

Modeling and Simulation in Enterprise Integration - a framework and an application in the offshore oil industry

L. C. Christensen^a, T. R. Christiansen^a, Y. Jin^b, R. E. Levitt^b & J. Kunz^b

*^aDNV Research, Veritas veien 1, 1322 Høvik, NORWAY,
Tel.: +47 - 6757 9900, FAX: +47-6757 7520, e-mail: tchr@vr.dnv.no*

*^bStanford University, Civil Engineering Departement, USA,
e-mail: rel@cive.stanford.edu*

Abstract

This paper presents an approach to enhance enterprise integration in engineering projects, that makes systematic use of model definition, formalization and analysis. Our definition of enterprise models is based on a philosophy for thinking about enterprise in terms of “an Organization, carrying out a set of Processes to create one or more Products which satisfy predefined Objectives.” This framework, which we have denoted the OPPO, illustrates our belief that enterprise models must give a complete and correct description of relevant aspects of reality in order to address their stated purpose. Based on the OPPO framework and an information processing view of project enterprise, we formalize our models using a methodology for describing coordination requirements. This is based on the well known interaction matrices of Quality Function Deployment (QFD), to develop so called “houses of complexity, uncertainty and interdependence” that describe important dependencies between project requirements (objectives), deliverables (products), activities (process) and team members (organization). We use these dependencies as a basis for deriving measures of the coordination which must take place between various project participants during project execution. We can then use these coordination requirements as input to analysis, in the Virtual Design Team (VDT) object oriented discrete event simulation environment. The simulation results can be used for systematically assessing predicted effects of proposed changes. The framework, methodology and analysis is illustrated by an example from engineering design of subsea modules for oil and gas production in offshore field development. The results from simulation gives various performance measures, including critical path duration, work volume (a substitute for cost) and process performance in coordination (error handling and communication attendance).

Keywords

Engineering management, Process modeling, Quality Function Deployment, Information models, Monte-Carlo simulation

1 INTRODUCTION

In this paper we describe an approach to enhancing enterprise integration by modeling of engineering projects, and how we use enterprise models to predict probable effects of proposed changes in the project organization. We illustrate our description with an example taken from offshore field development in the Norwegian part of the North Sea.

Our motivation for modeling and analyzing project enterprise is the increasing pressures for improved performance facing engineering projects in almost every industry today, and the belief that "there is a general lack of knowledge and appreciation of the capabilities of integrating systems to improve the productivity and economic return of enterprises of all types" [Williams et.al 93].

The objective of the current work is to demonstrate how modeling and analysis can be used in an attempt to establish a more complete and correct understanding of project enterprise, and how such understanding can be turned into actual performance measurement (which may in turn be used as metrics for real improvement). Enterprise Integration can be defined as; "*the collection, reduction, storage and use of data (and/or information) from the business entity involved, plus its environment, in order to optimize the operation of the business entity as a whole according to the criteria established by that business entity's management. It must also include the integration of the raw material, intermediate and final product flows, as well as machine organization, to further the enterprise optimization possible*" [Williams et. al 93]. From this follows that one must describe, analyze and evaluate different ideas for improvement in order to optimize performance. In this context Enterprise Models (EM) can be divided into three main uses [Christensen et.al 95]: (I) distribution of cognitive models, (II) analysis and (III) embedded models in information systems. So far we have focused on category I and II. In future work we aim to enact our project enterprise model as part of the actual project planning and project control during execution (category III).

Given this objective, we first describe our framework for thinking about project enterprise, in section 2. Next we describe our methodology for building formal enterprise models, and an associated enterprise modeling toolkit, in section 3. Then, in section 4 we briefly describe a discrete event simulator, The Virtual Design Team - VDT [Jin et.al 95], which uses our enterprise model of engineering design projects as input, and produces a set of predicted performance measures resulting from proposed changes in the execution of the project task or organization of the project team. All of these sections are illustrated by a project to design subsea modules for offshore oil production. Finally, in section 5, we outline some of the directions in which we are continuing our work.

It is our intention to give an overview of the totality of a complete, and fairly complex, research activity. In this paper we do not provide details of all aspects of our work, and some other references for this research activity are Christensen [94] and Christiansen [93A & 93B].

2 FRAMEWORK - INTEGRATED THINKING ABOUT PROJECT ENTERPRISE

Improving real engineering design projects requires a clear understanding of the current situation, a vision of the desired situation, and a strategy for achieving the necessary change [Davenport 93]. It is our belief that all of these requirements may be addressed by enterprise modeling and analysis. However, enterprise modeling takes time and costs money, and thus enterprise models should have a stated purpose. The model must be developed with this purpose

in mind. Reality is most often complicated and confusing, and insight is rarely achieved without considerable simplification. In order to improve project enterprise we must understand it, and thus the challenge in enterprise modeling is to make the necessary simplifications, without losing elements that are essential to representation and reasoning. In other words, all enterprise models must be developed in the spirit of giving a *complete* and *correct* description according to model purpose.

Our philosophy for thinking about enterprise is in terms of “an *organization*, carrying out some (set of) *process(es)* to create *products* which satisfy predefined *objectives*.” That is, we wish to highlight the four pillars on which purposeful human action rests: why we act (the objective), what is the result of action (the products), when and how we act (the process), and with whom we interact (the organization). Project enterprise is thus characterized by an assigned team of people, working together for a planned period of time to deliver according to specification, and thus achieve a stated purpose.

2.1 The Statfjord Subsea Satellites engineering design project

Engineering design projects in offshore field development are undertaken to develop design drawings and procurement specifications for complex and costly installation components. Typical design objects include fixed and floating structures, drilling and processing systems, and control and automation systems.

The Statfjord Subsea Satellites (SSS) project involves development of a set of subsea production units for the Statfjord Field. A total of eight subsea production units were installed at various locations, in order to increase the total amount of production, as field characteristics changed during production from three existing platforms. The subsea units, which produce and preprocess crude oil, were connected to an existing platform for further processing and transportation. The total SSS project involved several hundred thousand person-hours, of which the engineering design contract constituted somewhere around 30 thousand person-hours, carried out over 15 months by an engineering design team varying between 10 and 25 persons.

Figure 2.1 shows an illustration of the one of the Statfjord Subsea Satellites. This drawing is in itself part of the deliverables from the engineering design project. We shall use the SSS project throughout this paper as an example application to illustrate our framework and methodology for enterprise engineering.

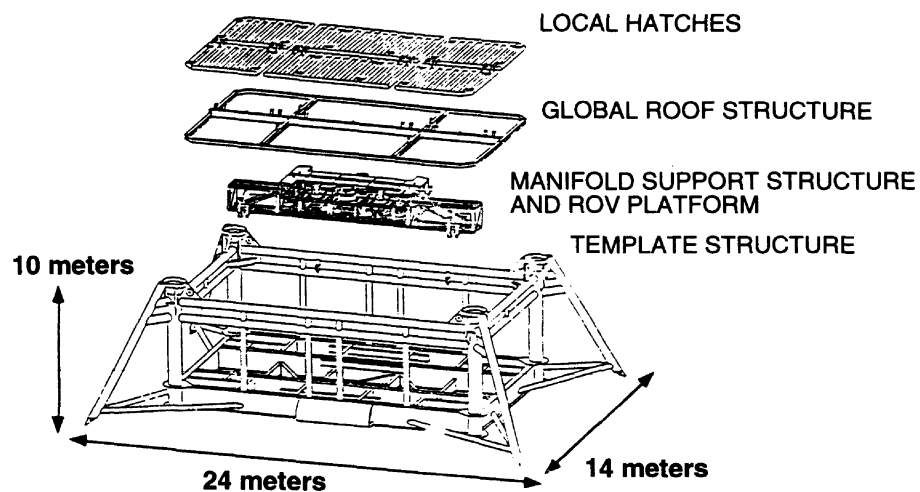


Figure 2.1 The project deliverable from the Statfjord Subsea Satellite (SSS) project

2.2 Describing objectives and products

For almost any purpose, we believe that *model completeness* is addressed by a suitable description of all of these dimensions, holistic thought about why project enterprise is undertaken, how and by whom it is undertaken, and what the result of enterprise is. In the next subsections we shall look at the various enterprise dimensions of the SSS project in somewhat more detail.

Questions about *why* and *what* relate to required and realized project results, and are part of the objective and product dimension. The difference between them determines project performance, which in this context includes measures of duration and cost (project efficiency), and a description of the behavior and characteristics of project deliverables (project effectiveness or quality)

To represent objects and products we use functional decomposition, and the FUTS technique [Willems 88], in which a top level requirement is met by a corresponding top level solution. This solution generates a set of lower level requirements, which in turn are met by more detailed solutions (which in turn generates new detailed requirements, and so on until a suitable level of detailing for describing procurable specifications).

Figure 2.2 shows the requirements-solution break-down structure for the subsea satellites in the SSS project.

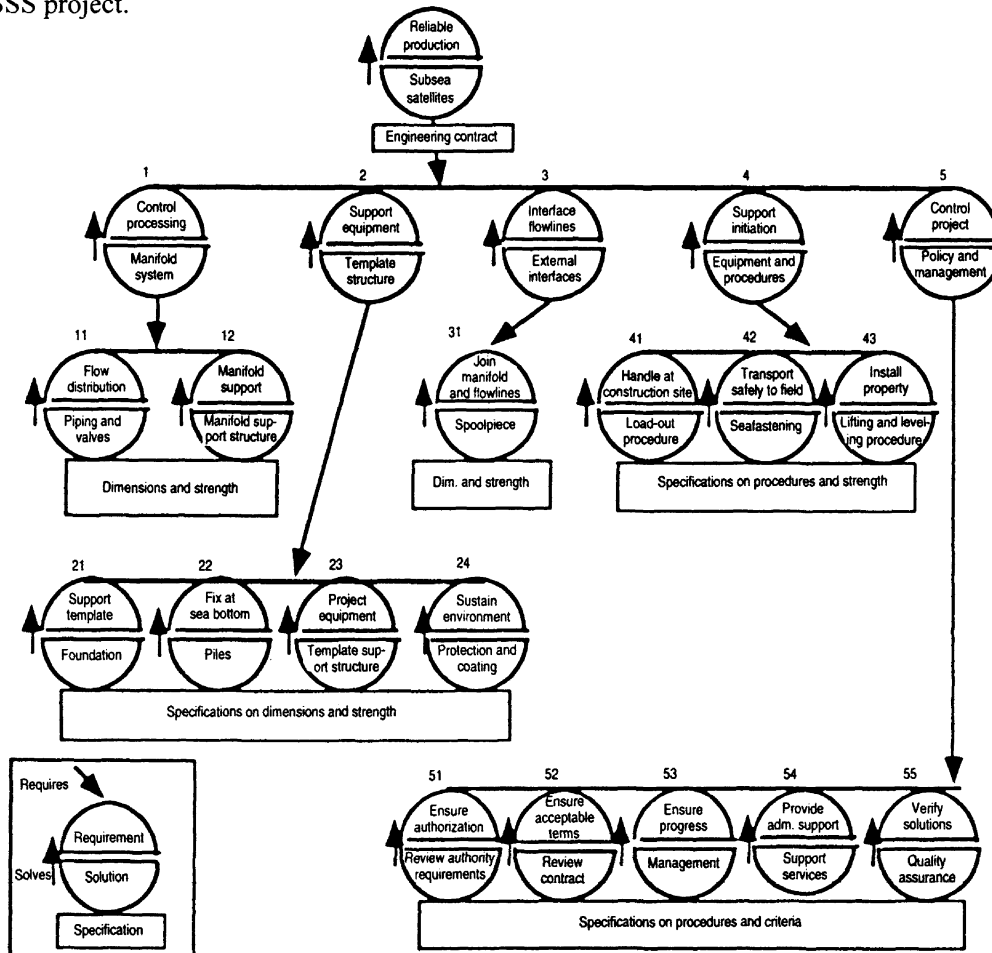


Figure 2.2 Requirements-solution break-down structure for the SSS

2.3 Describing the process

The question of *how* relates to planning and execution of the project, as defined by the process dimension. In the same manner that the difference between objectives and product deliverables defines product performance, the difference between planning and execution defines process performance. We describe processes in terms of *activities* related by precedence (predecessor successor) relations, showing the logic of project execution.

Figure 2.3 shows a simplified project plan for the engineering design phase of the SSS project. Note the high degree of concurrency caused by tight scheduling of activities to carry out both Engineering, Procurement and Construction (EPC) one full year faster than was typical in the past. Note also that the figure only shows the engineering activities for the first of a total of eight modules. Engineering analysis for this first module did of course involve a major part of the work, and the work for the remaining seven modules was performed as a “production line”, in direct continuation of the activities shown.

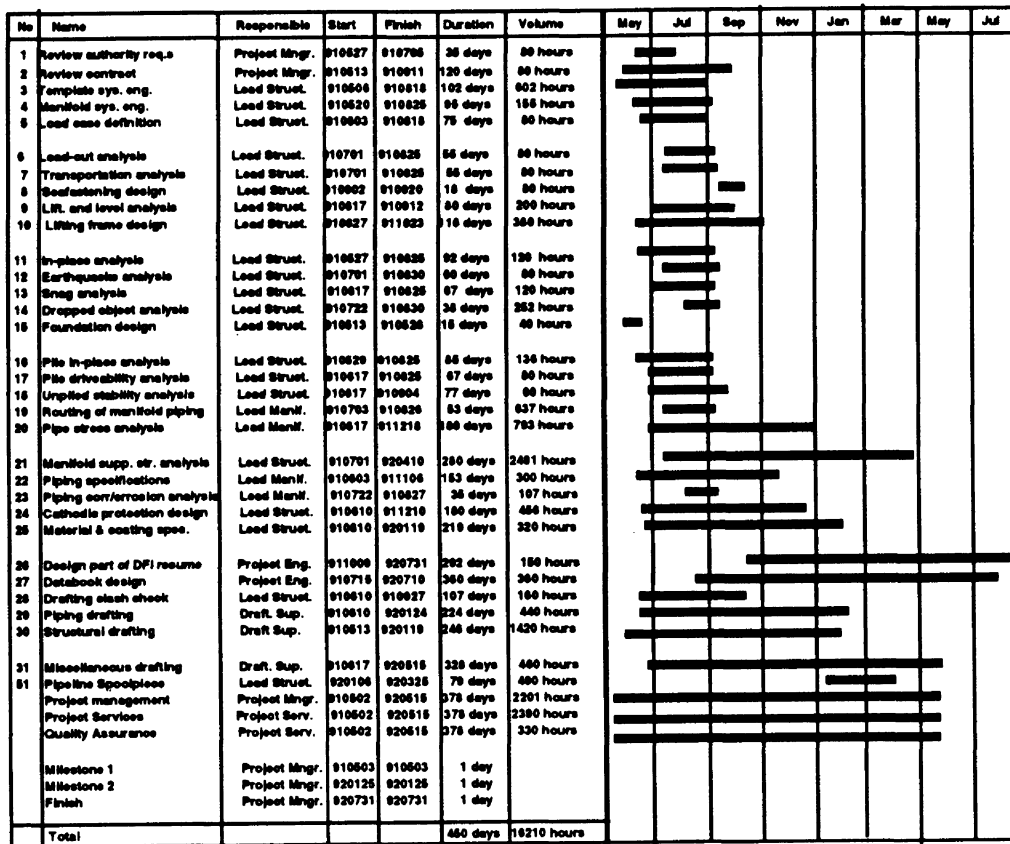


Figure 2.3 Project plan for engineering design of the SSS

2.4 Describing the organization

The question of *who* relates to responsibilities of and relations between project team participants, as defined by the organizational dimension. In addition we describe *actors* in terms of their abilities (craft, skill and experience) and preferences. The difference between project policies

and personal preferences of the project team participants will determine how planned action is translated into actual behavior. This determines organizational performance, and influences and constrains process and product performance.

Figure 2.4 illustrates the engineering design project team organization and their relations to other project team members. Note how dual reporting is caused by two different organizational hierarchies, the functional and project lines of command and control. This is typical for engineering project organizations, and often referred to as a matrix structure [Davis & Lawrence 77]. In the SSS project, the matrix was dominated by the project manager, who controlled both budgets, schedules, performance evaluation and incentive structures. This is the hallmark of a so-called strong matrix. Moreover, the engineering design team was physically co-located, which enabled them to rely on informal communication as their means for handling intense coordination requirements.

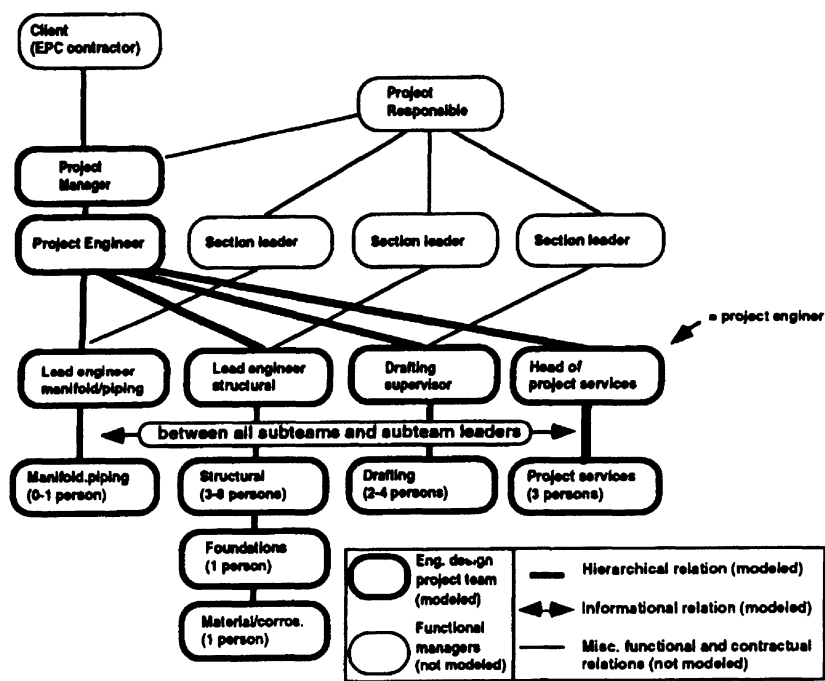


Figure 2.4 Engineering design team organization for the SSS project

2.5 Describing project performance

In the above we have pointed out how performance may be thought of as a measure of the fit between planned and actual project execution, and how this can be applied to both products, processes and organizations. This relates to a view of *model correctness* as the ability of the model to capture difference between ideal and real situations (i.e., planned action vs. actual behavior). Below we define action in enterprise using two very different views of causal logic, and compare their different predictions.

Purposeful engineering professionals like to believe that they inhabit a rational world, where enterprise is explained according to “a logic of intention” [March 88]. In this *normative* view causality is explained starting from objectives, which define some set of required products. One or more processes are devised to create the products, and a suitable organization designed to

carry out the process (which create the products, which satisfy the objective). Thus, project enterprise is seen as a rational means for achieving fulfillment of stated objectives.

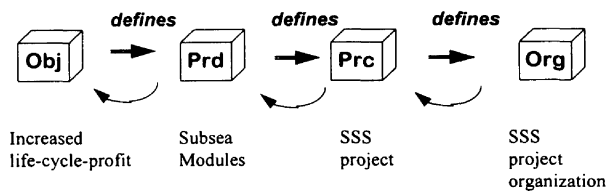
A natural systems view of enterprise [Scott 88], explains causality according to “a logic of implication” [March 88]. In this *descriptive* view enterprise is defined by some set of individuals (the organization). Between organizational members there exist a mix of rational and irrational relations, which determines what processes the organization can (and cannot) carry out. The possible processes determine a set of possible products, for which objectives are devised to explain (and defend) the existence of the organization. Thus, enterprise is seen as a-posteriori justification of action.

In reality, neither of the above models explain human enterprise in full. Reality is filled with bounded rationality [Simon 58], and causality can be viewed by an *interpretative* logic of both intention and implication. That is, a logic where causality is explained according to the chosen interpretation of reality. Note how the word made up by the first letter of our four enterprise dimensions, *OPPO*, is a palindrome. This symmetry symbolizes the way in which distance from enterprise whether in time, space or function, precludes accurate determination of causal mechanisms. That is, the difference between cause and effect is indistinguishable to the observer. The consequence of this duality for enterprise modeling is that representation and reasoning must include constructs that allow description of the differences between intended and implied action, as well as *performance metrics* for the difference between planned action and actual behavior.

Figure 2.5 illustrates the different accounts for causal logic in the SSS project. In an intentional explanation the SSS project was carried out in order to increase the recoverable reserves (life cycle profit) from the Statfjord Field. This was met by installing a set of subsea production units to operate a set of additional wells. Thus the SSS project plan specified design, building and installation of eight subsea modules. Finally, the SSS project team was set up in order to carry out the project.

Explaining the SSS project in action

A logic of intention:



A logic of implication:

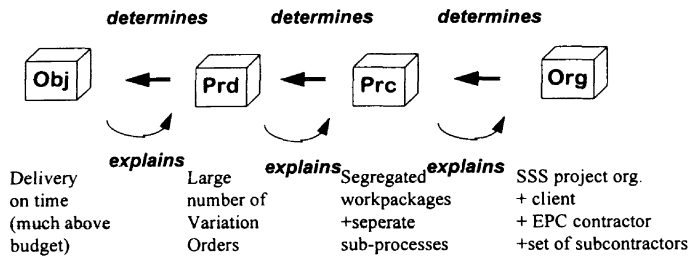


Figure 2.5 Different accounts for causal logic in the SSS project

Conversely, the implied view would look at a set of engineering organizations looking for work [March & Olsen 86]. Thus the SSS project was organized in terms of three main contractors, with a large number of subcontractors to each (lots of work for everyone). The process involved segregated work packages, and separate subprocesses. Because of the resulting communication problems, a large number variation orders (VOs) were generated to handle required changes due to confusion and misunderstanding between different dependent organizations. Consequently, project duration became the critical variable, and successful installation of the modules on time became the number one priority objective (above the intentional quality required to ensure satisfactory operation and reliable production increase), delivered at a cost more than 50 percent above the original budget estimate.

2.6 Summarizing the complete OPPO modeling framework

In figure 2.6 [Christensen et.al 94] we summarize the above framework for enterprise modeling. Starting from the overall objective we use functional decomposition to derive a structured set of requirements as a “desired model.” Given the set of requirements and associated product solutions, and the various types of dependencies between them, we define a sequence of activities to produce the desired deliverable. Between these activities, various types of precedence relations will determine the required control mechanisms throughout the process. By assigning responsibility for the various activities to project team participants (actors), we define a set of related actor dependencies, which determine communication needs. The execution of the project plan by the project team will produce the deliverable (realized solution), which may be compared to the desired solution. Thus, project performance can be explained in terms of definition, planning and assignment of a series of *coordination requirements*, and the associated execution to handle dependencies by a series of *coordination mechanisms*.

In the next chapter we shall outline our methodology for description, planning and assignment. Then in chapter 5 we shall outline simulated execution as a means for analysis.

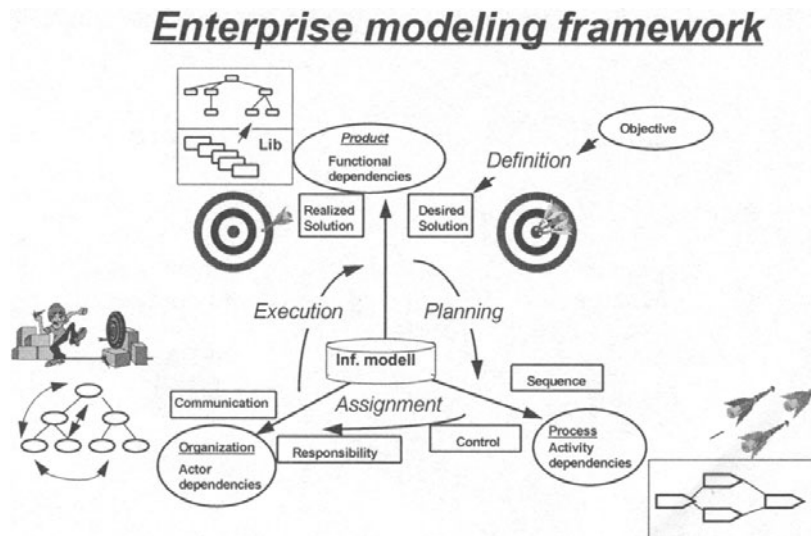


Figure 2.6 A summary overview of the OPPO enterprise modeling framework

3 METHODOLOGY - DEVELOPING FORMAL ENTERPRISE MODELS

In order to turn our modeling framework into a practical *methodology* for developing project enterprise models we view the project team as a structure of relations, and use an abstract analogy between organizational and physical structures. That is, both physical and organizational structures may be thought of in terms of elements with given material properties, connected by nodes in a given configuration. Furthermore, both are subject to loading from their environment, and have a capacity to meet this load as determined by their material properties and configuration. And for both, the match between required and realized behavior under load determines the performance of the structure.

3.1 An information processing model of coordination in engineering design

In this paper, we are interested in information processing (information flow) [Galbraith 73], and define *coordination load* among information processors, as a function of requirements, selected solutions, project plan and allocation of responsibility. We also use an object representation of organizational *capacity* in terms of actors and tools, with suitable attributes to describe their individual capacity and capability.

Given the above "organizational mechanics", our coordination load model attempts to define and operationalize coordination measures which influence the way projects are actually carried out. Since we are concerned with engineering design projects, we model work as a flow of information processing tasks, and describe the various coordination requirements arising between actors that are dependent on each other for producing, consuming and sharing information to carry out a flow of *routinized activities*. By *routinized* we mean that objectives and solution components are well understood, that the process is planned a-priori and coupled to the organization by clear allocation of responsibilities, and that the hierarchy of command and control in the project team organization is uniquely determined.

Figure 3.1 illustrates our view of engineering design teams as information processing entities (from [Cohen 92]), constrained by a set of relations defining their responsibilities for processing, reporting and communication.

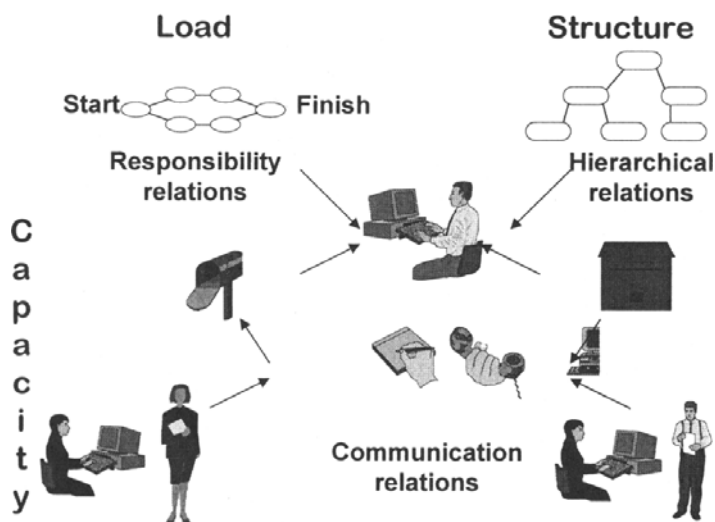


Figure 3.1 Engineering design teams as information processing entities

3.2 A formalized model of coordination load and capacity

Our view of coordination follows Malone's [91] distinction between different types of coordination mechanisms. Thus, coordination capacity is determined by actors' handling of different coordination mechanisms. We distinguish between *vertical* coordination — associated with the flow of control (verification) in organizational hierarchies — *horizontal* coordination — associated with the flow of information (communication), energy (collaboration), and resources (competition) between organizational actors — and *longitudinal* coordination — associated with flow of experience (adaptation) which promotes learning.

Coordination capacity also depends on the skill, experience, policies and preferences of project teams, and a set of coordination mechanisms for handling dependencies between project team members. This defines the information processing behavior of the various actors, in terms of decision making about attention allocation and participation [March 88].

To implement the above coordination model we use a set of matrix techniques based on Quality Function Deployment (QFD) [Hauser & Clausing 88], to calculate project specific measures of *complexity* of various parts of the project task, *uncertainty* of various parts of the project plan, and *interdependence* between various members of the project team.

Figure 3.2 summarizes our framework for defining and modeling the *load distribution* on various project team members during project execution. We see how a structured breakdown of requirements and corresponding solutions, together with activity plans and organization charts for the project team, is input to a set of matrix tools for deriving the relative distributions of complexity and uncertainty for the various activities, and interdependence between project team members.

We use these measures to quantify the probabilities of failure in producing solutions and satisfying requirements, the required communication frequency, and the required participation by project team members (see [Christiansen 93] for details). These are, in our view, important parts of describing how work processes are carried out in the real world by organizations consisting of boundedly rational (human) actors [Simon 58].

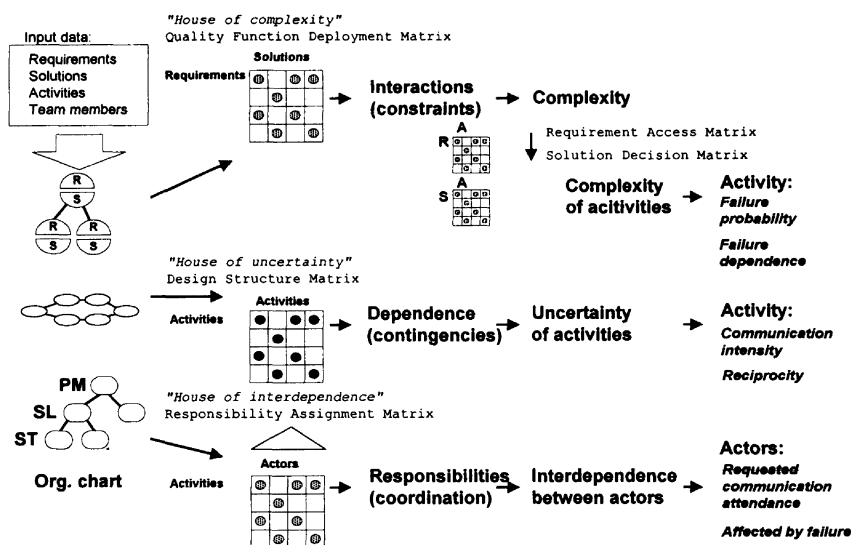


Figure 3.2 Overview of the load modeling framework

3.3 The house of complexity

To define coordination load, we describe the various interactions between project requirements and solutions in a QFD interdependence matrix [Hauser & Clausing 88]. We can use the resulting *house of complexity* to derive a relative distribution of the complexity of requirements and solutions in the project. Using Herbert Simon’s [58] notion of complexity as “the number of constraints an actor must simultaneously keep in mind while working”, we count the number of interactions between requirements and solutions to get complexity measures. The more requirements a given solution must contribute to satisfying, the more complex is the solution. Similarly, the number of solutions that contribute to a given requirement gives a measure of the complexity of the requirement.

For example, in figure 3.3 we see how the requirement for ‘Control processing’ (R1) interacts with the solution ‘External interface’ (S3), giving the value 1 for the interaction between the two. Note also, that the interaction between a requirement and its own solution is weighted by the number of sub solutions. Thus, for example the interaction between R1 and S1 has the value 2, since ‘Manifold system’ (S1) gives rise to two lower level requirements (R1.1 and R1.2, see figure 2.2). The same is done for interactions between higher level requirements and other solutions (e.g., interaction between R1 and S2 has the value 4, since the ‘Template structure’ solution gives rise to four lower level requirements).

Solution complexity may be viewed as a measure of the probability that actors producing a given solution will make errors when carrying out their work. Conversely, even if all solutions contributing to satisfy a requirement are in order, the customer may still not be satisfied, and requirement complexity is a measure of the probability of not satisfying the various requirements. As outlined in the next chapter, we use these measures during simulation of the design process.

Figure 3.3 show what we term “the house of complexity” for the SSS project.

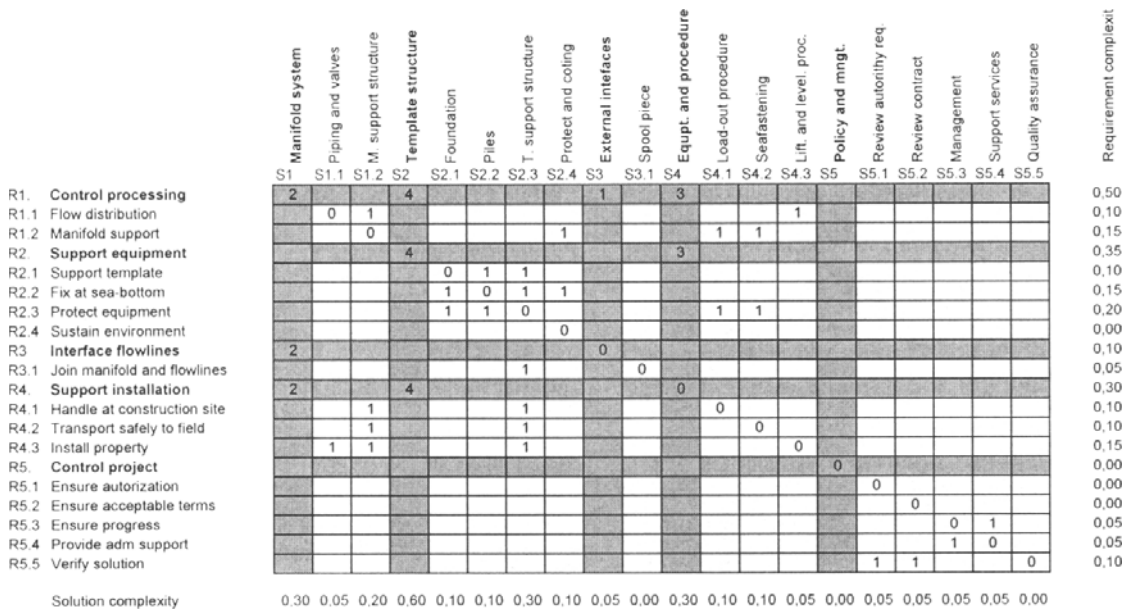


Figure 3.3 The house of complexity for the SSS project

3.4 Coupling the product to the process

Given measures of the complexity of requirements and solutions we relate those complexities to a resulting complexity of the various work process activities in the project. We may use so-called RequirementsAccess- and SolutionDecision matrices to transform the complexity of requirements and solutions into complexity distributions of the various activities by summing and weighting.

The solution complexity for a given activity, from the SolutionDecision matrix, indicates an *internal failure probability*, which is a measure of the chance of exceptions in producing design solutions (typically discovered in self-checks or peer reviews). In figure 3.4 “the SolutionDecision matrix” for the SSS project is shown.

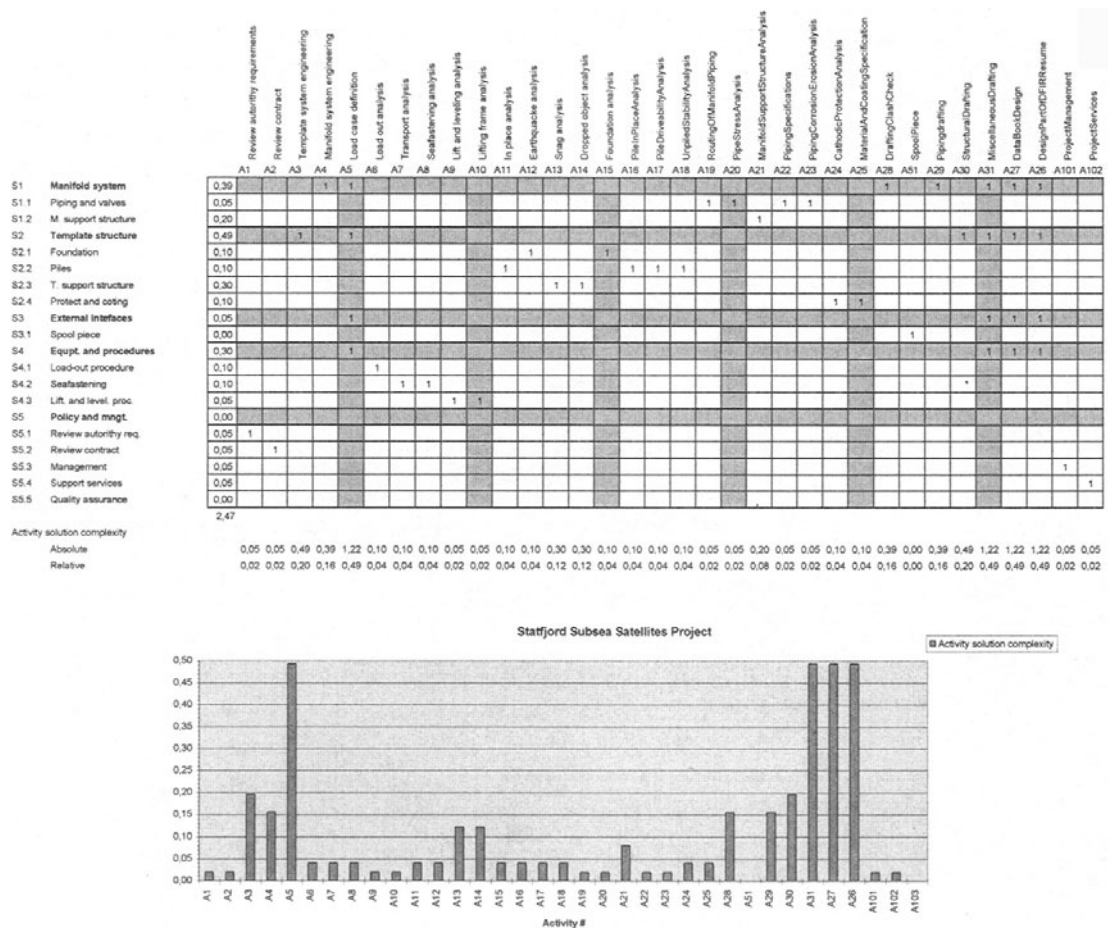


Figure 3.4 The Solution decision matrix for the SSS project

The RequirementsAccess matrix is similar to the SolutionDecision matrix, but the resulting requirement complexity for activities may be used to give an *external failure probability*, which gives a measure of the chance of non-conformance when carrying out work to satisfy customer requirements (typically discovered in project milestones, or client reviews).

3.5 The house of uncertainty

We next look at the production of, and need for, information by relating activities to activities in the same type of interaction matrix. This gives a so called “house of uncertainty”, which can be used to derive the relative distribution of uncertainty of activities. If we use Galbraith’s [73] notion of uncertainty as “a result of differences between the information which is needed to carry out a task and that which is available at the time the task is carried out.” The more information is needed, but not available, the more uncertain is the task.

Employing the Design Structure Matrices (DSM) technique [Gebala 88], we use a relationship matrix which places the project activities both along the rows and columns of the matrix. In the DSM notation a_{ij} means that activity j produces information which is needed by activity i . If we order the activity matrix according to order of execution, we see that any a_{ij} where j is larger than i (i.e., which lies to the right of the midline diagonal) represents information which is not available when it is needed. Thus we may sum all a_{ij} s where j is greater than i , to get a relative distribution of uncertainty for the various activities. Assuming that uncertainty gives rise to information needs we may use this uncertainty distribution as an indication of the required communication intensity of actors who are responsible for various activities. That is, actors who are responsible for activities with high uncertainty need to communicate frequently with actors responsible for activities which deliver delayed information.

Figure 3.5 shows what we term “the house of uncertainty” for the SSS project.

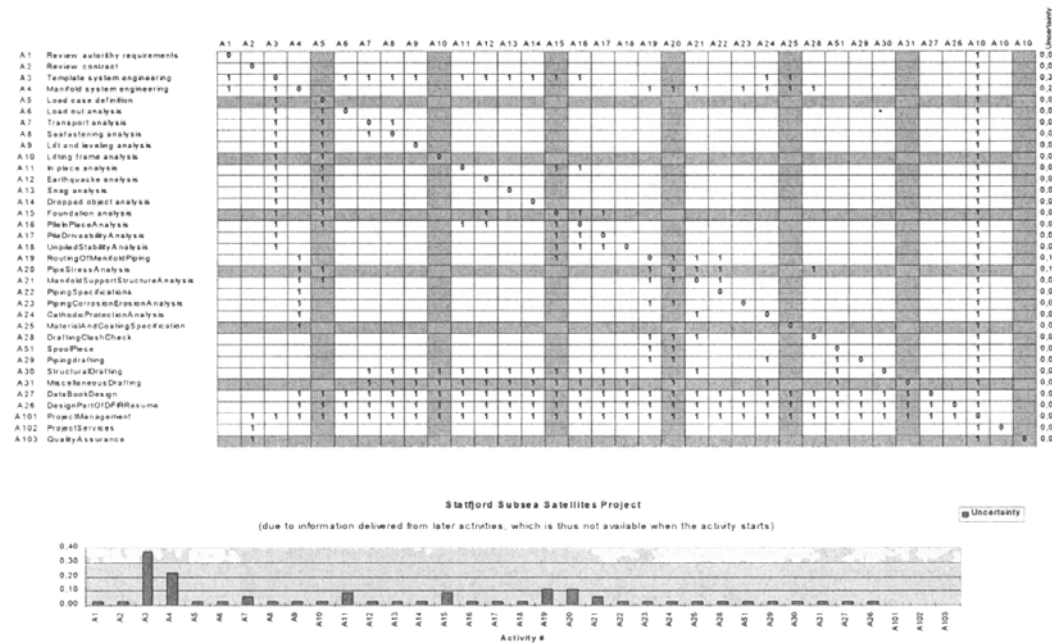


Figure 3.5 The house of uncertainty for the SSS project

We see, for example, how “Template system engineering” (A3) needs information from a series of activities which are carried out later in the process plan (e.g., A6 to A9). Thus, A3 has a high level of relative uncertainty (as shown in the graph in figure 3.5). As noted by Gebala [88] this may be used to optimize the design sequence by LU-decomposing the activity plan as far as possible to get a process with minimal uncertainty. However, we have not yet investigated the associated aspects of project design.

3.6 The house of interdependence

Figure 3.6 shows our tool for matching the responsibility of actors for activities, and relates this to the information flow between activities to find the interdependence relations between the various project team members. This house of interdependence shows which actors are responsible for given activities, and which actors need the information produced by those activities. We use the triangular interrelationship matrix between actors, at the “roof of the house”, to get a feel for the type of interdependence between actors. That is, whether they are, in James Thompson’s terms pooled, sequentially or reciprocally interdependent [Thompson 67]. What these terms mean can be seen in figure 3.6.

For example, the Project Manager is reciprocally interdependent with the Manifold Lead, since they both are responsible for activities from which the other needs information (e.g., A1 and A4). Similarly the Structures and Drafting Leads are only sequentially interdependent, since the Structures Lead needs information from the Drafting Lead’s activities but not vice versa. Also the Drafting and Support Leads do not need to share specific information, and are thus only pooled in their interdependence.

Thus, this tool tells us something about the required participation in information exchange (the various communication events) during project execution. Figure 3.7 shows what we term “the house of interdependence” for the SSS project.

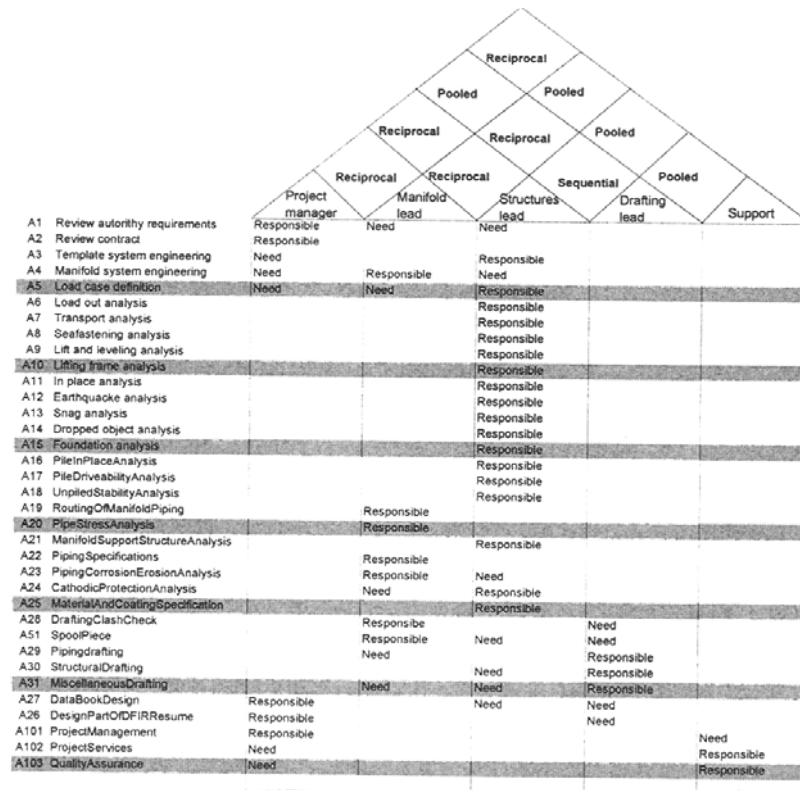


Figure 3.6 The house of interdependence for the SSS project

4 DISCRETE EVENT SIMULATION OF INFORMATION PROCESSING AND COORDINATION

In order to simulate our integrated project models we view project execution as an information processing task [Galbraith 73]. We represent the activities in the project plan, as well as the participants in the project team as objects, with attributes, holding data and functions, to describe their properties and behavior respectively. Between the various objects there are sets of relations to describe responsibility, supervision and communication.

4.1 The Virtual Design Team (VDT) discrete event simulator

Given the above information processing model of engineering design and the models of coordination load and capacity in projects, we can use them in the VDT [Cohen 92] to simulate project execution as a set of information processing tasks. VDT is a result of an ongoing project at Stanford University [Levitt 93], which have the aim of modeling and investigating various aspects of project team organization. In VDT we simulate project execution to get estimates of efficiency and process quality, using a set of stochastic (random number) process elements to model uncertainty in human decision making. Thus, by varying requirements, deliverables, plan, team, policies and preferences we can obtain predictions for the probable effect of proposed changes. VDT is implemented as an object oriented discrete event simulator where each processor (project team member) carries out work according to allocated responsibilities using a set of communication tools. The processing speed is determined by their own skill, experience and preferences, and the project policies for decision making in given situations.

Figure 4.1 gives an overview of input and output for analyzing information flow in the VDT. The input consists of a description of the load from the environment, the capacity of the project team, and behavior of the team. The load is described in terms of activities' work volume, failure probability and communication intensity. Team capacity is described in terms of their skill, craft specialization and experience, and team behavior is determined by a combination of policies (what should be done in given situations) and preferences (what is actually done in those same situations). The output from the simulation is project performance, defined in terms of the critical path duration, work volume (a substitute for project cost) and process performance in coordination (error handling and communication attendance) [Christiansen 93].

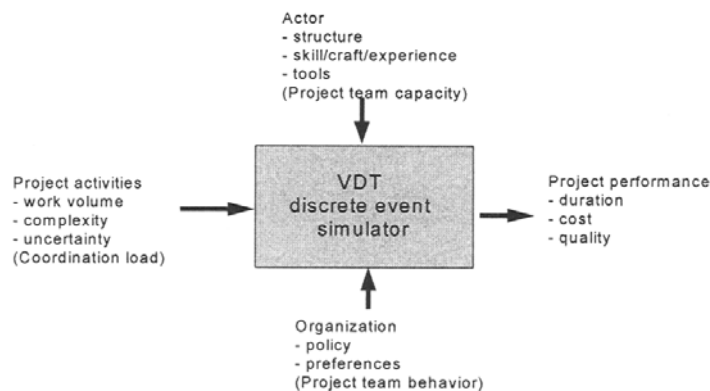


Figure 4.1 Information flow for the VDT simulation

Below we give examples of results from simulation of the SSS project in VDT, as obtained from the mean of a series of simulation runs with different random seeds for stochastic processes. The results show how a change in coordination policy (higher or lower value than the one used in the SSS project) is likely to affect performance. The simulation predictions are compared with predictions from contingency theory [Thompson 67] and predictions from the project manager (who planned and took part in executing the project). The results illustrate

- how the policy for *centralization* of decision making about non-conformances affect project duration and cost [Cohen 92], and process quality [Christiansen 93].
- how the match between policy for *formalization* of project communication and the organizational culture (matrix type) affect project duration, cost and process quality [Christiansen 93].

4.2 Centralization of decision making policies

Figure 4.2 shows simulation results for project duration, cost and verification quality, as a function of centralization in the SSS project, compared with predictions from contingency theory [Thompson 67] and the project manager of the project. Centralization in this context relates to "how high up in the hierarchy" decisions about exception handling "must travel" before reaching an actor with the authority to make a decision. These simulation results may be explained in terms of the model behavior, as summarized below.

For *duration* the expected behavior from contingency theory is based on the assumption that project managers have a more global view of different parts of the project, and thus will tend to prefer rework since they understand the potentially detrimental effect of ignoring failures in one activity on a number of dependent activities. Project team members, on the other hand, will often engage in local suboptimization of performance by ignoring and quick-fixing failures. In addition to this, decisions from managers will be delayed by other items in their "in-tray" (the manager's agenda). This causes higher centralization (more decisions by high level managers) to give both more rework, and more waiting time, and thus longer duration.

For *cost* the theoretical prediction is the same as for duration, based on the assumptions that managers favor rework, which increases the total volume of work carried out.

Verification quality is given by the ratio of reworked exceptions to all exceptions, and thus higher quality means that fewer exceptions are ignored. The simulation prediction of higher verification quality for higher centralization is therefore also consistent with both theory and the project manager.

We see from figure 4.2 how project performance results from rework decisions made by actors, and how these decisions are influenced by project policy for who makes decisions, the preferences of these decision makers, and coordination load on the decision makers. Moreover, the simulation results indicate that there was no universally "best" centralization policy for the SSS project. The most suitable policy depends on whether efficiency or quality has the highest performance, in which case one should choose either a decentralized or centralized policy. These simulation results show that the model behavior reflects this contingent nature of performance [Thompson 67] correctly, as predicted both from theory and real world experience. We claim that this is an indication of model correctness.

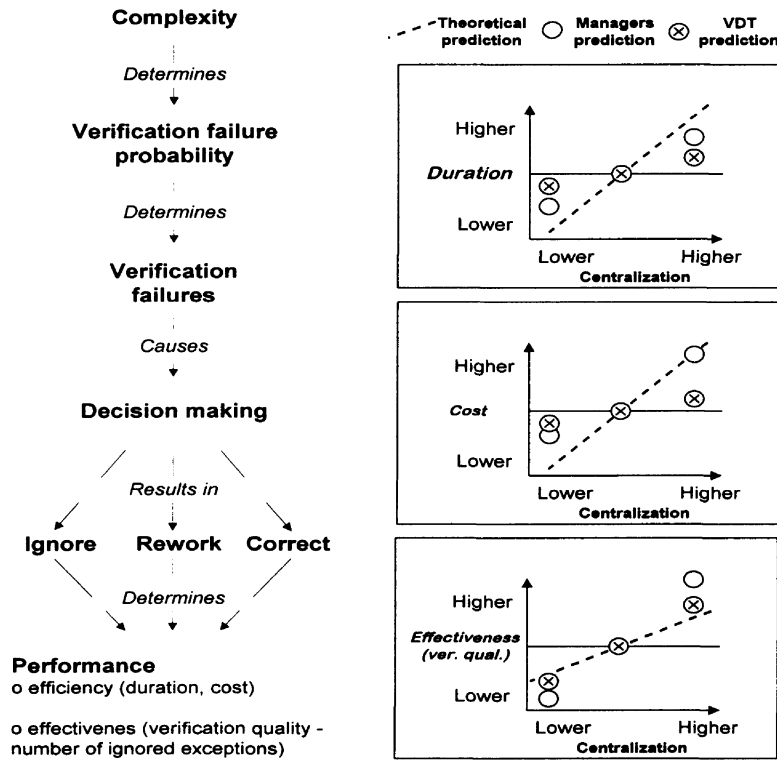


Figure 4.2 Project duration, cost and quality as a function of centralization in the SSS project

4.3 Formalization of communication policies

Figure 4.3 shows simulation results for project duration, cost and quality as a function of formalization in the SSS project, compared with predictions from contingency theory [Thompson 67] and the project manager of the project. Formalization relates to communication mechanisms used during project execution. Specifically, a formal communication policy mandates pre-scheduled project meetings with mandatory attendance by selected team members. Conversely, an informal communication policy relies on frequent face-to-face communication between physically co-located project team members. As above, we may explain the simulation results in terms of the model behavior.

For *duration* the prediction from contingency theory depends on whether or not the effect of learning is taken into account. Without learning, the assumption is simply that formal communication takes more time, and thus leads to longer duration and higher cost. With learning effects the predicted behavior is based on the assumption of implicit coordination and organizational learning, which allows a reduction in explicit communication for communication patterns that are well known to the team. Thus, both higher and lower formalization will give higher duration than the medium value (the actual pattern employed previously by the team). The learning explanation is consistent with the prediction from the project manager that project duration will increase for any change in formalization. The results from the simulation are at least qualitatively consistent with this explanation.

For *cost*, the predictions from contingency theory with learning and the predictions from the project manager are similar to duration. Again, the simulation results are qualitatively consistent.

Communication quality is given by the ratio of non-attended communications to all communications. The theoretical prediction is that project teams in project-oriented organizations will be more effective than their communication processes with an informal communication policy, which allows flexible coordination. The idea behind this is that frequent communication makes the effect of missing any given exchange very small, which allows for flexible "fluid participation" in communication, and which fits the cultural preferences of the project team. The basis for the project manager's prediction is the belief that an overly formal communication policy will not be able to handle the intense coordination required between project team participants responsible for highly concurrent activities (and will thus encourage team members to not participate in communication at all).

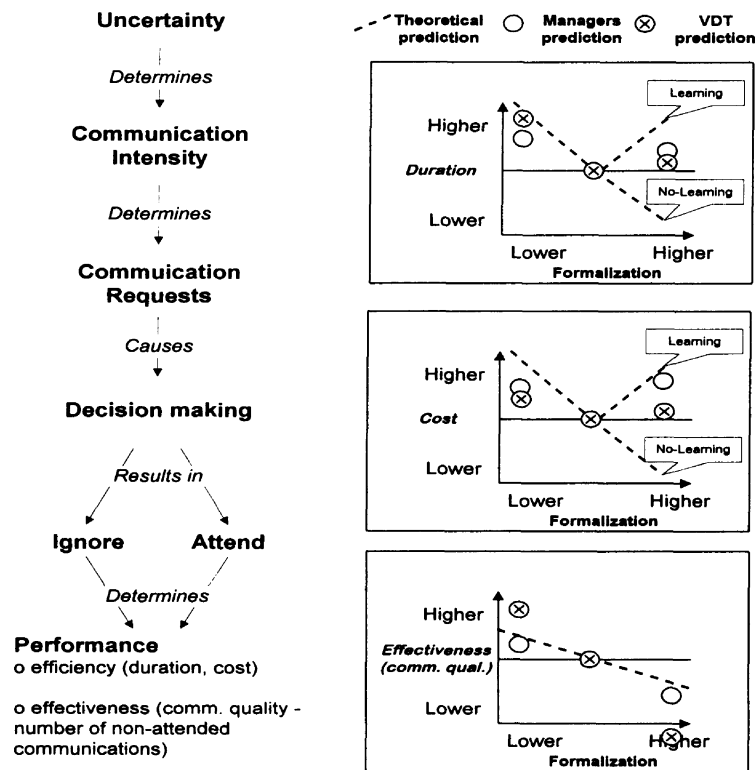


Figure 4.3 Project duration, cost and quality as a function of formalization in the SSS project

We see from figure 4.3 how project performance results from decisions made by actors about communication attendance, and how these decisions are influenced by the match between communication policy (formal vs. informal), the culture (preferences) of the project team (a strong matrix team in this case, preferring informal communication), and coordination load on the decision maker. Again project performance is contingent on a set of situational factors. There is "no best way" to formalize communication, since lower formalization, which is predicted to enhance communication effectiveness is also predicted to give higher duration and cost. Again we claim that model behaves "correctly" in displaying the contingent nature of performance [Thompson 67] as predicted both from theory and real world experience.

Both sets of results indicate how project performance is contingent on proper organization, and how differences in performance may be expected for different policies applied to different project teams. This is a result of model completeness and correctness according to purpose.

It is worth pointing out that both of the above results indicate the order of magnitude of qualitative change as selected input variables are altered. Because of a set of stochastic element in the VDT model, several simulations are necessary to obtain statistically stable results (mean and standard deviation). The simulation results are stable in the sense that changes in input variables produce a consistently larger change in output variables than the standard deviation of the mean of those same output variables. This is another aspect of model correctness.

Alternatively, we may study how performance is a function of changes in coordination requirements: the choice of different product solutions, division of the project into work packages (activities), assignment of responsibility between project team members. Another possibility is that we study the likely effects of training the project team. This extensibility is another aspect of model completeness (extensibility). Given that our model is similarly "correct" for these alternative aspect of project enterprise, we may use it to study the various trade-offs between alternative ways to plan, man and execute projects.

5 RELATED RESEARCH, AND FUTURE DIRECTIONS

In this section we make some comments on research areas which we think contribute to research on enterprise engineering. We also describe the current and future directions of our research.

5.1 Related research

Organization design, coordination science, axiomatic design and management policy all contribute to broadening the theoretical foundation for defining and studying enterprise. Total Quality Management and Business Process Reengineering put enterprise integration in a broader context, and operationalizes it for actual improvement in business, but has, in our view, lacked a unifying foundation so far. The ultimate challenge for enterprise integration is the systematic "building" and "operation" of its theoretical constructs. Below we comment briefly on some areas of research which we believe may give valuable contributions.

Organization design focuses on the task of *designing structure and processes of organizations*. Organization in this context is often defined as a system of roles (structure) and stream of activities (processes) designed to accomplish shared purposes [Robey 91]. This is closely consistent with our own views about how to describe the various enterprise dimensions.

Axiomatic design theory is a method which us the structure between requirements and solutions, as well as ideas about information content as a basis for *evaluation of design solutions* [Suh 90]. In other words, we use both design theory and information theory to focus on the effect of dependencies on system performance. Again, this is close to our own methodology to describe coordination requirements as dependencies in a set of matrix tools.

Coordination science is an attempt to develop a new area focusing on different aspects of *coordination*, as the study of dependencies. This has resulted in a typology of different types of dependencies [Malone 91] which seems a good starting point for studying how dependencies affect performance.

Management policy is focusing on *the management of the total organization*, with particular emphasis on its decisional behavior [Mintzberg 79]. The focus on organization structure in much of this research [Mintzberg 83] defines a promising point of departure for formalizing theories about interactions between organization structure and behavior.

Total quality management (TQM) focuses on a continuous improvement, with the ultimate vision that quality means doing the right thing the first time [Talley 91]. In TQM

implementation (incremental) change is important both on an organizational and individual level.

Business process reengineering (BPR) is the fundamental rethinking of business processes to achieve dramatic improvements in critical, contemporary measures of performance, such as cost, quality, service and speed [Hammer & Champy 93], identifying the enabling role of information technology in this process. BPR gives a description of the approach for change, which consist of the following steps; 1) building an *As-Is* model and analyzing it to identify opportunities, 2) designing a *To-Be* model and assessing predicted performance improvements and 3) implementing selected parts of the *To-Be* model [Davenport 93].

BPR and TQM are interrelated in the way that an innovation often is followed by a time of continuous improvement, which is then followed by an innovation, and so on.

Organizational development is the applied behavioral science discipline dedicated to improving organizations and the people in them through the use of the theory and practice of planned change. The focus is on developing (improving) both organizations and individuals. As stated in French & Bell [95] "to manage the culture of an organization in such a way that the goal and purposes of the organization are attained at the same time that human values of individuals within the organization are furthered."

It is essential that the field of enterprise integration be aware of the substantial efforts connected to organizational and individual change when it comes to implementation of organizational design solutions. Organizational Development (OD) focuses on how an enterprise can improve its capabilities [Galbraith 94] through creation of ideas, design of organizations and implementation, and supports the idea that it is possible to design organizations. Recently the increased focus upon change has led to an enhanced term "Organizational Development and Change (ODC)" [Golembiewski 94].

5.2 Ongoing and future work.

We are currently working on implementing various aspects of the model described above using a set of integrated tools in the MS Windows environment. Our Enterprise Development Tools (EDT) is being developed with the aim of defining, describing, analyzing and interpreting enterprise, and contains a set of tools for modeling and analysis.

The EDT tools, which are implemented on top of the Microsoft Windows desktop and make use of Object Linking and Embedding (OLE) and Dynamic Data Exchange (DDE), include -

- A *Definition* Tool using the Visio (TM Shapeware Corporation) object oriented graphics editor.
- A *Description* Tool using Excel (TM Microsoft).
- A *Load Calculation* tool using the Powersim (TM ModellData). system dynamics tool.
- An *Analysis* tool using the VDT simulator on top of Kappa (TM Intellicorp).
- A *Results Presentation* tool using Excel (TM Microsoft).
- A *Data Storage* tool using Access (TM Microsoft).
- A *Reporting* tool using Word (TM Microsoft).
- A *Data Interchange* tool based on OPDL (Organization and Project Description Language) [Jin 95].

Collectively, these tools enable us to investigate various aspects of enterprise design and development systematically. VDT represents one (and so far the only) analysis tool, coupled with the various preprocessors for definition and description of enterprise by means of the OPD format and OPDL language. Similarly, VDT is also coupled to our OPD (MS Windows based) presentation tool by the same OPDL.

Further development of VDT is taking place both at Stanford University in the US and at Det Norske Veritas Research in Norway. Perhaps the most interesting work under way is work that will widen the scope of projects that may be studied by simulation beyond the assumption of routinized activities, as well as initial investigation into adaptation and learning [Levitt et. al 94].

The former will allow us to model knowledge work, the latter to model and simulate knowledge workers.

Moreover, we are studying *the implementation* of proposed organizational changes in different projects to understand what is needed to make enterprise engineering and integration work in practice, and thus make engineering projects more efficient and effective.

In our future work we will aim to enact our project enterprise model as part of the actual project planning and project control during execution (category III).

5.3 Acknowledgments

The background for this work carried out as part of the VDT project at Stanford University, and additional work carried out in DNV Research, the research company within Det Norske Veritas ship classification society. The VDT project at Stanford was supported by National Science Foundation, Division of Information, Robotics and Intelligent System, Grant Number IRI-9725441 and by Seed Research Grants from the Center for Integrated Facilities Engineering, Stanford University. We also would like to thank Statoil, Det Norske Veritas Industry, and in particular project manager Dan Kyrre Stangeby of DNVI's engineering services, Veritech.

6.0 REFERENCES

- Christensen, L. C., Christiansen, T. R. & Syvertsen, T. G.** (June 1994) *What is needed to model human enterprise - elements of a unified enterprise engineering model*. Presented at TIMS cluster on BPR, Anchorage ALASKA, Published as DNV Research report no. 94-2020, Oslo
- Christensen, L. C., Johansen, B. W., Midjo, N., Onarheim, J., Syvertsen, T. G. and Totland, T.** (1995) *Enterprise Modeling - Practices and Perspectives* in proceedings of the 9th ASME Engineering Database Symposium, Boston, Massachusetts
- Christiansen, T.R.** (Oct 1993) *Modeling efficiency and effectiveness of coordination in engineering design teams*. unpublished Ph.D. dissertation, Civil Engineering Department, Stanford University, Published as DNV Research report no. 93-2063, Oslo
- Christiansen, T.R. and Thomsen, J.** (Sept. 1993) *CAESAR - an architecture for enterprise modeling in the AEC industry*. Published as DNV Research report no. 94-2019, Oslo
- Cohen, M.D., March, J.G. & Olsen, J.P.** (1972) *A garbage can model of organizational choices*. American Sociology Quarterly, Vol 17 no. 1, 10-25
- Cohen, G.P.** (July 1992) *The Virtual Design Team: An object oriented model of information sharing in project design teams*. unpublished Ph.D. dissertation, Civil Engineering Department, Stanford University, Stanford
- Daetz, D.** (1990) *Planning for customer satisfaction with quality function deployment*. Proc. 8th international conference of the ISAQ - Jerusalem, Jerusalem
- Davenport, T. H.** (1993) *Process innovation - reengineering work through information technology*. Harvard Business School Press, Boston, Mass.
- French, W. & Bell, C. H. jr.** (1973) *Organization development - Behavioral science interventions for organization improvement*. Prentice Hall, Inc., Englewood Cliffs

- Galbraith, J.** (1973) *Designing complex organizations*. Addison-Wesley, Reading, Mass.
- Galbraith, J.** (1994) *Competing with Flexible Lateral Organizations*. Addison-Wesley, Reading, Mass.
- Gebala, D. & Eppinger, S. D.** (Sept. 1991) *Methods for analyzing design procedures*. Third Intl. ASME Conf. on Design Theory and Methodol., Miami, Florida
- Golembiewski, R. T.** (July 1994) *Organizations development and change as prismatic*. The International Journal of Organizational Analysis 1994, Vol.2, No.3 (July), 229-235
- Hammer, M. & Champy, J.** (1993) *Reengineering the corporation*. Nicholas Brealey Publishing, London
- Hannan, M.T. & Freeman, J.** (1982) *The population ecology of organizations*. American Journal of Sociology, pp. 929-964,
- Hauser, J. R. and Clausing, D.** (1988) "*The House of Quality*", Harvard Business Review May-June 1988, Boston
- Jakobson, I., Ericsson, M. & Jacobson A.** (1994) *The object advantage, business process reengineering with object technology*. Addison-Wesley Publishing Company, New York
- Jin, Y., Levitt, R.E., T.R. Christiansen & J. Kunz** (1995) *The Virtual Design Team: A computer simulation framework for studying organizational aspects of concurrent design*, Simulation, Vol 64 no. 3 (March), 160-174
- Kleinman, D. L.** (1990) *Coordination in human teams: theories, data and models*. in Proc. of the 11th IFAC World Congress, Talinn, Estonia, 159-164
- Levitt, R.E., Jin, Y., Oralkan, G.A., Kunz, J. & Christiansen, T.R.** (1995) *Computational enterprise models: Toward analysis tools for designing organizations*. CIFE working paper no. 36, Stanford University, Stanford
- Malone, H.& Crowston, K.** (April 1991) *Towards an Interdisciplinary Theory of Coordination*. MIT Sloan School Working Paper # 3294-91-MSA, Boston
- Mintzberg, H.** (1979) *The Structuring of Organizations*. Prentice-Hall, Englewood Cliffs, New Jersey
- Mintzberg, H.** (1983) *Structuring in fives: Designing Effective Organisations*. Prentice-Hall Inc., Englewood Cliffs, New Jersey
- Robey, D.** (1991) *Designing organizations*. Irwin, Homewood Illinois
- Scott, W.R.** (1987) *Organizations: Rational, Natural and Open systems*. Prentice-Hall Inc., Englewood Cliffs, NJ,
- Simon, H.A.** (1958) *Administrative behavior*. Macmillan, New York
- Simon, H.A.** (1969) *The Sciences of the Artificial*. MIT Press, Cambridge, Mass
- Suh, N. P.** (1990) *The principles of design*. Oxford University Press, New York
- Thompson, J.** (1967) *Organizations in Action: Social Science Bases in Administrative Theory*. McGraw-Hill, New York
- Willems, P.** (July 1988) *A functional network for product modeling*. PLI-88-16, IBBC-TNO, Netherlands
- Williams, T. J. et.al** (1993) Architectures for integrating manufacturing activities and enterprises, in *Information infrastructure systems for manufacturing* (Yoshikawa, H. & Gossenaerts, J. eds.), Proceedings of the JSE/IFIP/ TC5/WG5.3 Workshop on Design of Information Infrastructure Systems for Manufacturing, DIISM'93 in Tokyo, North-Holland, Amsterdam

Lars Christian Christensen is a Doctoral Student in the Department of Civil Engineering at the Norwegian Institute of Technology, currently visiting the Department of Civil Engineering at Stanford University, where he is working on developing computer tools to model and analyze project enterprise.

Tore Christiansen is working as Principal Research Engineer, in the Department for Information Systems of DNV Research, the strategic research company of Det Norske Veritas ship classification society. He is educated in Aeronautical engineering from the University of London and the Massachusetts Institute of Technology, and obtained his PhD at Stanford University, Department of Civil Engineering in 1993, working in the VDT project team. His current work involves enterprise modeling and the use computer tools to improve the efficiency and effectiveness of project organizations.

Yan Jin is Research Associate at Department of Civil Engineering, Stanford University. He earned his PhD degree in Naval Engineering from the University of Tokyo in 1988. Since then Dr. Jin has been doing research on knowledge based planning systems, distributed problem solving, organizational modeling, and their application to marine traffic control, collaborative design and concurrent engineering. His work on the Virtual Design Team integrates his current interests in organizational modeling and coordination science.

John Kunz is Senior Research Associate at the Center for Integrated Facility Engineering in the Department of Civil Engineering at Stanford University, where he teaches and does research on non-numeric modeling in engineering. He served as the Intellicorp Chief Knowledge System Engineer and Director of Manufacturing Applications, and has led AI systems application development in diverse areas including design, plant control and project management. The Virtual Design Team research has extended his interests in modeling from products and processes into social systems.

Raymond Elliot Levitt is Professor of Civil Engineering in Stanford's Construction Management Program and Associate Director of the Center for Facility Engineering. Dr. Levitt earned MS and PhD degrees in Civil Engineering at Stanford, and a BSCE at the University of Witwatersrand. He was on MIT's Civil Engineering faculty from 1975 to 1980, before moving on to Stanford. The Virtual Design research, in which Dr. Levitt is currently engaged integrates his organization theory and artificial intelligence interests.