Journal of Organizational Computing and Electronic Commerce

ISSN: 1091-9392 (Print) 1532-7744 (Online) Journal homepage: http://www.tandfonline.com/loi/hoce20

Modeling and Simulating Coordination in Projects

Lars Chr. Christensen , Tore R. Christiansen , Yan Jin , John Kunz & Raymond E. Levitt

To cite this article: Lars Chr. Christensen , Tore R. Christiansen , Yan Jin , John Kunz & Raymond E. Levitt (1999) Modeling and Simulating Coordination in Projects, Journal of Organizational Computing and Electronic Commerce, 9:1, 33-56, DOI: 10.1207/s15327744joce0901_3

To link to this article: https://doi.org/10.1207/s15327744joce0901_3

Published online: 18 Nov 2009.

JOURNAL OF ORGANIZATIONAL COMPUTING
AND ELECTRONIC COMMERCE 9(1), 33–56 (1999)

Modeling and Simulating Coordination in Projects

Lars Chr. Christensen
Tore R. Christiansen
Det Norske Veritas Research
Høvik, Norway

Yan Jin
Department of Mechanical Engineering
University of Southern California

John Kunz
Raymond E. Levitt
Center for Integrated Facilities Engineering
Stanford University

A main challenge in managing projects is identification and understanding of interactions between subtasks. These interactions give rise to dependencies between activities in the project plan. The resulting interdependence between members of the project team requires them to coordinate extensively during project execution. Project managers need a systematic methodology for describing and analyzing coordination requirements on project teams. This need is not met in traditional tools for project planning and scheduling. In this article, we describe an object-oriented framework for modeling projects and a methodology for formalizing these models such that they can be used for discrete event simulation of information processing and coordination in project execution.

Our modeling framework represents projects in terms of objective (requirements), product (solution deliverables), process (activities), and organization (participants and relations). We then use matrix techniques to explicate the constraints between project requirements and deliverables (complexity), the contingencies in information flow

The background for this work was a part of the Virtual Design Team (VDT) project at Stanford University, California and additional work was performed at DNV Research, the research company within the Det Norske Veritas, Høvik, Norway ship classification society. The VDT project at Stanford was supported by the National Science Foundation, Division of Information, Robotics and Intelligent Systems (Grant IRI–9725441) and by Seed Research Grants from the Center for Integrated Facilities Engineering, Stanford University. This article is based on a presentation by the second author at INFORMS–95, Los Angeles, April 1995. We wish to thank an anonymous reviewer for many insightful suggestions for improvement. Further development of VDT is taking place both at Stanford University and at DNV Research.

Correspondence and requests for reprints should be sent to Tore R. Christiansen, Det Norske Veritas Research, P.O. Box 300, N–1322 Høvik, Norway. E-mail: tore.christiansen@dnv.com

between activities (uncertainty), and the resulting coordination requirements between project team members. The model and coordination measures can be used as input for simulation of project execution and give predictions for the probable effects of carrying out proposed changes in planning and managing projects.

To illustrate how enterprise modeling and analysis can inform project planing and execution, we apply our framework and methodology to model and simulate a simplified project for development of hydraulic systems. Our simulation results demonstrate how project performance is contingent on the fit between the project policies and the objectives and preferences of the project team.

enterprise modeling, project management, coordination, information processing, quality function deployment, simulation

1. INTRODUCTION

The objective of this article is to demonstrate how modeling and analysis can be used to understand project enterprise and how such understanding can be turned into performance predictions. In the article, we give an overview of a modeling methodology [1] for explicating coordination requirements in projects [2] and links the resulting model to analysis tools for simulating information processing and coordination during project execution [3–5].

Our motivation for modeling and analyzing projects is an increasing demand for effective and efficient project management [6]. Projects are becoming more difficult to plan and manage, with demanding customers, tight budgets and schedules, complex technology, and project teams that work concurrently in different locations. Consequently, project managers need methods and tools to help them make the right decisions during project planning and execution. However, such decision support tools must be based on true understanding of the decision situation, acquired by careful description and analysis. If we can develop models that consistently represent relevant aspects of the selected model domain and correctly re-create observed behavior in simulation, we believe that a combination of model building and simulation will improve understanding and produce useful management tools.

1.1 A Model-Based Approach to Managing Coordination Requirements

To develop methods and tools, we define *project enterprise* as "an *organization*, carrying out some (set of) *process(es)* to create *products* which satisfy predefined *objectives* in a given environment" ([7], p. 10). Based on this definition, we model the project team, plan, deliverables, and requirements. We take an information processing view of project execution and define a methodology for explicating the associated coordination. This methodology uses matrices to identify and quantify dependencies between different parts of requirements, deliverables, activities, and team members. The project model and associated dependencies can be inputted to the Virtual Design Team (VDT) discrete event simulator [3]. For a given project, the simulation produces measures of duration, cost, and quality. Thus, the simulation results can be used for assessing the effect of deploying different project teams, executing different project plans, and managing different coordination policies.

Our methodology provides (a) a framework that can be used by project participants to develop a common reference frame facilitated by shared models [8] of their work process, including deliverables, task breakdown structure, responsibilities, and several kinds of interactions; and (b) a formal model that can be analyzed by enterprise simulation tools such as the VDT [3] to generate specific performance predictions and diagnose bottlenecks and other problems in proposed project configurations. In combination, these effects have the potential to significantly improve the efficiency and effectiveness of project enterprise.

In the following sections of the article, we give an overview of our approach. Details of various aspects can be found elsewhere [1–3, 5, 9]. In Section 2, we outline our framework for modeling project enterprise. In Section 3, we describe modeling of coordination requirements. In Section 4, we explain how these requirements are used in discrete event simulation of project execution and give examples of typical simulation results. In Section 5, we discuss relevant research. Finally in Section 6, we outline planned extensions.

2. A FRAMEWORK FOR DESCRIBING PROJECT ENTERPRISE

Project enterprise consists of an assigned team working together for a planned period of time to deliver according to specification. To enhance understanding, a consistent model of project enterprise must therefore address why we act (requirements), what the result of action is (deliverables), when and how we act (activities), who acts (team), and where we act (environment). Our framework [2] represents project enterprise in terms of objectives, product, process, and organization (OPPO) as well as the various dependencies that exist within and between them. In Section 2.1, we develop a simplified description of a hydraulic system development (HSD) project for the offshore oil industry using the OPPO-model framework.

2.1 Describing Objectives and Products

To represent project deliverables we integrate the description of objectives and products using functional decomposition. Our so-called requirement function solution (ReFuSo) diagram is based on the functional unit technical solution (FUTS) technique [10], which matches functional units (FU) with corresponding technical solutions (TS). In the ReFuSo, we view design as a two-step process. Conceptual design transforms operational and performance requirements (Re objects) to a corresponding functional description (Fu objects). Detailed engineering then transforms this functional description to detailed solution specifications (So objects) for construction or procurement. This gives a description of the requirement structure, the function structure, and the topological structure of components, subassemblies, and assemblies in the product.

Hydraulic systems serve a variety of control tasks on offshore oil platforms subjected to a range of functional and operational requirements and serving a number of different users. Figure 1 illustrates how a requirement for hydraulic energy is

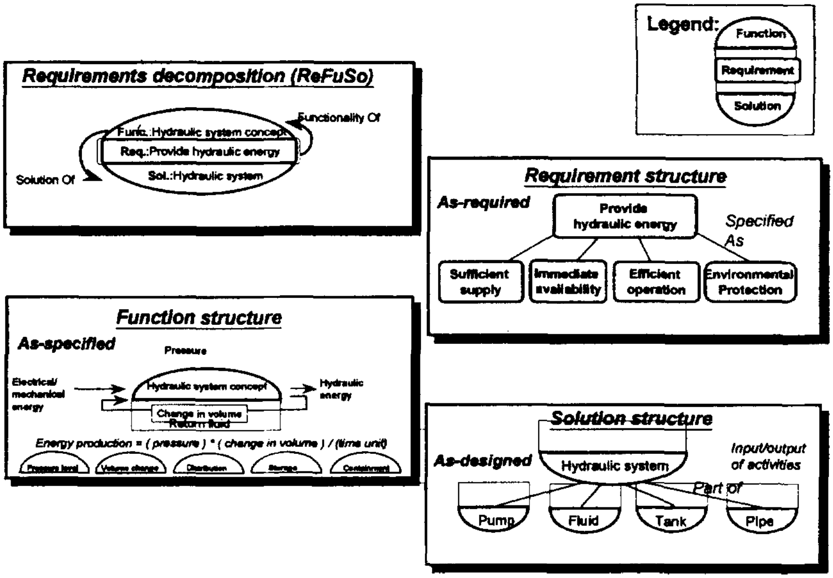


Figure 1. Requirements, function, and solution model structures for the hydraulic system development project.

of different users. Figure 1 illustrates how a requirement for hydraulic energy is met by an abstract hydraulic system concept, which is solved by the actual hydraulic system. The requirement structure decomposes the overall hydraulic energy requirement into more detailed requirements for supply, availability, and efficiency. The function structure describes the function and behavior of the hydraulic design concept and decomposes the functional structure into a set of functions for production, storage and distribution of energy, and environmental protection. The topological component structure describes the technical solution as a decomposition hierarchy of hydraulic system components, such as pump, tank, and piping.

In actual projects the decomposition is continued until a suitable level of detail has been reached to specify characteristics for design and procurement. In the example presented here we wish only to illustrate how we describe objectives and products, and have (arbitrarily) terminated the breakdown at an artificially high level of detail. In the next section, we use this description to identify product interactions and resulting needs for coordination. We point out that our representation of objectives and products relates required and realized characteristics of the project deliverables. The difference between them is an important part of the product performance of project enterprise.

2.2 Describing the Process and Organization

Describing the process dimension involves representing the activities in the project plan as well as their work volume and successor relations (the order in which they are planned to be executed). Figure 2 shows a typical life cycle for hydraulic sysscribing processes [11]. Our process model is made up of activity objects and precedence relations. The work volume is represented as an attribute of each activity object. We also represent various coordination policies (for command, control, and communication) as attributes of an overall process object. To cover the complete life cycle of project deliverables we use a simplified, generic life cycle model in which the various life cycle phases are represented as sequential activities, including conceptual design, detailed engineering, approval, fabrication and installation, commissioning, operation, maintenance, and decommissioning. Each activity (phase) contains a number of concurrent or overlapping activities.

In the same manner that the difference between objectives and product deliverables defines product performance, the difference between the process plan and execution defines process performance in terms of duration and person cost (efficiency).

Describing the organization involves representing the various project team members (actor objects). We describe actors in terms of their craft, skill, experience, preferences, and relationships between them, including the formal hierarchy of command and control. We also represent the responsibility relations between actors and various activities, and the communication relations due to these responsibilities. Figure 2 shows the project team and their relation to the project plan. In the same manner that the difference between objectives and product deliverables defines product performance, the difference between planning and execution defines aspects of project efficiency, such as duration with respect to schedule (process per-

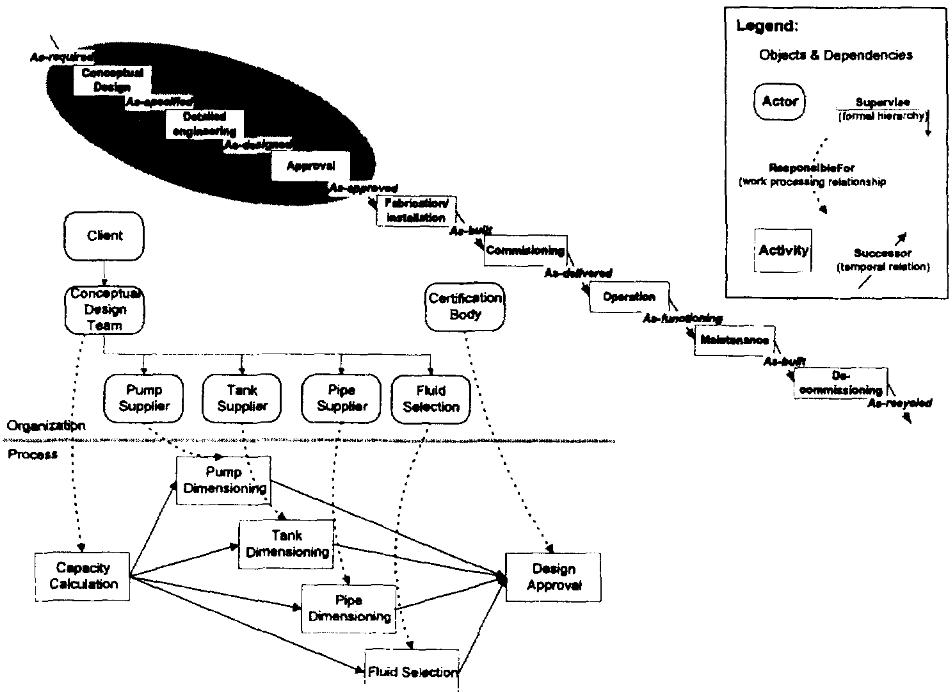


Figure 2. Project plan and project team model structures for the hydraulic system development project.

formance). The difference between project policies and personal preferences of project team participants will determine how planned action is translated into actual behavior. This determines organizational performance, which constrains process and product performance.

2.3 Consistency Between Model Purpose and Representation

We have already 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 products, processes, and organizations. We view model consistency as the ability of the model to capture differences between ideal and real situations (i.e., planned action vs. actual behavior). This requires us to consider action in project enterprise according to different views of causal logic.

A rational systems view of enterprise [12] explains project execution according to a logic of intention [13]. 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 is designed to carry out the process (which creates the products, which satisfies the objective). Thus, project enterprise is seen as a rational means for fulfilling stated objectives. A natural systems view of enterprise [12] explains project execution according to a logic of implication [13]. Enterprise is defined by some set of individuals (the organization). Between organizational members there exists a mix of rational and irrational relations that determines what processes can and cannot be carried 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 these two views explain project enterprise in full. Real actors have limited rationality [14], and causality must be understood in terms of both intention and implication. We think of this as a dual explanation of project execution. The consequence of this duality for enterprise modeling is a requirement that the model representation and reasoning must allow description of the differences between intended and implied action and include performance metrics for the difference between planned action and actual behavior. Our coordination load model attempts to address these requirements by supporting identification and explication of major reasons for deviation between project plans and actual execution.

In Figure 3, we illustrate how project execution can be viewed in terms of our OPPO-model framework. Starting from a given set of objectives, project enterprise proceeds by definition, identification, and assignment of a set of dependencies (*coordination requirements*), and subsequent execution to handle these dependencies according to defined policies (*coordination mechanisms*). Project performance may be viewed as a result of the alignment between project coordination policies and the preferences of project team members and can be assessed by comparing the realized and desired solutions. The project can then be evaluated by comparing performance to the objective. This comparison between achievement and aspirations

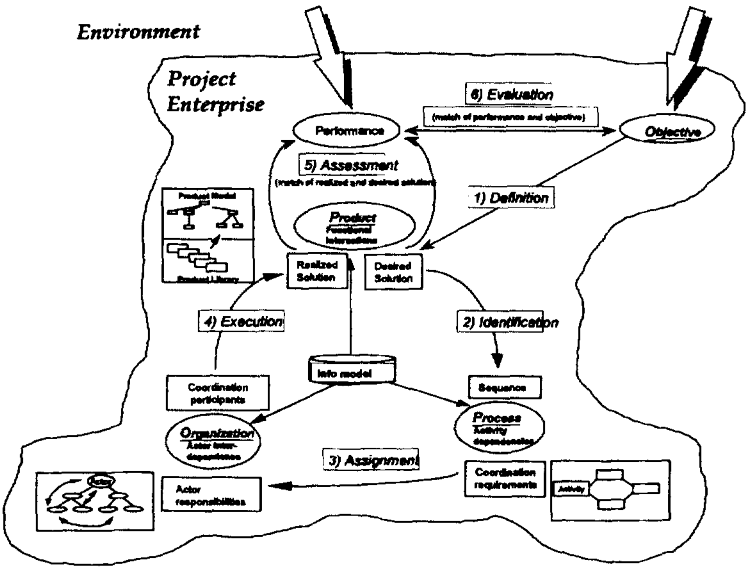


Figure 3. An overview of the enterprise modeling framework.

[13] will influence both. In addition, both performance and objectives are influenced by the environment.

The definition in the OPPO modeling framework can be used to identify coordination requirements and assign them to project team actors. This forms a basis for execution, assessment, and evaluation.

3. A METHODOLOGY FOR MODELING COORDINATION REQUIREMENTS IN PROJECTS

In this section, we describe our methodology for identifying and quantifying coordination requirements based on the model of project enterprise developed in Section 2.

3.1 An Information Processing View of Coordination in Engineering Design

To describe coordination [15] in projects we use an analogy between organizational and physical structures. Both physical and organizational structures may be thought of in terms of elements with given material properties, connected by nodes in a given configuration. Both are subject to load from their environment, and for both the capacity to meet this load is determined by their material properties and configuration. For physical as well as organizational structures, the match between required and realized behavior under load determines the performance of the structure.

We take an information processing view of project execution in terms of a set of processors (actors) who work by processing information (tasks) to complete activities in the project plan. In addition to work arising from planned project activities, we model coordination items arising due to various types of dependency objectives, product deliverables, and process activities. We define *coordination load* among information processors in terms of the demand for their attention when processing information. The more coordination items due to causal and informational dependencies in their activities, the more coordination to be attended, and the higher the load. Thus, coordination load is a function of the requirements, selected solutions, project plan, and allocation of responsibility. Similarly, *organizational capability* is given by the sum of the ability of actors (craft, skill, and experience), the processing capacity of the team (manpower, structure, and tools), and the coordination capacity of the team. The latter is determined by project policies and actor preferences for handling coordination. For a given project enterprise, the match between coordination load due to project requirements and the team's capability to meet that load will significantly influence performance.

Given these organizational mechanics, our coordination load model attempts to define and operationalize measures that are important for determining the performance of real projects. Traditional project planning assumes an ideal situation where different parts of the project deliverable are uncoupled, so that an error in one part will not affect any other part. Another traditional assumption is that project activities are sequenced so that all necessary information is available when required. In reality these assumptions are very seldom satisfied. Consequently, delays, cost overruns, and poor quality occur due to lack of information and error propagation. Experienced project managers account for this heuristically in their planning and scheduling. However, most project plans are still optimistic, leading to frequent disappointment [16].

3.2 Interactions and Complexity Measures

To identify dependencies in the project deliverables, we describe the various interactions between project requirements and solutions in a quality function deployment (QFD) interdependence matrix [17–19]. In the QFD notation any matrix element a_{ij} represents an interaction where solution j affects requirement i. That is, the solution needs to satisfy the requirement, and any change to the requirement may necessitate a corresponding change in the solution. It follows also that any error in the solution may imply that the requirement is not satisfied.

We can use the interaction matrix as a house of complexity to calculate project-specific measures of the complexity arising from coupling in the project task [19]. Using Simon's notion of complexity as the number of constraints an actor must simultaneously keep in mind while carrying out a task [20], 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 the solution. Thus, solution complexity is a measure of the probability that actors producing the solution will make errors when carrying out their work. Similarly, the number of solutions that contribute to a given requirement gives a measure of the complexity of the requirement. Even if all solutions contributing to satisfy a requirement are in order, the customer may still not be satisfied. We use the requirement complexity as a measure of the probability of failure to sat-

House of Complexity (Constraint function deployment)

Requirements \ Solutions	Hyd. sys. con.	Press. level	Vol. change	Distribution	Storage	Containment	Requirement complexity	
Provide hydraulic energy	1	1	1	1	1	1	6	1.00
Sufficient supply	1				1		2	0.33
Immediate availability	1	1	1	1			4	0.67
Efficient operation	1	1	1	1		1	5	0.83
Safe and environmentally friendly	1	1		1		1	4	0.83
Solution Complexity Absolute	5	4	3	4	2	3		
Relative	1.0	0.8	0.6	0.8	0.4	0.6		

For each requirement:
Complexity = weighted sum of interactions with solutions
For each solution:
Complexity = weighted sum of interactions with requirements

Scaled by number of requirements (i.e., divide by 5)

Scaled by number of solutions (i.e., divide by 6)

Figure 4. Interaction and complexity measures for the hydraulic system development project.

isfy various requirements. Figure 4 shows the house of complexity for the HSD project.

Figure 4 shows how the top-level requirement to provide hydraulic energy interacts with all functions (value 1 in all cells of the first row of the interaction matrix), giving the value 5 for the requirement complexity, or 1 if scaled by the number of solutions. The function pressure level must meet three lower level requirements (immediate availability, efficient operation, and safe and environmentally friendly) in addition to the top-level requirement. This creates a functional complexity of 4, or 0.8 if scaled by the number of requirements. In the matrix we have chosen the value 1 for all interactions. In the standard application of QFD to product design [18] these interactions often have different weights depending on the relative strength of the interaction. However, we do not yet have enough experience from application to projects that we can meaningfully derive such weights.

From this example it is clear that the aggregation of interactions represents a serious simplification. When adding interactions, no account is taken for the fact that higher level interactions are simply aggregated abstractions of lower levels. This may be accounted for by several schemes [21]. However, application of this method to derive complexity measures from engineering design projects for electrical power supply [22] and offshore field development [23] indicates that the resulting inaccuracy is not critical with respect to the overall purpose of indicating where and how complexity arises in the project deliverable.

3.3 Information Flow and Uncertainty Measures

We next use the same type of interaction matrix to describe dependencies due to production and consumption of information when carrying out the various project activities. Employing the design structure matrices (DSM) technique [24–26] we list the project activities both along the rows and columns of the interaction matrix. In

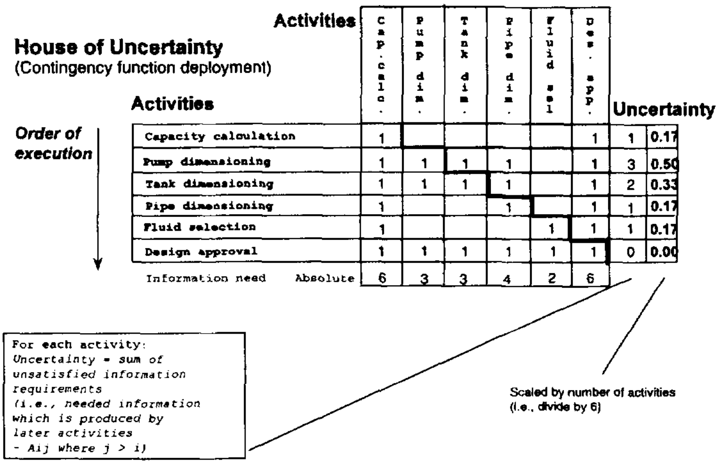


Figure 5. Information flow and uncertainty measures for the hydraulic system development project.

the DSM notation a_{ij} means that activity j produces information needed by activity i. If we order the DSM interaction matrix so that activities are listed according to their order of execution, we see that any a_{ij} where j is larger than i (i.e., to the right of the midline diagonal in the matrix) represents information that is not available when it is needed.

We use Galbraith's [27] 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" (p. 5). That is, uncertainty arises from lack of necessary information. The more information is needed but not available when carrying out an activity, the more uncertain is the activity. Thus, we may sum all a_{ij} where j is greater than i to get a relative measure of uncertainty for various activities.

The DSM interaction matrix may be thought of as a house of uncertainty that can be used to derive the distribution of uncertainty of different activities. Assuming that uncertainty gives rise to need for communication, we use this uncertainty distribution as an indication of the required communication intensity between actors who are responsible for various activities. Figure 5 shows the house of uncertainty for the HSD project.

In Figure 5, we see how the pump dimensioning requires information from both tank and pipe dimensioning, whereas tank dimensioning requires information from pipe dimensioning. Because pipe dimensioning is carried out after tank dimensioning and pump dimensioning, uncertainty is created. We also see how all activities depend on information from design approval (i.e., whether they satisfy given criteria for safe operation). For illustration in this example, we assume that the exact approval criteria are not known a priori. Because design approval is the last activity it introduces uncertainty for all other activities. Consequently the uncertainty of, for example, pump and tank dimensioning has the value 3 and 2, or 0.5 and 0.33 if scaled by the number of activities.

It is evident that the usefulness of DSM to derive uncertainty measures depends largely on the activity description. If the project plan describes activities at a level that is too high, the information flows that can be identified between activities will probably not be meaningful in describing the real communication requirements of actors in the project team. If activities are defined for the complete project duration, they will both start before and end before all other activities, and any information needed in them would seem to be produced by activities that start later. This would indicate a level of uncertainty that may not be consistent with the actual use of information during project execution. The solution to this problem of representation is to detail such activities further.

As noted by Gebala and Eppinger [24, 25], the DSM technique may be used to optimize the sequencing of project activities, by LU-decomposing the activity plan (lower upper decomposing the DSM matrix by rearranging the sequence of activities) as far as possible to get a process with minimal uncertainty. This would result in a DSM matrix where most of the matrix elements are located below the leading diagonal (representing information that is available when it is needed). So far our approach has been to describe projects where scheduling has already been determined, and we have no experience in using matrix techniques prescriptively. In future work we plan to use DSM as a tool to prescribe and study project design.

3.4 Responsibility and Project Team Interdependence

Finally, we wish to identify dependencies between actors who are responsible for producing information in given activities and actors who need the information produced by those activities. That is, we relate activities, information produced and consumed by those activities, and actors responsible for carrying out the activities. In the resulting interaction matrix $a_{ij} = 2$ means that an actor j is responsible for producing information in activity i, whereas $a_{ij} = 1$ means that actor j needs the information produced in activity i. The matrix forms a house of interdependence that illustrates the required information to exchange during project execution.

If we relate the production and consumption of information to Thompson's [28] typology of pooled, sequential, or reciprocal interdependence, we can represent the type of interdependence between actors in a triangular interrelation matrix at the roof of the house [19]. Actors who are responsible for activities that have no information dependence have *pooled interdependence*. Actors who are responsible for activities where one activity needs information from a previous activity have *sequential interdependence*. Actors who are responsible for activities where both activities need information from the other have *reciprocal interdependence*. Figure 6 shows the house of interdependence for the HSD project.

Figure 6 shows that the pump supplier needs information from tank pipe dimensioning, which is not yet carried out at the time of pump dimensioning. Thus, the pump supplier is reciprocally interdependent with the tank supplier during the execution of the pump dimensioning activity. The tank supplier also needs information from pump dimensioning. However, pump dimensioning precedes tank dimensioning, and thus the tank supplier is sequentially interdependent with the pump supplier. The pump supplier does not need specific information about fluid selection (in this simplified example), and the fluid supplier does not need information from pump dimensioning (the two zero elements in the matrix). Thus, the pump and fluid suppliers only have pooled interdependence.

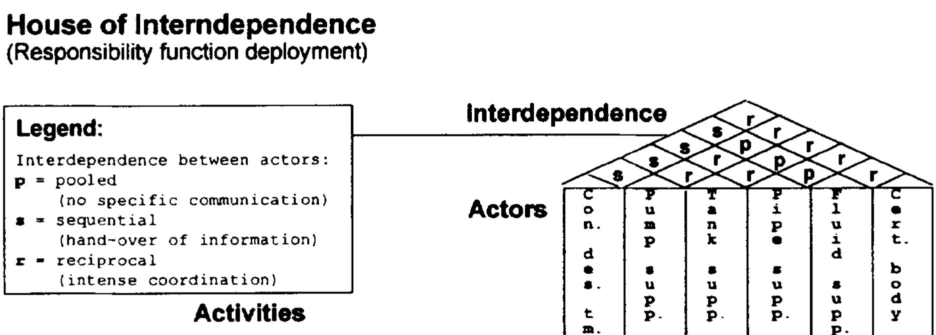
We can also relate the type of interdependence to the frequency of communication required between various pairs of actors during execution. For actors who have pooled interdependence, no specific communication is necessary while carrying out their activities. For actors who have sequential interdependence, the actor responsible for the latter activity needs to communicate with the actor responsible for the former activity. For actors who are responsible for activities in which both activities need information from the other (reciprocal interdependence), intense communication is required while carrying out their activities.

3.5 Overview and Limitations of the Load Modeling Methodology

Figure 7 summarizes our load modeling methodology. We see how structured representations of requirements and corresponding solutions, process activities, and project team actors are used as inputs to a set of matrix tools for deriving the relative distributions of complexity and uncertainty for the various activities and interdependence between team members.

We use these measures to quantify (a) the relative probabilities that solutions generated by given activities will contain errors, (b) the relative probabilities that solutions will fail to satisfy given requirements, (c) the relative measures of uncertainty and associated communication frequency for activities, and (d) the required participation in communication by project team members. In our view these measures are an important part of a consistent description of project enterprise, necessary for achieving correct simulation behavior of information processing and coordination.

Our methodology describes the detailed load distribution on individual actors, as opposed to a traditional description of coordination requirements as a point load through the center of gravity. This is an extension of traditional descriptions that only describes the aggregate complexity of the organization's technology and task ([12], p. 210), without identifying which part of the organization is subjected to load.



	Con. des. tm.	Pump supp.	Tank supp.	Pipe supp.	Fluid supp.	Cert. body
Capacity calculation	2	1	1	1	1	1
Pump dimensioning		2	1		0	1
Tank dimensioning		1	2			1
Pipe dimensioning		1	1	2		1
Fluid selection		0			2	1
Design approval	1	1	1	1	1	2

Figure 6. Responsibility and interdependence in the hydraulic system development project.

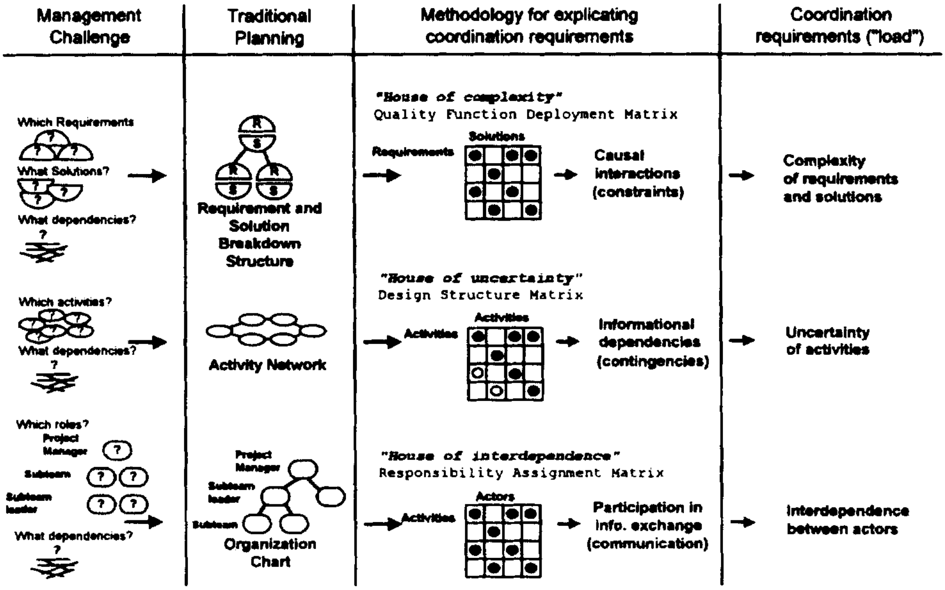


Figure 7. An overview of the methodology for modeling coordination load.

However, the decoupling of complexity due to causal dependencies in the project deliverables, and uncertainty due to informational dependencies, is an oversimplification. In most projects, causality and information need are not independent. Contingencies arising from missing information have the effect of making tasks seem more complex, and simultaneous constraints introduce uncertainty. Typically, missing information leads to rework, and errors lead to the need for additional information. We view this linearization as an initial approach to describe project dependencies. We feel that our simplification is valid if (and only if) we keep the limitations in mind when evaluating predictions made from the model.

4. DISCRETE EVENT SIMULATION OF INFORMATION PROCESSING AND COORDINATION

In this section, we outline how the coordination load described in Section 3 may be used in simulation of information processing and coordination handling, and how simulation gives estimates of project performance that may be used to predict probable effects of proposed changes to project design.

4.1 The VDT Discrete Event Simulator

The VDT [9] discrete event simulator is a result of an ongoing project at Stanford University, California [9, 29] with the aim of using simulation to investigate various aspects of project team organization. VDT is implemented as an object-oriented discrete event simulator in which each processor (actor) uses communication tools to carry out work generated by activities for which they are responsible and coordination with other actors in the project team.

Because VDT actors are modeled with limited rationality [14], they must engage in coordination—exception handling, rework, and communication—in addition to "working," or processing according to the project plan. This leads to a series of decision-making events [13], in which actors must allocate their attention to requests for communication and handling of failures discovered during verification. VDT uses a set of stochastic (random number) process elements to model uncertainty in human decision making. The simulation continues until all work and coordination items are processed, giving predictions for project performance in terms of the critical path duration, work volume (an approximation of project cost), and coordination performance (communication and error handling) [5].

The input to VDT consists of a description of the coordination load, the capability of the project team, and policies and preferences for handling coordination. The load is described in terms of activities' work volume, failure probability, and communication intensity, as described in Sections 3.2 and 3.3. Organizational capability (processing speed) is determined by the capacity of the team (manpower, structure, and tools) and the ability of team members (skill, craft, and experience) as described in Section 3.1. The team's handling of coordination during simulation is determined by the match of policies (what should be done in given situations) and preferences (what is actually done in those same situations). This match defines the information processing behavior of the various actors in terms of their decision making about attention allocation and participation [13]. Both project policies and preferences of actors are explicitly modeled [3] and may be altered between simulation runs to study the predicted effect on performance. Figure 8 gives an overview of input and output for the VDT.

Given the measures of the complexity of requirements and solutions described in Section 3.2, we must derive corresponding values for the complexity of the various activities. The higher the number of requirements a designer has to keep in mind when designing a given solution, the higher the chance he or she will make errors while carrying out the activity to produce the solution. A solution–decision matrix [2] relates solutions to activities and produces an *internal failure probability* for each activity, which is a measure of the chance of making mistakes (exceptions) while working. Such exceptions are typically discovered in self-checks or peer reviews. Similarly, the higher the number of solutions needed to satisfy a given requirement, the higher the chance that the requirement will not be satisfied, even if each individual solution may be according to specification. The requirements–access matrix [2] relates requirements to activities and gives an *external failure probability*, a measure of the chance of nonconformance when carrying out work to satisfy customer requirements. Such nonconformances are typically discovered at project milestones or during client reviews. The uncertainty of activities will determine the frequency with which responsible actors will generate communication requests. The interdependence between actors will determine to whom these communication requests are sent.

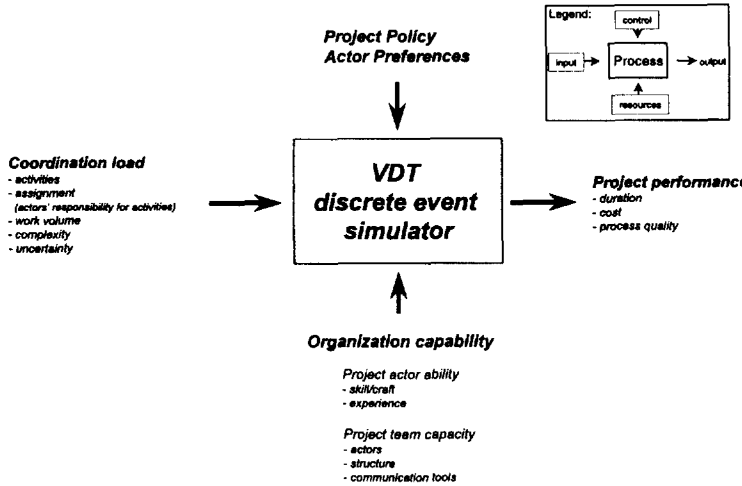


Figure 8. Information flow for the Virtual Design Team simulation.

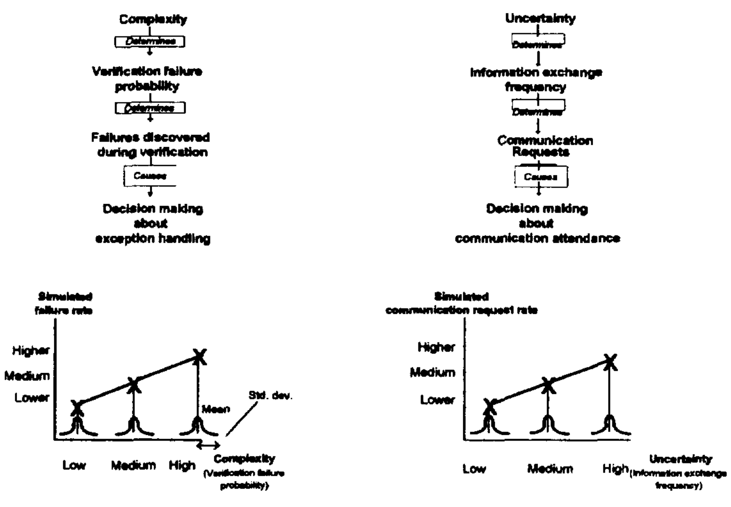


Figure 9. Using coordination load in the Virtual Design Team simulation.

4.2 Coordination Behavior in Simulation

For a specific set of inputs the VDT simulation will give the critical path duration, overall man hours (project cost), and process measures of the quality of coordination. Figure 9 illustrates how the coordination load for activities and actors is transformed to measures of verification failure probability and information exchange frequency (communication intensity) for each activity, and how these measures are used during simulation.

The top part of Figure 9 shows how complexity and uncertainty cause decision making about coordination. The bottom part of the figure shows how the outcome of this decision making determines project performance. The lower left graph shows how decoupling of requirements and solutions reduces the simulated fail-

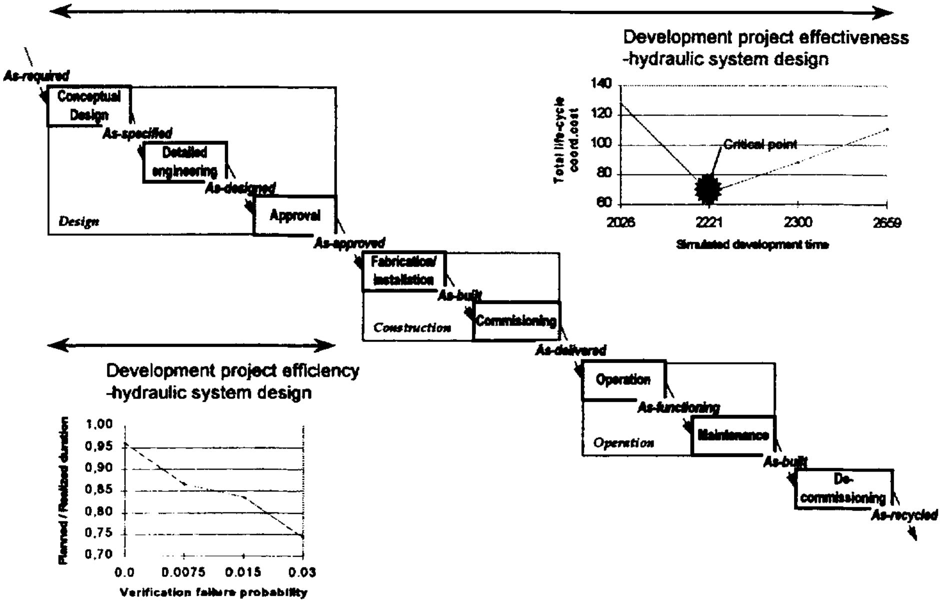


Figure 10. The cost of complexity and concurrency for the hydraulic system development project.

ure rate (i.e., higher complexity leads to more simulated failures). The lower right graph illustrates the number of requests for communication as a function of the uncertainty of activities. As shown, higher uncertainty leads to more simulated requests for communication.

Figure 10 shows simulation results for project coordination. The left graph shows the ratio of planned (specified) to actual (simulated) project duration, as a function of verification failure probability.

Our simulation results illustrate how decoupling of requirements and solutions (lowering the verification failure probability) is likely to lead to a project duration closer to the project plan. The nondimensional ratio of ideal to actual duration may be thought as a form of Mach number in analogy with fluid mechanics [30]. In the current model, projects can not be executed faster than planned, corresponding to the sonic barrier (Mach number 1; if the project could somehow be executed without coordination). As shown, coordination slows down execution, resulting in "subsonic projects." Allowing actors to assist each other on activities (collaboration, a coordination mechanism not yet included in the VDT) would allow "supersonic projects" (i.e., execution faster than plan).

The right graph in Figure 10 shows the *coordination cost* (the total number of man hours used in communication and rework during simulation) for the complete life cycle, as a function of (simulated) development time. The coordination cost is obtained by summing the time spent for rework and communication by all actors in all activities. *Development time* is the simulation time at completion of design approval. This was adjusted indirectly by altering the successor relations between activities in the project plan. That is, more concurrency between activities should give shorter development, whereas more sequential relations should give longer development.

Our simulation results indicate that life cycle cost does not necessarily decrease for any shortening of development time. Rather, there seems to be a critical development time with respect to minimizing coordination cost. Above the critical time, sequential activities will reduce uncertainty. However, because actors have more time for the same amount of work, they also have more time to attend to coordination. That is, lower coordination load gives more time to participate in all communication requests and leads to an increase in communication by all actors. Below the critical development time, further concurrency will result in a marked increase in coordination cost; that is, increased uncertainty leads to more communication requests. Consequently, the load on project team increases, and not all decisions can be dealt with satisfactorily. In particular, exception handling suffers, and rework is not carried out. The result of these unresolved problems in development is a higher failure probability in manufacturing and operation, and thus more rework in total.

We may use another analogy from fluid mechanics to view this ideal development time as critical Reynolds number [30]. Shortening development time below the critical value corresponds to a transition from laminar to turbulent flow. That is, the orderly manner in which the project team can process work and coordination items breaks down and processing becomes chaotic and unpredictable. This also indicates how coordination of the project plan is contingent on performance requirements. If rapid development is the only target in development, there may be a penalty due to higher coordination cost during manufacturing and operation.

4.3 The Effect of Project Policy on Performance

In this section, we briefly review simulation results (from [5]), for performance as a function of project policy in the HSD project, obtained from the mean of a series of simulation runs with different random seeds for stochastic decision-making processes. These results predict how a change in coordination policy (higher or lower value than the one used in the HSD project) is likely to affect duration, cost, and quality. The simulation predictions are compared with predictions from contingency theory [28] and predictions from the project manager (who both planned and managed execution of the project). (For a more detailed discussion of simulation results such as these, see [22, 23].)

Specifically, Figure 11 shows duration, cost, and verification quality—measured by the number of uncorrected exceptions—as a function of centralization. Our use of the term centralization [12] relates to the probability for how high up in the hierarchy decisions about exception handling must travel before reaching an actor with authority to make a decision. Carrying out rework involves time and cost but ignoring it lowers coordination quality. Thus, performance is influenced by actors' decisions about exception handling.

For the effect of centralization on duration the prediction from contingency theory is based on the assumption that project managers have a global view of dependencies between different parts of the project, and thus will tend to prefer rework, as they understand the potentially detrimental effect of ignoring failures in

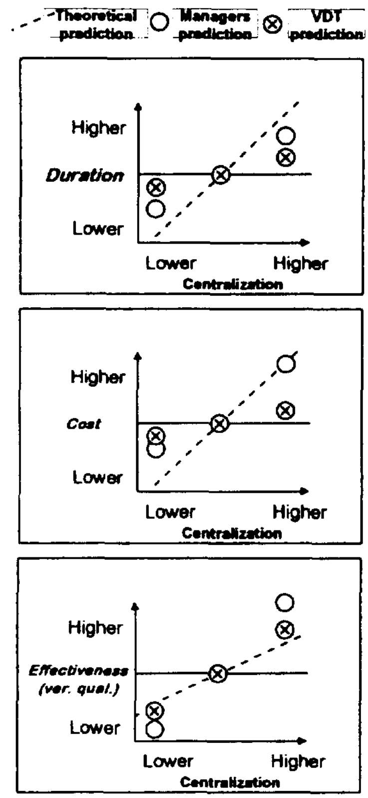


Figure 11. The effect of project policy on performance for the hydraulic system development project.

one activity on a number of dependent activities [6]. Project team members 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 result is that higher centralization (more decisions made by high-level managers) tends to give more waiting time for rework decisions as well as more rework. Both effects lead to longer duration. For cost the contingency prediction is the same as for duration, based on the same assumption that managers favor rework, which increases the total volume of work carried out. Higher quality means that more exceptions are reworked, and thus higher centralization gives higher verification quality. The simulation results are in good agreement with the predictions from the project manager—and consistent with contingency theory—and illustrate that there is no universally "best" centralization policy for the HSD project. The most suitable policy depends on the degree to which efficiency or quality has the highest focus, in which case one should choose a decentralized or centralized policy, respectively. That is, the choice of coordination policy for exception handling is contingent on project objectives.

Similarly, we have studied the effect of formalization on project performance [22, 23] as a function of the match between communication policy and the communication preferences (culture) of the project team. From all of our studies we observe how project performance is contingent on situational factors. This contingency illustrates

Thompson's [28] principle of contingency: There is no best way to coordinate projects, but for the objectives, products, processes, organization, and environment of a given project, not all ways to coordinate are equally good.

5. DISCUSSION OF RELATED WORK

In this section, we briefly discuss research relevant to project management in the fields of information systems modeling [31], coordination science [15], and enterprise integration [32].

5.1 Dynamic Modeling of Information Systems

Dur and Bots [31] reviewed a number of techniques for modeling of (aspects of) organizations and summarized the main requirements for modeling and evaluating the structure of organizations. Their review includes system behavior models (e.g., discrete event simulation), information system models (e.g., analysis and design models and timed systems models), information models and computer system models (e.g., data flow models, and task/agent/monitor models), formal organization models (e.g., organization structure charts, process models), and mathematical models (e.g., optimization models and equation simulation). In their view, the main requirements for models that give insight into dynamic interactions are (a) definition of processes as the sequencing of activities, (b) explicit representation of the time dimension, and (c) availability of functionality for investigation of different perspectives and levels of abstraction of dynamic behavior. They concluded that only discrete event simulation models, timed system models, and task/agent/monitor models qualify as truly informative dynamic models. Thus, our approach to modeling and simulation should qualify.

5.2 Coordination Science

Malone and Crowston [15] characterized coordination and coordination mechanisms. However, the work so far is theoretically oriented with the aim of understanding and extending theories about coordination. Thus, no operational model that can be utilized by project managers is yet available. As it has an object-oriented and easily extendable implementation, our framework and methodology could serve as an environment for theory testing, where the predicted effect of introducing different coordination mechanisms on overall project effectiveness and efficiency could be highlighted.

5.3 Enterprise Integration

Enterprise integration (EI) is a rapidly growing research area attempting to create operational enterprise models for handling coordination [32–34]. Its origin is computer integrated manufacturing. Growth is fueled by growing awareness that an integrated enterprise demands integration of hardware and software and humanware. A number of frameworks, architectures, and methodologies have been proposed for enterprise modeling within EI [35–37]. This is generally based on an assumption of perfect resources (no effect of real-world complexity and uncertainty, and thus no rework or communication). An important difference between the various EI architectures is that the levels of granularity of the different enterprise dimensions vary radically. The dimensions that are important in information integration and for representing process logic tend to be very detailed as opposed to organizational aspects that are generally represented in a coarse-grained and simple manner.

In summary, we feel that our approach is the most thorough attempt to cover all dimensions of enterprise necessary to assess the performance of a given project configuration. Our explicit representation of the limits between the various dimensions and our integration of representation and reasoning give a powerful tool for description and analysis of project enterprise.

6. SUMMARY ON RELEVANCE AND RESEARCH DIRECTIONS

Many of the problems associated with work in traditional enterprise result from poor coordination between different work processes taking place simultaneously and by a great number of dependencies between these processes. Work processes typically involve different functions, in different places, and often over an extended period of time. Also, most enterprises are not set up with sufficient information processing capacity to match the information processing load imposed by a global fast-paced business environment. This makes it difficult to draw clear lines between the different processes in an enterprise. Consequently, the value addition of work carried out is not clearly understood, and there is no clear overview of cost elements. The default result is involuntary introduction of organizational slack [27]; that is, performance falls below the maximum levels attainable by the enterprise.

Recent widely popularized efforts at business process reengineering [38–40] attempted to transform unfocused cross-departmental work processes, such as insurance claims processing or settlement of accounts payable, into reengineering projects, each with its own process owner (equal to project manager), suppliers, and customers, to clarify objectives and focus resources. This approach to reengineering is promising as projects traditionally have been characterized by clearly identified deliverables, project managers, cross-functional teams, activity plans, milestone events, limited life span, and resource constraints. A project focus on work processes thus supports delivery in compliance with customer requirements and execution according to project plan. This facilitates creation of unambiguous procedures for deploying and monitoring resources and explicit definition of responsibilities. Consequently, work processes may be easier to manage and improve and may give more predictable performance in an enterprise that has been project focused (project-based management) [42].

However, the challenge of a project focus should not be underestimated. Although projects have clear goals and well-defined procedures, they typically involve a high number of interactions between their subtasks and with tasks in other projects. Requirements for shortened time to market and reduced development cost demand drastically shorter duration of project-oriented work processes. As the duration is continually shortened, the concurrency between process activities is increased, creating exponentially greater coordination requirements. This creates dependencies between (a) actors responsible for activities in the same process, who were previously accustomed to sequential handover, but who must now coordinate the contents and progress of their work; and (b) actors in different processes (projects), who must coordinate their projects to ensure progress of all their tasks. These dependencies must be handled by extensive coordination between project team members. Managers almost always underestimate the resulting information processing load [41]. Thus, to plan projects efficiently and effectively managers must be able to identify and understand these dependencies. This requires a systematic methodology for describing and analyzing the coordination load within and between project teams. This need is not met in traditional project management tools [6] such as the critical path method [42], as they do not represent interdependence between parallel activities, or propagation of impacts due to nonconformances during project execution.

In Section 4.3, we varied policy and preference variables for a given coordination load and organizational capability to obtain performance estimates from a series of simulations. Given that our model is consistent and correct for other aspects of project enterprise, we may use it to study trade-offs between alternative ways to plan, man, and execute projects. This should allow us to address project managers' questions like, "Should I try to shorten the schedule by running these two activities in parallel?" "Will the team really work faster if I increase their manpower, or just spend more time communicating?" "Will the team members make too many mistakes if I increase their workload?" And "Will we exceed the budget if I mandate all communication?" We feel that our coordination load model and integration with discrete event simulation tools are significant steps toward giving meaningful answers to such questions. However, we also feel that further testing, application, and evaluation are required of both the modeling methodology and simulation tool. Integrated modeling and simulation allow systematic evaluation of simulation results. This will lead to better understanding of the probable effects of proposed changes. If work processes in traditional enterprise are increasingly organized and managed as projects, it becomes vital to gain better insight into the behavior of projects. We therefore need tools that support rapid and accurate prediction of performance for different project enterprise scenarios [4].

In future work we plan to extend our modeling and simulation of sets of projects to obtain a better understanding of the nature of coordination requirements between projects and the resulting effect on project performance of project enterprise. We also plan to model variations in the configuration of ensembles of processes and teams to investigate the competitiveness of different forms of project enterprise by simulation. We plan to do this in the spirit of Hannan and Freeman's [43] organizational ecology by determining those project enterprise configurations (organizational forms) that exhibit the highest rates of survival and prosperity according to different performance criteria. We expect that this should tell us something about how to organize work processes as projects and increase the usefulness of computer tools in designing efficient organizations [44]. We also plan to study the implementation of proposed changes in projects to understand how to use models in project execution. We hope that our framework and methodology will be used to build models as part of project planning, and that these models may be used during project execution to predict probable effects of proposed change [45]. Our vision is that managers may use models of project enterprise as a base for turning performance predictions into performance improvement.

REFERENCES

[1] T. R. Christiansen and J. Thomsen, "CAESAR—An architecture for enterprise modeling in the AEC industry," Det Norske Veritas Research, Høvik, Norway, Tech. Rep. 94–2019, Sept. 1994.

[2] L. C. Christensen and T. R. Christiansen, "QFD and process modeling for organizational analysis," Det Norske Veritas Research, Høvik, Norway, Tech. Rep. 94–2034, Dec. 1994.

[3] Y. Jin, R. E. Levitt, T. R. Christiansen, and J. Kunz, "The Virtual Design Team: Computer simulation framework for studying organizational aspects of concurrent design," *Simulation*, vol. 64, no. 3, pp. 160–174, 1995.

[4] R. E. Levitt, Y. Jin, G. Oralkan, J. Kunz, and T. R. Christiansen, "Computational enterprise models: Toward analysis tools for designing organizations," Stanford University, Palo Alto, CA, CIFE Working Paper 36, Feb. 1995.

[5] T. R. Christiansen, "Modeling efficiency and effectiveness of coordination in engineering design teams," Det Norske Veritas Research, Høvik, Norway, Tech. Rep. 93–2063, Oct. 1993.

[6] A. Laufer and R. L. Tucker, "Is construction planning really doing its job? A critical examination of focus, role and process," *Construction Management and Economics*, vol. 5, pp. 243–266.

[7] L. C. Christensen, T. R. Christiansen, and T. G. Syvertsen, "What is needed to model human enterprise—Elements of a unified enterprise engineering model," Det Norske Veritas Research, Høvik, Norway, Tech. Rep. 94–2020, June 1994.

[8] J. D. W. Moorecroft and J. D. Sterman, *Modeling for Learning Organizations*. Portland, OR: Productivity Press, 1994.

[9] R. E. Levitt, G. P. Cohen, J. C. Kunz, C. I. Nass, T. Christiansen, and Y. Jin, "The 'Virtual Design Team': Simulating how organization structure and information processing tools affect team performance," in *Computational Organization Theory*, K. M. Carley and M. J. Prietula, Eds. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc., 1993, pp. 1–18.

[10] P. Willems, "A functional network for product modeling" in *Proc. PLI–88–16, IBBC-TNO*, July 1988.

[11] J. J. Moder, C. R. Philips, and E. W. Davis, *Project Management With CPM, PERT and Precedence Diagramming*. New York: Van Nostrand Reinhold, 1983.

[12] W. R. Scott, *Organizations: Rational, Natural and Open Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[13] J. G. March, *Decisions and Organizations*. Oxford, England: Blackwell, 1988.

[14] H. A. Simon, *Administrative Behavior*. New York: Macmillan, 1958.

[15] T. Malone and K. Crowston, "Towards an interdisciplinary theory of coordination," MIT Sloan School, Boston, Working Paper 3294–91–MSA, April 1991.

[16] P. W. G. Morris, "Project organization structures for managing change," in *New Dimensions on Project Management*, Kelley, Ed. Lexington, MA: Heath, pp. 155–171.

[17] J. Bossert, *QFD: A Practitioner's Approach*. Milwaukee, WI: Quality Press, 1992.

[18] D. Daetz, "Planning for customer satisfaction with quality function deployment," in *Proc. Eight Int. Conf. of the ISQA*, 1990.

[19] J. R. Hauser and D. Clausing, "The house of quality," *Harvard Business Review*, pp. 63–73, May–June 1988.

[20] H. A. Simon, *The Sciences of the Artificial*. Cambridge, MA: MIT Press, 1969.

[21] S. M. Davis and P. R. Lawrence, *Matrix*. Reading, MA: Addison-Wesley, 1977.

[22] L. C. Christensen, T. R. Christiansen, Y. Jin, J. Kunz, and R. E. Levitt, "Enterprise modeling and simulation of AEC projects—A framework and an application to engineering design in the electric power industry," *Microcomputers in Engineering*, vol. 12, pp. 157–170, May 1997.

[23] L. C. Christensen, T. R. Christiansen, Y. Jin, J. Kunz, and R. E.Levitt, "Enterprise integration by modeling and simulation—A framework and an application to engineering design in the offshore

oil industry," *International Journal for Concurrent Engineering Research and Applications*, vol. 4, no. 3, pp. 247–259, 1996.

[24] S. D. Eppinger, "Model-based approaches to managing concurrent engineering," in *Proc. Int. Conf. on Engineering Design, ICED 91*, 1991.

[25] D. Gebala and S. D. Eppinger, "Methods for analyzing design procedures," in *Proc. Third Int. ASME Conf. on Design Theory and Methodology*, 1991.

[26] D. W. Steward, "The design structure system: A method for managing the design of complex systems," *ASME Journal of Engineering and Industry*, pp. 127–130, May 1978.

[27] J. Galbraith, *Designing Complex Organizations*. Reading, MA: Addison-Wesley, 1973.

[28] J. Thompson, *Organizations in Action: Social Science Bases in Administrative Theory*. New York: McGraw-Hill, 1967.

[29] G. P. Cohen, "The Virtual Design Team: An object oriented model of information sharing in project design teams," Civil Engineering Department, Stanford University, Palo Alto, CA, unpublished doctoral dissertation, July 1992.

[30] D. J. Tritton, *Physical Fluid Dynamics*. New York: Van Nostrand Reinhold, 1977.

[31] R. C. J. Dur and P. W. G. Bots, "Dynamic modeling of organizations using task/actor simulation," in *Dynamic Modeling of Information Systems*, Vol. 2, H. G. Sol and R. L. Crosslin, Eds. Amsterdam: Elsevier Science, 1992.

[32] P. Bernus and L. Nemes, Eds., *Modeling and Methodologies for Enterprise Integration*. London: Chapman & Hall, 1996.

[33] J. P. Petrie, Ed., *Enterprise Integration Modeling*. London: MIT Press, 1993.

[34] F. B. Vernadat, *Enterprise Modeling and Integration: Principles and Applications*. London: Chapman & Hall, 1996.

[35] T. J. Williams, P. Bernus, J. Brosvic, D. Chen, G. Doumeneigts, L. Nemes, J. L. Nevins, B. Vallespier, J. Vliestra, and D. Zoetekouw, "Architectures for integrating manufacturing activities and enterprises" in *Information Infrastructure Systems for Manufacturing*, H. Yoshikawa and J. Goossenaerts, Eds. Amsterdam: Elsevier Science, 1993.

[36] P. Bernus and L. Nemes, "Enterprise integration—Engineering tools for design enterprises," in *Modeling and Methodologies for Enterprise Integration*, P. Bernus and L. Nemes, Eds. London: Chapman & Hall, 1996, pp. 3–11.

[37] K. Kosanke, "Process oriented presentation of modeling methodologies," in *Modeling and Methodologies for Enterprise Integration*, P. Bernus and L. Nemes, Eds. London: Chapman & Hall, 1996, pp. 45–55.

[38] M. Hammer and J. Champy, *Reengineering the Corporation*. London: Brealey, 1993.

[39] M. Hammer and S. A. Stanton, *The Reengineering Revolution—A Handbook*. London: Harper Business, 1994.

[40] T. H. Davenport, *Process Innovation—Reengineering Work Through Information Technology*. Boston: Harvard Business School Press, 1993.

[41] D. Nadler and M. Tushman, *Strategic Organization Design: Concepts, Tools and Processes*. Glenview, IL: Scott, Foresman, 1988.

[42] J. R. Turner, *The Handbook of Project-Based Management*. Berkshire, England: McGraw-Hill, 1993.

[43] M. T. Hannan and J. Freeman, "The population ecology of organizations", *Am. Jnl. of Soc.*, vol. 82, pp. 929–964, 1977.

[44] R. M. Burton and B. Obel, *Designing Efficient Organizations: Modeling and Experimentation*. New York: North-Holland, 1984.

[45] Y. Jin, T. R. Christiansen, R. E. Levitt, and P. Teicholz, "Process modeling for design-build project management," in *Proc. Third ASCE Conf. on Computers in Civil Engineering*, 1996.