - Patient Record
  - Physician Authorization
  - Treatment Plan
  - Patient/Family Consent

**State:**
- Patient Record
- Patient Care Notes
- Log of Patient Orders

**Roles and Activities:**
- Care Giver: Identify need, Understand order, Analyze order, Record order
- Physician: Identify need, Analyze order, Authorize order
- Patient/Family: Understand order
- Clerical: Record and transmit order

**Outputs:**
- Updated Patient Record
- Notification of Order to Responsible Department
- Order Protocols
- Modification of Order

---

Critical Business Process: Patient Care - Provision of Service: Responsible department performs the service and reports the results.

**Inputs:**
- Service Request Order
- Laboratory
- Pharmacy
- Social Services
- Rehab
- Consults
- Imaging
- Nuclear
- Surgery
- Anesthetics
- Nursing
- Respiratory
- Patient Record
- Requesting Physician/Nurse
- Treatment Plan
- Time Needed

**State:**
- Patient Record
- Department Record File
- Chronological Log of Service
- Billing Records

**Roles and Activities:**
- Care Givers: Scheduling, Preparation, Delivery of Patient, Delivery of Results, Analysis
- Clerical: Record Results
- Physician: Authorization, Analysis, Reorder
- Patient: Understand
- External Service Provider: Provide Service and Results

**Outputs:**
- Performance of Service
- Transmit to External Provider
- Request for Clarification or Modification
- Outcome of Service Results
- Updated Patient Record
- Patient Education
- Charge Generation

---

Figure 4: Request for service CBP

Figure 5: Provision of service CBP
Proceedings of the 33rd Hawaii International Conference on System Sciences - 2000

Figure 6: In-Patient Process Flow
4.3 SISP project outcomes

The outcomes of the SISP projects include the organizational CBPs, an integrated IS architecture that supports all CBP requirements, and an incremental development plan for developing the IS architecture in well-defined phases. The representation of these outcomes in box structure formats provides a clear starting point for the design and implementation of the integrated information systems in the healthcare organizations. In the referenced projects, information from the SISP outcomes is being used as input to RFPs for vendor selection of systems that meet the specifications of the strategic IS architecture.

5. Conclusions and future research

Strategic information systems planning (SISP) is a highly visible, critical activity for business organizations. Recent surveys have provided evidence of a Specification Gap in most strategic IS plans that leads to significant problems in implementing the desired systems. The use of rigorous box structure methods in SISP projects has the potential to bridge this specification gap. We present a comprehensive SISP process with box structure methods and a case study of the use of box structure representations and methods in a large SISP project.

Box structure methods provide a number of important advantages for strategic planning:

- Ease of Use – With a modicum of training, non-IS managers and users are able to participate in the elicitation of information on inputs, outputs, state, and activities for the description of critical business processes (CBPs).

- Natural Form of Representation – Processes are described in a natural progression from black box to state box to clear box. Team members are able to see how state decisions are based on available inputs and required outputs, and how activities are organized based on the processing of inputs and state data.

- Rigorous Description Methods – The mathematical basis of box structures provides a rigor not found in more informal methods. This allows the verification of principles such as specification completeness, consistency, and closure.

- Integration of CBPs in a Box Structure Usage Hierarchy – Composition and decomposition techniques are used to integrate the CBPs into an enterprise IS architecture. Innovative ideas can be applied along with traditional methods for building an integrated architecture.

- Same Representation for IS Planning and IS Implementation – Box structures are scale-free. Thus, box structures are effective representations for both planning and implementation of strategic systems. This provides a common means of communication between the planners and the developers of the software systems. Ambiguity of the IS recommendations is significantly reduced.

Future research will involve using the proposed SISP process in several major projects with industrial partners. Particular attention will be directed toward measuring the ease and effectiveness of implementing the recommended systems. Also, a controlled experiment is being planned in which one project team will use a traditional SISP process and another team will use the new SISP process with box structures. Ease of use and usefulness of the box structures will be studied.

Acknowledgements

We acknowledge the contributions of Lauralee Pasko to this work. We also thank the team members in the healthcare organizations who participated in the SISP projects.

References


APPENDIX: BOX STRUCTURE FOUNDATIONS

Box structure methods are a central technology in Cleanroom software engineering [11]. Box structure methods provide mathematics-based technology, processes, and expressive forms that can be applied to software system definition and verification [14]. Box structures emphasize technical rigor and management simplicity. They permit definition of system components and their behavior, data, and control in terms of three fundamental system structures that can be nested and sequenced over and over in box structure hierarchies. These system structures are black box, state box, and clear box.

The fundamental principle of box structures is that systems and their components can be regarded as rules for mathematical functions (or relations). That is, systems and components carry out transformations from input (domain) to output (range) that can be specified as function mappings. The three box structures are special forms of mathematical functions that correspond to useful and natural system views that can be derived in a stepwise decomposition/composition and verification process.

A black box maps the current stimulus into a response that also depends on the history of stimuli received. A black box is uniquely determined by its stimulus history. Black box definitions are state-free and procedure-free, referencing only external stimuli and responses. Black boxes define required behavior in all possible circumstances of use, including expected, error, and stress uses. The behavior required for all usage scenarios must be defined for mathematical completeness. Research on the application of black boxes for requirements specification is reported in [7].

A state box maps the current stimulus and the current (i.e., old) state into a response and a new state. In the state box, the stimulus history of the black box is replaced by persistent state data necessary to achieve black box behavior. A state box definition is procedure-free and isolates and focuses attention on state invention. The principle of transaction closure requires that the transactions of a system or system component be sufficient and necessary for the acquisition and preservation of all its state data and that its state data be sufficient and necessary for completion of all transactions.

A clear box is a program, or set of programs, that implements the state box and introduces and connects components in an execution structure for independent decomposition at the next hierarchy level. Such connections are a key feature of box structure methods and are critical to maintaining intellectual control in large-scale system development. Clear boxes are expressed in familiar control structures such as sequence, conditional (i.e., if-then-else), iteration (i.e., while-do), and concurrency structures. Correctness verification of clear boxes is carried out through application of the Correctness Theorem [12], typically in verbal proofs of correctness in team reviews. The theorem reduces verification to a finite number steps, each based only on localized reasoning, even though programs contain a virtually infinite number of paths.