PRODUCTION AND OPERATIONS MANAGEMENT

POMS

Vol. 16, No. 2, March-April 2007, pp. 217–240
ISSN 1059-1478 | 07 | 1602 | 217$1.25

A Survey of Activity Network-Based Process Models for Managing Product Development Projects

Tyson R. Browning, • Ranga V. Ramasesh

M. J. Neeley School of Business, Texas Christian University, TCU Box 298530, Fort Worth, TX 76129, USA
M. J. Neeley School of Business, Texas Christian University, TCU Box 298530, Fort Worth, TX 76129, USA
t.browning@tcu.edu • r.ramasesh@tcu.edu

Given the crucial role of process modeling in product development (PD) project management research and practice, and the variety of models proposed in the literature, a survey of the PD process modeling literature is timely and valuable. In this work, we focus on the activity network-based process models that support PD project management and present a comprehensive survey of the literature published in the last decade. To organize our survey, we use a framework based on the purposes of PD process models: project visualization, project planning, project control, and project development. For each purpose, we provide an overview of the relevant models, highlight their key assumptions and findings, synthesize key insights, and illuminate avenues for further research. Although the survey reveals many insights and opportunities, five major areas for future study became apparent: activity interactions, global process improvements, process models as an organizing structure for knowledge management, modeling in cases of uncertainty and ambiguity, and determining the optimum amount of process prescription and structure for an innovative project.

Key words: product development; process model; literature survey; activity network; project management

Submissions and Acceptance: Received May 2005; revision received October 2005, March 2006; and June 2006; accepted July 2006 by Christoph Loch and Vish Krishnan.

1. Introduction

Product development (PD) comprises the myriad of multifunctional activities conducted by a firm between "defining a technological or market opportunity" and "starting production" of a unique product or service.[1] The goal of PD is to create a "recipe" (Reinertsen 1999) for producing a product or service. PD can be a major competitive lever for a firm. Companies such as Toyota have effectively reduced PD time to beat competitors to markets, contained PD costs by using fewer resources, and used PD activities as an opportunity to "design in" quality. The importance of PD has increased with the heightened pace of new product introductions and the mushrooming of product variety (Holman, Kaas, and Keeling 2003). The concurrent engineering approach to PD (e.g., Prasad 1996) increased the managerial challenges and the need for coordination and integrated decision support. The significance of PD and the pressures for "faster, better, cheaper" products has captured the attention of researchers in a variety of disciplines, including engineering, project management, operations management, organizational science, and marketing, which in turn has generated an extensive body of literature on PD in general and on project management in the PD area—especially the PD *process*. A process is "an organized group of related tasks that work together to create a result of value" (Hammer 2001) or a network of customer-supplier relationships and commitments that drive activities to produce results of value (Pall 1999). In recognition of the value of

[1] There may not necessarily be a clean break between development and production: some test and evaluation units may be produced prior to the "official" start of production, and some development work may continue beyond the start of production.

models to represent, understand, engineer, manage, and improve PD processes, a variety of models have been proposed in the literature. Most PD process models have used the activity network as a fundamental framework. We feel that a survey, synthesizing this extensive body of work, is both timely and valuable.

This paper presents a survey of the activity network-based process modeling literature pertaining to PD project management. Our goal is to provide an overview of the relevant papers published in the last decade and (i) highlight the models' main assumptions and findings, (ii) bring across key insights, and (iii) identify connections and gaps that suggest avenues for interesting research with high value to practitioners. The rest of this paper is organized as follows. Subheading 2 describes our scope and organizing framework and 3 our research methodology. Subheading 4 contains the literature survey, organized by modeling purposes; it summarizes insights and suggests areas for future research. Subheading 5 provides concluding remarks, including five major themes.

2. Scope and Organizing Framework

Given the vast literature on PD project management, developed in a variety of disciplines and with different methodological approaches, we limit the scope of our survey to facilitate a thorough presentation within space constraints. We define the scope as follows. First, in terms of the topic area, PD project management, we focus on the *process* rather than the *end product* (recipe) or the (human resources) *organization.* Thus, we do not include in our survey the literature dealing with the product architecture (e.g., Ramdas 2003; Ulrich 1995) or the organization design (e.g., Galbraith et al. 1993). However, we do mention several instances in which process interacts with product and organization. Second, we focus on *single projects*,[2] not on project portfolios. Thus, we do not survey the literature on which projects to undertake or how to evaluate one project relative to another. Where we include models that address both single- and multi-project characteristics, we focus on their contributions in the single-project case. Third, in terms of the methodological approach, we focus on a particular class of models, *activity-based models*, which view a project as a process, decomposed into a network of activities. Thus, we do not include in our survey most *systems dynamics models* (e.g., Sterman 2000), which view a project as stocks and flows of generic work to be done. Nor do we include *causal models* or *parametric models*, which might use techniques such as regression analysis.

[2] We use the term "project" broadly in this review to include the term "program" in cases where that term refers to a large project but not in cases where that term refers to a portfolio of projects.

Before discussing the framework we use to organize our survey, we provide a brief discussion of the differences between the general topic of project management and PD project management. The Project Management Institute defines a project as "a temporary endeavor undertaken to create a unique product, service, or result" (PMI 2004). Clearly, PD project management is a subset of the generic project management domain. However, PD projects tend to involve greater amounts of innovation, creativity, concurrency, and iteration than many other types of projects (Kline 1985). Ambiguities, uncertainties, and interdependencies among activities, their results, people, and their tools make PD processes complex and challenging to model. We identify three fundamental propositions that provide support and motivation for developing process models of PD projects: (1) Despite its novelties and ambiguities, the PD process has some repeatable structure (e.g., Austin et al. 2000a; Tatikonda and Rosenthal 2000). This proposition stems from the engineering design literature (e.g., Pahl and Beitz 1995), where design is something of an art but with many consistent patterns. That is, although a PD project seeks to do something unique, an individual or organization tends to follow a similar approach in each instance and learns and adapts (more or less) through successive instances. (2) Project management is facilitated by a structured approach, especially one supported by models of what work can and should be done when, and what information can and should be created when—i.e., process models. Although this standard assumption underlies most project management literature (e.g., Meredith and Mantel 2003; PMI 2004), it becomes especially important as the information flows become more complex, as in PD projects. (3) Processes are systems and can be engineered, facilitated by appropriate process models (e.g., Negele, Fricke, and Igenbergs 1997; Browning 2002). Indeed, PD processes are quite complex systems, yet they *can* be designed (Whitney 1990), and models become increasingly important to designers as complexity grows. Although these propositions are axiomatic and widely accepted, they could bear further scrutiny in research. Nevertheless, they provide a theoretical basis for the subject matter of our survey.

To organize our survey, we use a *purpose-based* framework. We ask, "What are the key issues managers face in designing and managing the PD process, for which process models provide support?" We identify four broad categories of purposes, as illustrated in Table 1.[3] We believe that these four categories provide a coherent and fairly complete framework to present an overview of the studies we survey. We

[3] These categories of purposes are derived from earlier compilations by Fricke et al. (1998) and Browning (2002).

Table 1 Four Categories of Purposes for PD Process Modeling[3]

1. PD project visualization
   a. Actions, interactions, and commitments
   b. Customized "views"
2. PD project planning
   a. Making commitments
   b. Choosing activities
   c. Structuring the process
   d. Estimating, optimizing, and improving key variables (time, cost, etc.)
   e. Allocating resources
3. PD project execution and control
   a. Monitoring commitments
   b. Assessing progress
   c. Re-directing
   d. Re-planning
4. PD project development
   a. Continuous improvement
   b. Organizational learning and knowledge management
   c. Training
   d. Compliance

recognize that other frameworks could be used to organize the literature. For example, Smith and Morrow (1999) organized their paper from the perspective of the *frameworks* (or methodologies) within which the models are built. Krishnan and Ulrich (2001) organized their paper around the *decisions* in PD. We do not claim that the purpose-based framework is superior to other classification schemes. Because our goal in this work is to bring across insights and identify future research opportunities rather than to develop a typology for classifying the existing literature, we find the purpose perspective particularly appealing. The purpose perspective is useful because it highlights the research areas in a broad sense (instead of pointing only to small changes to existing models). It helps identify purposes with a paucity of supporting models in the literature, thereby uncovering opportunities for further research. We feel that it is beneficial for uncovering previously unnoticed connections among seemingly disparate streams of work and for identifying knowledge gaps and research opportunities.

Our survey complements several previous papers. Brown and Eisenhardt (1995) focus on empirical work on the structures and processes by which individuals create products. They summarize the factors affecting PD project success but do not emphasize process modeling. Similarly, Gerwin and Barrowman (2002) and Shane and Ulrich (2003) provide empirical and general reviews of PD literature, but they do not emphasize process modeling. Elmagrabhy (1995) focuses on activity networks and computer-based software implementations of them in generic project management. Finger and Dixon (1989) and Kusiak (1999) review PD process models from an engineering design standpoint, whereas our survey focuses on the managerial purposes supported by process models. Smith and Morrow (1999) also concentrate on engineering models of the PD process, categorizing the papers by modeling framework and objectives such as sequencing and scheduling, decomposition, and design review timing. They also offer a set of criteria for evaluating models. Our survey is closest to the Smith and Morrow paper in the territory it covers, although we address a broader set of models, more recent papers, and the managerial purposes of modeling. Krishnan and Ulrich (2001) review the decisions made in PD, some of which (in the categories of Product Development Organization and Project Management) overlap with the purposes presented in our paper. Our survey focuses even more on the decisions in setting up a PD project, specifically the decisions supported by process models. Overall, our survey complements these earlier papers but goes beyond them in several aspects (such as covering a much broader set of literature and a wider set of managerial purposes for process modeling), thereby adding important insights and illuminating new research opportunities.

3. Research Methodology

We followed, much like Krishnan and Ulrich (2001), a loosely structured approach to comprehensively survey the vast and expansive literature relating to PD process modeling within the defined scope. We focused on works that self-identify with the term "PD." However, because many papers address similar issues without this term, using the term "project" instead, we also surveyed some non-PD-specific literature in areas such as project management, knowledge management, business process modeling, and systems engineering, where these otherwise fit our scope. First, we created a superset of papers[4] related to PD process modeling through several steps: (i) We searched the tables of contents of 18 major journals from 1994 to early 2005: *ASME Journal of Mechanical Design*, *Decision Sciences*, *Design Studies*, *European Journal of Operational Research*, *IEEE Transactions on Engineering Management*, *Interfaces*, *Journal of Engineering Design*, *Journal of Marketing Research*, *Journal of Operations Management*, *Management Science*, *Manufacturing & Services Operations Management*, *Marketing Science*, *Operations Research*, *Organization Science*, *Production and Operations Management*, *Project Management Journal*, *Research in Engineering Design*, and *Systems Engineering*. These journals span the engineering design, management science, marketing, operations management, and project management areas. (ii) We conducted a general search of the literature based on key words, looking also at the broader literature on software development, business processes, and knowledge management. (iii) We used

[4] For simplicity, we refer to all publications as "papers."

the reference lists from highly cited papers. (iv) We surveyed 44 members of the PD research community, seeking inputs on what they thought were the influential reports on PD process modeling, or the most influential PD process models. These steps gave us a master list of about 400 papers, from which we culled a working list of about 200 papers by filtering out ones that were (a) outside our scope, (b) not in archival publications, and (c) devoted to software tools and vendors.

4. Literature Survey

We organize our survey around the categories of PD process modeling purposes shown in Table 1. In each category, we identify the key purposes, in most cases by listing them in tables with references to associated papers. We distinguish purposes that we feel are in special need of research by using stars in the tables. These tables serve as stand-alone guides for the reader. Considering each purpose (or, in some cases, sub-purposes), we first present a brief discussion of the literature, then bring across the key insights, and finally identify avenues for further research pertaining to that purpose.

4.1. PD Process Visualization Purposes

Documenting what, how, and when work should be done—in a highly visual medium—surfaces latent assumptions and sparks innovation (Dougherty 2001). For example, a large process flow map in a conference room can be the focal point for group discussion and the vehicle for moving towards shared mental models. Process models can provide "situation visibility" (Steward 2000): they empower the workforce to achieve focused, committed, and accountable collaboration (Nonaka 1994), which is becoming increasingly important in large, complex organizations and supplier networks.

In supporting the visualization purposes, a process model may be "*viewed*" in several different ways. Whereas a process model includes the *attributes* of and the underlying *assumptions* about the process which are deemed sufficient to describe it, a *view* is an arrangement of symbols, a table, or other depiction chosen to display a selected subset of those attributes and assumptions. For example, the ubiquitous process flowchart provides a *view* of the activities in a process and their precedence relationships via a set of symbols—i.e., boxes and arrows. This view tends to emphasize the activities—they are often labeled or named—and may include notation about their durations, start and finish times, etc. PD processes are complex, and it is impossible to describe their behavior fully from a single standpoint, using a single view. Because the inclusion of too much information crowds the view, everything known about the process is deliberately not included in a view. Thus, a process model contains a superset of information about a process, while a view contains a subset of that information in a chart, table, or other depiction. By reducing complexity and focusing on key leverage points, views (also called "representations") can be a significant driver of innovation in system design (Alexander 1964; Simon 1981; Zachman 1987) and PD decisions (Krishnan and Ulrich 2001).

Within this category of purposes, we identify the two purposes shown in Table 2. From an organization design and performance perspective, several papers model and simulate the micro-level actions and interactions of people and teams spawned by a process network (Christian and Seering 1995; Levitt et al. 1999; Moser et al. 1998). In these models, endowing the process participants with aligned goals and understanding increases organizational effectiveness. While these models do not emphasize a particular view, they exemplify the advantages of a workforce performing with the benefits of a common one. Other reports discuss the function of process model visualization and specific views for project planning. Haque and Pawar (2001) recommend a view emphasizing the assignment of people to the "roles to be filled" in each activity. (Roles are often represented on flowcharts through the use of "swim lanes," which orient the activities performed by a specific individual or organization in a row demarcated by dashed lines.) Malone et al. (1999) organize processes at various levels of abstraction (e.g., "develop product" can be consid-

Table 2 PD Process Visualization Purposes

Purpose	Selected references		
★ How can process participants visualize their actions, interactions, and commitments?	(Christian and Serring 1995)	(Moser et al. 1998)	(Levitt et al. 1999)
How can the workforce visualize and understand the project's planned process?	(Haque and Pawar 2001) (Haque 2003)	(Malone et al. 1999)	(Chung et al. 2002)
★ What view or views of a process model highlight the most pertinent information for various "user segments"?	(Kusiak and Larson 1995) (Tomberg et al. 2002) (Qian and Shensheng 2002)	(Basu et al. 1997) (Yu et al. 2000) (Sousa et al. 2002)	(Bond 1999) (DoD 2001) (Presley et al. 2001)

ered generically or in terms of its modes or specializations, such as "develop derivative product" or "develop new product") to facilitate process planners' navigation of a repository of process data. Chung, Kwon, and Pentland (2002) emphasize the importance of visualizing a project's potential *process space*—the range of process scenarios that could unfold.

Because different users of process models come with specific purposes and needs, the issue arises as to which subset of the information in a "rich" process model (i.e., one characterized by a large number of descriptive attributes) is relevant to a particular set of users, and then how to filter the view of that facet. Basu, Blanning, and Shtub (1997) point out that the users should be able to customize their views. They define rules for composing and projecting models so that a simple model can be abstracted from a complex one by hiding certain variables and relationships. Basu, Blanning, and Shtub (1997), Bond (1999), and Presley et al. (2001) all note the importance of keeping model information in *one place with multiple views* instead of spread across *disparate models*, particularly to facilitate process integration. A process model can provide the foundation and integrative structure for several other models and views, such as activity-based cost accounting (Tornberg, Jämsen, and Paranko 2002) and the assignment of resources and personnel to activities (Qian and Shensheng 2002).

In most cases, views are a moderator of a process model's effectiveness in meeting another purpose. That is, while some may build a process model for the primary purpose of visualization, visualization is an important secondary aspect of a process model built for any other purpose. When another purpose is paramount, better visualization of the model improves the means toward that end.

This part of our survey illuminates several research opportunities. First, although many researchers, consultants, and practitioners have observed that a significant portion of the value of process modeling accrues from merely building a model and discussing its accuracy and other characteristics, it is important to develop approaches to measure the benefits of process visualization as a way to help justify investments in process models. How can we measure the value of increased organizational alignment and the contribution towards it provided by process models? Second, future research could explore which views are most useful to particular constituencies and to support each of the other purposes discussed below. What valuable, new views can be created? On one hand, each view should be structured and designed independently, to suit its users' needs. On the other hand, it is desirable to synthesize all the views into a single architecture (Basu, Blanning, Shtub 1997; Bond 1999; Yu et al. 2000; Sousa, Aken, Groesbeck 2002), like the U.S. Department of Defense Architecture Framework (DoD 2001) does for product architectures, so that each view draws from the same foundational process model. The techniques of information hiding (e.g., Petitcolas, Anderson, and Kuhn 1999) may enable a variety of users to draw from a common database while addressing security concerns—e.g., when supplier or partner companies wish to integrate their processes for a limited time without giving away all of their process knowledge. Finally, how can a view be verified as suitable for one purpose yet designated as unverified to support other purposes and decisions? Such tagging would help practitioners use the best view (i.e., the correct subset of process information) to support a particular decision and avoid basing decisions on the wrong views.

4.2. PD Project Planning Purposes

Process models support many aspects of PD project planning. Given a set of goals or objectives for PD, planners must determine the appropriate way to achieve them by answering the questions in Table 3.

4.2.1. Making Commitments. Commitments define "who owes what to whom and at what time." The question "What commitments should be made by and within a project?" is crucial to answer, especially when planning complex and/or ambiguous projects that entail many *ad hoc* activities. Pall (1999) provides a framework for modeling a project's process as a *network of commitments*. Each activity in and connected to the project is viewed as both a supplier and a customer—i.e., as an entity that is owed inputs by others, uses these inputs to do work, and in turn owes one or more outputs to others. An activity network thus implies the appointment of a set of agreements and commitments about inputs and outputs. This approach pertains to both internal and external suppliers. Spear and Bowen (1999) show the power of making such customer-supplier connections "direct" and "unambiguous" at Toyota, even in a project context. Having a *de facto* (or standard) network of commitments enables a project planner to know his or her capability to make further, downstream commitments. Although commitment networks are relatively easy to manage for repetitive processes, future research could explore three areas: (1) How to form and manage dynamic commitment networks in highly ambiguous PD projects? (2) What is the value of maintaining a *de facto* network of commitments as a generic template for a new PD project? (3) How general or specialized should a *de facto* network of commitments be in order to apply to a firm's full spectrum of PD projects?

4.2.2. Choosing PD Activities. PD project planners must determine which activities to do. As a starting point, the product design and systems engineering

Table 3 PD Project Planning Purposes

Purpose (Decision Supported)	Selected references and subject areas		
★ What commitments should we make?	(Pall 1999)		
★ What activities should be done? — What are the standard activities?	(Song and Montoya-Weiss 1998) (Fairlie-Clarke and Muller 2003) (Malone et al. 1999)	Canonical models (Radice et al. 1985) (Austin et al. 2000a)	(Sim and Duffy 2003) (Eggersmann et al. 2003) (SEI 2002)
★ What are the main areas of ambiguity, uncertainty, and risk?	Risk management literature (MacCormack and Verganti 2003)	(De Meyer et al. 2002) (Browning et al. 2002)	(Pich et al. 2002)
— How much and what kind of testing should be done?	(Thomke 1998) (Dahan and Mendelson 2001)	(Thomke and Bell 2001) (Loch et al. 2001)	(Engel and Barad 2003)
— What critical decisions will need to be made?	(Krishnan and Ulrich 2001)	(Buede and Powell 2001)	
★ How and to what level should we decompose a process?	(Michelena and Papalambros 1995) (Chen et al. 2005) (Kusiak and Larson 1995)	(Alexander 1964) (Simon 1981) (von Hippel 1990)	(Altus et al. 1996) (Rogers 1999) (Bras and Mistree 1991)
★ How can we account for process ambiguity (unforeseen uncertainty and chaos)?	(Clarkson and Hamilton 2000) (Baldwin and Clark 2000) (O'Donovan et al. 2003) (Lévárdy and Browning 2005)	(Pich et al. 2002) (Sommer and Loch 2004) (Danesh 2001) (MacCormack et al. 2001)	(Chung et al. 2002) (Chung et al. 2003) (Girard et al. 2002) (Loch et al. 2006)
★ How should we structure the process? — What macro process structure should we employ?	(Unger and Eppinger 2002) (Boehm 2000)	(NASA 1995) (Camel and Becker 1995)	(Cooper 2001) (Oppenheim 2004)
— When should activities be done?	(Smith and Eppinger 1997a; 1997b; 1998) (Browning and Eppinger 2002)	(Ha and Porteus 1995) (Krishnan et al. 1997b) (Bhattacharya et al. 1998)	(Ahmadi and Wang 1999) (Ahmadi et al. 2001) (Chou 2002)
★ Which activities should be overlapped and by how much?	(AitSahlia et al. 1995) (Calantone and Di Benedetto 2000) (Terwiesch and Loch 1999b) (Roemer and Ahmadi 2004; 2000)	(Krishnan et al. 1997a) (Loch and Terwiesch 1998) (Ford and Sterman 1998) (Loch et al. 2001)	(Xiao and Si 2003) (Joglekar et al. 2001) (Hu et al. 2003) (Chakravarty 2001)
★ How should we structure the information flow?	(Eppinger et al. 1994) (Lockledge and Salustri 1999) (Loch and Teriwiesch 2005)	(Yassine et al. 1999) (Crowston 1997) (Mihm et al. 2003)	(Eppinger 2001) (Terwiesch et al. 2002) (von Hippel 1990)
★ How should we integrate processes?	(Presley et al. 2001) (Casatie and Discenza 2001) (Hong and Hong 2001)	(Morelli et al. 1995) (Pall 1999) (Crowston 1997)	(Fricke et al. 2000) (Browning 2002)
★ How should we estimate, optimize, and/or improve key project variables? — Lead time (duration) and/or cost?	(Neumann 1990) (Belhe and Kusiak 1996a) (Bhuiyan et al. 2004) (Browning and Eppinger 2002) (Abdelsalam and Bao 2006)	(Elmaghraby 1995) (Goldratt 1997) (Ahmadi et al. 2001) (Cho and Eppinger 2005)	(Eppinger et al. 1997) (Carrascosa et al. 1998) (Kara et al. 1999) (Jun et al. 2005)
— Resource requirements and constraints?	(Anderson and Joglekar 2005) (Belhe and Kusiak 1997; 1996b) (Kumar and Ganesh 1998)	(Herroelen 2005) (Taylor and Moore 1980) (Brucker et al. 1999)	(Adler et al. 1995) (Luh et al. 1999) (Yan et al. 2002)
★ Quality of results?	(Paquin et al. 2000)	(Browning et al. 2002)	
★ Uncertainty, risk and opportunity?	(Hillson 2003) (Ben-Haim, 2006) (Loch et al. 2006)	(Pich et al. 2002) (Luh et al. 1999)	(Browning et al. 2002) (Browning and Eppinger 2002)
★ Robustness, flexibility, and adaptability (and their value)?	(Huchzemeier and Loch 2001) (Ben-Haim, 2006) (Santiago and Vakili 2005)	(Pall 1999) (Pentland 2003)	(Haeckel 1999) (Ford and Sobek 2005)
★ More than one of the above variables?	(Roemer and Ahmadi 2004; 2000) (Browning and Eppinger 2002)	(Luh et al. 1999) (Cohen et al. 1996)	(Elmaghraby 1995) (Graves 1989)
★ Where and when should we allocate resources?	(Joglekar and Ford 2005) (Thomke and Fujimoto 2000) (Repenning 2001)	(Ahmadi and Wang 1999) (Cohen et al. 1996) (Bassett et al. 2004)	(Lee et al. 2004) (Khanna and Iansiti 1997)

literature is replete with generic, *standard* (or canonical) design processes (e.g., Pahl, Beitz, and Wallace 1984; Suh 1990; Whitney 1990; Forsberg, Mooz, and Cotterman 2000; Ulrich and Eppinger 2004). The project management literature traditionally proposes the use of a *work breakdown structure* (WBS) based on the desired outputs of the project—i.e., determine the high-level activities required to satisfy a project's intents and then decompose these activities (e.g., Meredith and Mantel 2003; PMI 2004). A firm may prescribe *standard process models* (i.e., a template of typical activities and relationships—Radice et al. 1985) or process handbooks (Malone et al. 1999) at various levels of detail, and it may use these as a basis for project planning (e.g., Austin et al. 2000a). In certain contexts, such as government contracting, some activities (such as safety tests and progress reviews) may be mandated (e.g., DoD 1998). Finally, process standards (not to be confused with standard processes) such as the ISO 9000 series[5] and SEI's CMMI^{SM} (SEI 2002)[6] also provide guidance on activities to include to ensure an effective process. Any and all of these templates and approaches may feed a PD project's initial list of activities.

In addition, PD projects should include activities that address critical uncertainties and decisions. Viewing PD as a process of generating information that reduces uncertainty, several authors (Browning et al. 2002; Pich et al. 2002; MacCormack and Verganti 2003) advocate that an activity's selection occur on the basis of its contribution to uncertainty or risk reduction. For example, an activity that creates information that will confirm a product design's conformance to its requirements adds value (Browning 2003). The risk management literature (e.g., Conrow 2000) includes numerous checklists and guides for including activities to reduce foreseen uncertainty. Specific models have explored the number and types of test (verification) activities to include: Thomke (1998) and Thomke and Bell (2001) determine when to select a few high-fidelity tests versus a larger number of low-fidelity tests. Activities that make critical decisions must also be planned. In fact, many view PD as a decision-based process (e.g., Muster and Mistree 1988; Bras and Mistree 1991; Hazelrigg 1998; Ullman 2001). Krishnan and Ulrich (2001) and Buede and Powell (2001) catalog these decisions, but they do not define the specific activities that should be done to make and support them.

The determination of specific activities to be included in an activity network also depends on the method and extent of process decomposition. The decomposition paradigm is common in system design and modeling (e.g., Alexander 1964; Simon 1981). A key idea here is how the decomposition of the "problem system" (what to do) strongly affects the decomposition of the "process system" (how to do it). Hence, the engineering design literature on the decomposition of design problems into sub-problems (e.g., Pahl et al. 1984; Bras and Mistree 1991; Michelena and Papalambros 1995; Altus, Kroo, and Gage 1996; Rogers 1999; Chen, Ding, and Li 2005;) strongly influences the PD process modeling literature. Modularity is advocated in both contexts. Von Hippel (1990) notes how the resulting decomposition affects the ease of subsequent process integration (discussed in Subheading 4.2.3.). Of particular importance to PD, he recognizes that modular process decomposition enables scheduling activities in parallel without causing additional iteration and rework.

Furthermore, when choosing activities, PD project planners must deal with ambiguity and complexity, which inhibit the ability to pre-specify all of the actions required to deliver a satisfactory outcome. Several of the most recent PD process modeling papers recognize this situation. Pich et al. (2002) characterize a project's process in terms of its information structure (knowledge about the *state of the project* and the world) and *policies* (contingency plans), which can be compared to dynamically re-determine the appropriate activities. While providing conceptual insights, this model addresses only generic, undifferentiated activities. At the level of individual designers who must collaborate, Danesh (2001) models the PD process as an ongoing set of decisions about which activities to do, coordinated to convergence through common policies. (Baldwin and Clark 2000 also discuss PD policies, calling them "design rules.") Clarkson and Hamilton's (2000) *signposting* method dynamically selects activities during PD process execution based on the confidence level of a potential activity's inputs. Which activities to do and when is governed by policies based on the state of the information inputs and the capabilities of the activities. A pre-evaluation selects the appropriate version of an activity and a post-evaluation step determines if iteration is needed. Similarly, Chung, Kwon, and Pentland (2002) advocate a "grammatical approach"[7] to process specification and

[5] ISO is an acronym for the French version of International Organization for Standardization. The ISO 9000 series of standards pertains to a company's quality management systems. The theory is that documenting what and how work is done improves quality (Corbett, Montes-Sancho, and Kirsch 2005).

[6] Software Engineering Institute's Capability Maturity Model®—Integrated and CMMI^{SM} are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

[7] "As the grammar for a language describes all possible sentences, a process grammar describes all possible arrangements of tasks in a design process. Rather than focusing on a particular process, a grammatical approach draws attention to the set of alternatives." (Chung, Kwon, and Pentland 2002) For further background on process grammars, see Pentland (1995).

define a *process space* of all possible activities and their arrangements. Finally, Lévárdy and Browning (2005) model an adaptive process that dynamically selects the best version or mode of an activity based on the state of the project and the expected value added by the activity.

Some key insights on choosing PD activities may be summarized as follows: (i) Standard process templates, canonical models, and process standards might provide a useful baseline. (ii) Activities can be chosen on the basis of their ability to create information that will reduce key uncertainties and risks and enable vital decisions. (iii) Lower-level activities may be chosen according to the decomposition of the product design subproblems to be solved. (iv) When activities interact intensely, their assignment to modular groups can facilitate their subsequent integration and management. (v) The dynamic state of an ongoing project should be monitored and used to redetermine the most appropriate activities or versions thereof. (vi) Policies or rules can be put in place to guide this ongoing selection of activities.

We identify four broad avenues for future research. First, how much should a firm invest in maintaining standard process models and ensuring they comply with external standards? Although adherence to good standard processes will ensure that all projects in a firm use best practices, too much standardization can be stifling to and will be ignored by the workforce. What are the best ways to tailor and scale standard process models to suit the unique requirements of a particular PD project? Second, because a project's level of uncertainty, ambiguity, and complexity should dictate its management style and tool set, what special kinds of activities should be chosen in contexts of high ambiguity and risk? Is it possible to develop a systematic approach to choosing the activities whose outcomes would reduce the most critical unknowns? If a project should define policies for dynamic activity selection, how can managers know when these policies themselves require adjustment? Third, PD project planners often struggle with the *level* to which a PD process model should be decomposed. That is, at what level of granularity should activities be specified? While a general guideline is to decompose a process to the level necessary to sufficiently understand, plan, monitor, and control it, specific stopping points are unclear. For example, planners may specify the activities they know the most about in great detail while glossing over the areas they know the least about (e.g., leaving them as the "long bars" on a Gantt chart). Just the opposite approach would seem to be appropriate, but is it tenable? Can the additional planning effort be justified? Fourth, processes can be decomposed in several possible ways—e.g., in temporal phases, by product component, by organizational unit, etc. In practice, the decomposition is often forced to follow the decomposition of the product or organizational architectures (either of the project at hand or of similar projects). What is the preferred approach to decomposing a PD process? Decompositions based on product, problem, organization, etc. could be compared and contrasted for their advantages, disadvantages, and insights.

4.2.3. Structuring the PD Process. In conjunction with determining which activities to undertake, PD project planners must decide how to structure a process. That is, how should they arrange the chosen activities in a network? How do the activities depend on each other? Which activities should they plan to do only with finalized information, and which might initially proceed with preliminary information or based entirely on assumptions? Where should they position the interim reviews of project progress? How should they manage iterations? Depending on the size of the project, planners may have to make these decisions at many levels.

At a macro level, a number of PD process structures have been proposed. For example, the basic stage-gate process (e.g., Cooper 2001; NASA 1995) illustrated in Figure 1 conceives of PD as a (potentially overlapping) sequence of activities and reviews. When the results of a stage or phase are reviewed against that stage's exit criteria, a successful outcome allows continuation to the next stage, while a failure requires further iteration within a stage. For another example, the Spiral Development process (e.g., Boehm 2000) illustrated in Figure 2 conceives of PD as a set of planned iterations through all of the major stages in Figure 1. Unger and Eppinger (2002) assess these and other PD process

Figure 1 **A basic stage-gate PD process (adapted from Cooper 2001).**

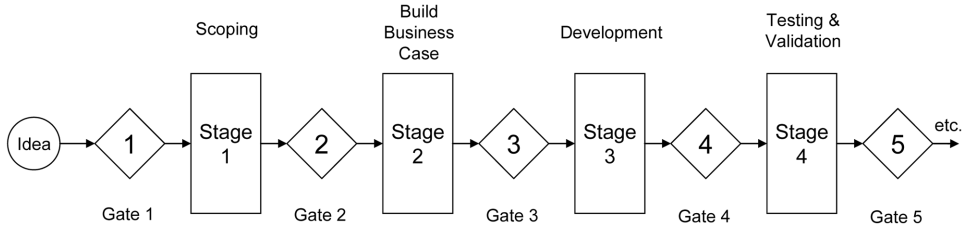
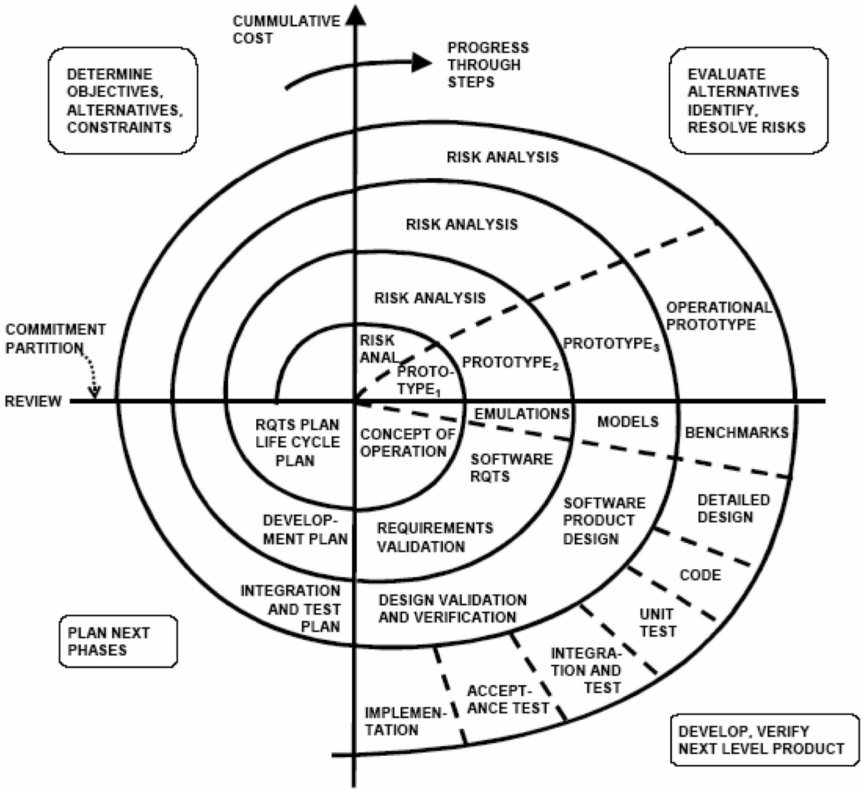


Figure 2 Boehm's depiction of a spiral development process (Boehm 2000).



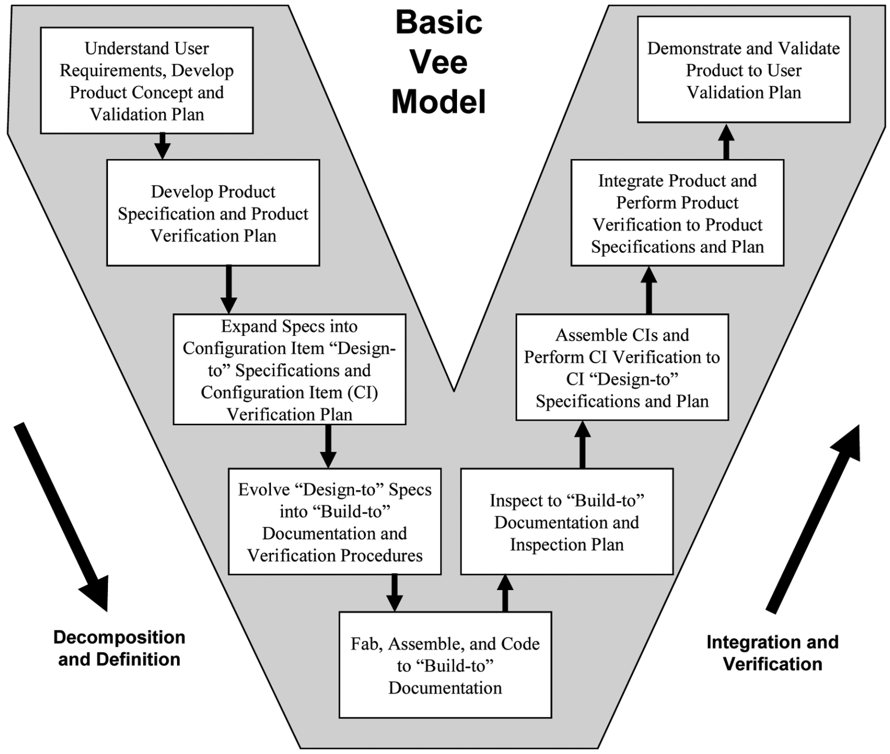
structures using the number and location of iterations and reviews as the distinguishing characteristics. They compare each process structure in terms of its approach to managing risk and propose matching a PD project's risk profile to a process structure with an appropriate iteration/review combination. Another macro structure popular in industry is the systems engineering "V" model (e.g., Forsberg, Mooz, and Cotterman 2000; NASA 1995) illustrated in Figure 3. In this model design problems and requirements are decomposed in the early stages of a process and the developed components are integrated in the later stages. Under conditions of low uncertainty, Oppenheim (2004) proposes a "lean" PD process in which activities are broken into one-week increments. This task sizing allows regularly positioned reviews (each Friday) and a built-in buffer (the weekend) for rework. Overall, the number, size, and location of project *reviews* and *iterations* would seem to determine the best macro structure.

Several models help guide *when* to perform activities such as reviews, decisions, and iterations. Ha and Porteus (1995) model the optimal positioning of design reviews as a tradeoff between the preparation (setup) time for each review and its benefit in catching errors early. They show that merely including the reviews (without optimizing their location) realizes 90% of their benefits. Ahmadi and Wang (1999) build on this work by using a Markov model to track confidence in the evolving conformance level of a product design. They define "stage confidence" as the conformance level at the completion of a given process stage, based on the effort spent on that stage and the results of prior stages. Their policy is for activities in a stage to iterate until they achieve a specified level of stage confidence. PD project planners can also schedule activities that will make critical decisions. For example, by modeling the tradeoff between getting locked into incorrect requirements and not having time to optimize a product's unit cost, Bhattacharya, Krishnan, and Mahajan (1998) determine the optimal design review in which the decision about when to freeze design requirements should be made. By modeling the loss of design freedom over the course of a PD process, Krishnan, Eppinger, and Whitney (1997b) stipulate the optimal activity sequence as the one in which the decisions impose the least constraint on downstream activities.

Turning to the timing of iterations, several models based on the *design structure matrix* (DSM)[8] framework provide insights. As illustrated in Figure 4, a DSM is a square matrix representing the activities in a process (the shaded cells along the diagonal) and their inter-

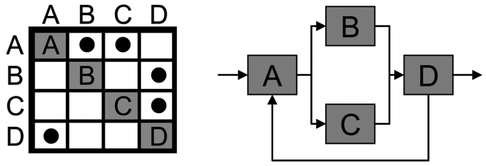
[8] For further information on the DSM, see (Browning 2001) and (Eppinger 2001).

Figure 3 A basic systems engineering "Vee" process (adapted from Forsberg *et al.* 2000).



actions (the off-diagonal marks). One reads down an activity's column to see its inputs and across its row to see its outputs (although the opposite convention is also used). For example, the DSM in Figure 4 shows activity A providing outputs to B and C and receiving an input from D. Thus, the super-diagonal region of this DSM shows the traditional precedence relationships among activities, while its sub-diagonal region highlights potential feedback loops or cycles, which imply both the need for upstream activities to make assumptions about unavailable inputs and the potential for iteration should those assumptions prove inadequate. Several DSM-based models provide guidance on sequencing activities to minimize iteration (e.g., Smith and Eppinger 1997b; Ahmadi, Roemer, and Wang 2001). Browning and Eppinger (2002) account for activity overlapping and show that minimal iteration is not optimal when accounting for learning curves and the possibility of moving long activities off of the critical path. Finally, rather than prespecifying the order of any activities, Chou (2002) proposes that the activities begin when their *entry criteria* are satisfied and resources are available, and that planners thereby structure the activities incrementally.

Figure 4 Equivalent DSM and flowchart representations of a process with four activities.



Especially with the preeminence of concurrent engineering approaches in PD, process structure and activity timing greatly depend on the extent of activity overlapping. AitSahlia, Johnson, and Will (1995) use PD process models to show that complete overlapping is suboptimal and note that *uncertainty* may erode the advantages of concurrency. Krishnan, Eppinger, and Whitney (1997a) optimize the overlap in an activity dyad based on the upstream activity's evolution rate and the downstream activity's sensitivity and show that different types of overlapping drive time, cost, and quality tradeoffs. Extending this work, Loch and Terwiesch (1998) account for communication and uncertainty effects in determining the optimal degree of concurrency and show that improved communication allows increased overlapping (by mitigating rework). A further extension by Joglekar et al. (2001) accounts for resource constraints and rework generation and shows that factors exogenous to the activity dyad are important to consider when determining the optimal amount of overlap. Roemer and Ahmadi (2004) explore overlapping and crashing in a two-activity model. Moving slightly beyond the activity dyad, Roemer, Ahamdi, and Wang (2000) model the overlapping of adjacent phases of the PD process and demonstrate the time-cost tradeoffs driven by the amount of overlapping. Thus, the presence of uncertainty, the effectiveness of communication, the available resources, and other project characteristics, such as relative time-cost preferences, figure into the overlapping decision.

The *information processing* perspective (e.g., Burns and Stalker 1961; Galbraith 1977; Tushman and Nadler 1978; Pahl et al. 1984; March and Simon 1993) has heavily influenced PD process modeling. This perspective views PD as a network of activities that collect, create, interpret, transform, analyze, synthesize, and transfer information. The question of how to structure the information flow—or, more generally, the *deliverable* flow[9]—that spawns the dependencies or *interactions* in an activity network is inseparable from the question of how to structure the activities (actions): they are essentially two sides of the same coin. Eppinger et al. (2001, 1994) use the DSM to structure information flow so that upstream activities will have as much of the information they need as possible before they begin. In the context of activity overlapping, Terwiesch, Loch, and Meyer (2002) present a framework for communicating the accuracy and stability of preliminary information, and in contexts of low ambiguity recommend a *set-based* approach (Sobek, Ward, and Liker 1999) wherein each activity maintains a set of possible solutions to accommodate the variations in its inputs caused by other activities. More generally, Loch and Terwiesch (2005) recommend seven principles of how to exchange information in concurrent processes. Crowston (1997) explores the benefits of *coordination mechanisms* (i.e., the policies and techniques by which the participants in various activities exchange information) in the design and management of the information flow, showing that some process decompositions require a greater number of coordinating activities than others. Overall, however, a majority of PD process models take an activity-centric view,[10] which can be myopic since it reinforces the tendency of many to view the decomposition of a complex process merely as a set of vertical, hierarchical relationships (e.g., a Work Breakdown Structure) while ignoring each level's horizontal relationships, which drive the deliverable flows that give rise to the emergent behaviors of the overall process. Thus, the approach to process decomposition and the choice of activities determines the ease of subsequent coordination and integration of the information flow (von Hippel 1990).

[9] We use "deliverable" in its most general sense, meaning anything that is potentially delivered from one activity to another, including information, materials, artifacts, etc. At times, assumptions can serve as surrogates for some input deliverables.

[10] The Lean philosophy of process improvement also exhibits this tendency, since it assumes a process can be optimized merely by removing non-value-added activities, without regard for the inefficiencies caused by the activity structure itself (Browning 2003).

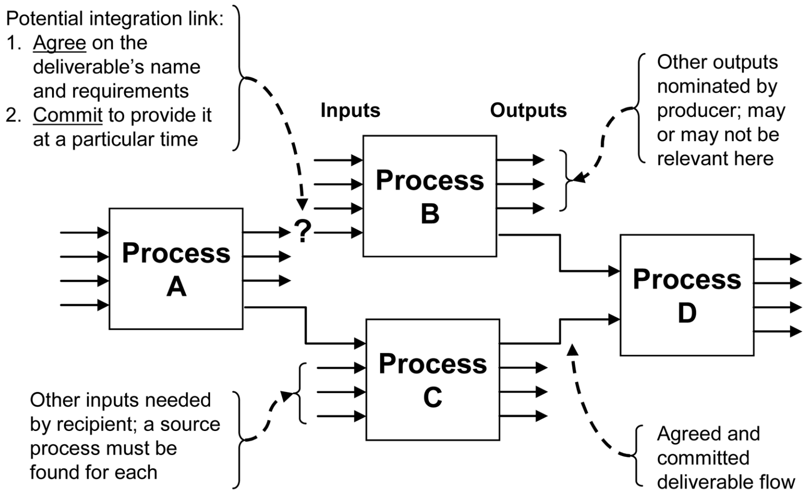
Just as using a *standard process* template can help PD project planners with choosing activities, it may also give them a head start in structuring the process—*if* it accounts for interactions as well as actions. Often planners must integrate standard processes from different organizations, teams, and companies.[11] In the realm of business process modeling, Presley et al. (2001) propose a nested model to support the integration of processes contributed by a company's partners and suppliers, and Casati and Discenza (2001) define a syntax for interactions where activities can elect to publish and subscribe to events (i.e., show and request to see specific information when it becomes available). In a PD context, Fricke et al. (2000) and Browning (2002) use the *input-process-output* (IPO) framework as the basis for process integration and call on project participants (the ones who will actually do the activities) for verification. The IPO model shown in Figure 5 encourages the experts and participants in each process or activity to self-define their requisite inputs and outputs and then verify them by linking each input with its supplier and each output with its customer. Disagreements and disconnects are identified and reconciled. It is thus a precursor to the "network of commitments" approach described in Subheading 4.2.1. A standard process defines a *de facto*, pre-agreed subset of the input-output links to jump-start the commitment-gathering process.

We now capture some of the key insights on structuring the PD process. (i) The number, size, and location of project *reviews* and *iterations* affect both the macro and micro structures of processes. (ii) Reviews often *cause* iterations by creating information that indicates the product design fails to meet expectations. (iii) One can plan and manage the *deliverable flows*, such as information feedback loops, that precipitate iteration. (iv) Iterations are often undesirable, except where they provide high amounts of uncertainty reduction (e.g., rapid prototyping) or can be used to pull activities off the critical path. (v) Activity overlapping becomes less advantageous as *risk* (*uncertainty* that has an impact) increases—i.e., as activities are performed with more assumptions and less firm information. (vi) To avoid iteration, activities must use appropriate information to do their work. That is, their inputs must meet their *entry criteria*. Flawed assumptions or inputs containing mistakes cause unplanned iteration (rework).[12] (vii) To reduce the likelihood of iteration, activities should *coordinate* and *integrate* their deliverable

[11] Process integration is strongly linked to organizational coordination (e.g., Crowston 1997; Morelli, Eppinger, and Gulati 1995) and product integration (e.g., Hong and Hong 2001).

[12] Because iteration is helpful at some times and unhelpful at others, various classification schemes have been proposed, including planned vs. unplanned iterations; see further categorizations in (Unger and Eppinger 2002) and (Clausing 1994, Ch. 3).

Figure 5 IPO framework representation of a process being built by integrating four other processes.



flows and strive to improve their *communication* with other activities in the PD process.

We identify four streams of future research opportunities pertaining to the purpose of structuring PD processes. First, while the overlapping of two activities has received a great deal of attention in the literature, it is important to generalize these methods for an entire activity network. Second, whereas most project management methods focus on activities (actions), new methods and tools could help in planning and managing activity *interactions* (the flow of deliverables). Third, research could develop approaches to accelerate and maintain process *integration*. Fourth, research might further illuminate the tradeoffs between (1) choosing and structuring activities at the outset of a project, when the leverage to affect the outcome and the uncertainty are both high, and (2) staying flexible, so as not to over-constrain downstream options. Perhaps a longitudinal study of the size of the *process space* (the diversity of possible paths to the desired result) over the course of a project could shed some light on the rate at which process options are maintained or eliminated.

4.2.4. Estimating and Improving PD Project Variables. To make schedules and realistic commitments, project planners need estimates of project variables such as duration, cost, output quality, resources, and flexibility—and the uncertainties, risks, and tradeoffs among these. In project management, activity network models have traditionally endeavored to forecast project duration and cost, primarily using the Project Evaluation and Review Technique and the Critical Path Method. However, with the exception of the Graphical Evaluation and Review Technique (e.g., Pritsker and Happ 1966; Neumann 1990), these models *do not capture the iterative nature of the PD process*, which has been shown to drive a significant portion of a project's duration and cost (e.g., Cooper 1993; Osborne 1993). We therefore focus our attention on PD process models that do account for iteration. Belhe and Kusiak (1996a) and Eppinger, Nukala, and Whitney (1997) enumerate all paths and their respective durations in small branching and looping networks. Carrascosa, Eppinger, and Whitney (1998) and Ahmadi et al. (2001) combine aspects of the DSM framework with a Markov model to investigate the effect of partially overlapped activities and learning (which allows activities to go faster when they are reworked), respectively, on project completion time in small networks. However, these models are too difficult to build and solve analytically for networks with more than about 10–20 activities. Hence, Browning and Eppinger (2002) use a simulation model that accounts for activity learning curves, the rework probabilities and impacts (risks) spawned by deliverable flows, and the possibilities of second- and higher-order iterative loops to estimate project duration, cost, and risk. However, all of these PD process time and cost estimation models make the limiting assumptions that all activities and dependencies are known *a priori* and that their durations and costs are independent. Also, they do not account for resource constraints.

Accounting for the effects of resource constraints and allocations, Brucker et al. (1999) and Herroelen (2005) review the vast literature on resource-constrained project scheduling. Yet, these models do not account for iteration. Adler et al. (1995) provide a rich queuing model of iterative PD activities as stochastic processors that receive jobs. They simulate their model to calculate project lead time and resource utilization. Belhe and Kusiak (1996b) schedule resource-constrained PD activities dynamically, based on the importance (to other activities) of the information they generate, to minimize total weighted lateness. Luh, Lui, and Moser (1999) use a stochastic dynamic program to maximize on-time completion for a cyclical network. Yan, Wang, and Jiang (2002) model a stochastic project network to determine the resource requirements at a specified level of utilization to minimize lead time.

A PD project must produce quality outputs, including a product recipe that conforms to requirements. However, the literature is sparse on process models that support this purpose, perhaps because the quality of a product *design* is difficult to verify completely upon its initial availability. Paquin, Couillard, and Ferrand (2000) and Browning et al. (2002) map the influences of individual activities to project-level requirements and use these maps to anticipate the likelihood of meeting specified performance levels. However, these models do not account for activity dependencies. Although many empirical studies explore how a myriad of factors affect project success and quality, these studies do not isolate the impacts of varying the characteristics of the *activity network*. (In what is perhaps the closest empirical study, Harter, Krishnan, and Slaughter (2000) examine the effect of increased process maturity on the quality of the final product.) Overall, we could not find an activity network-based model used to estimate or measure the quality of a project's output, even though it would seem that a model of what work is done and when in a project could provide significant insights in this area.

PD planners also use models to estimate the risks and opportunities facing their projects. The literature on project risk management (e.g., Hillson 2003) deals mainly with listing and mitigating specific, individual threats to and opportunities for a project. The subset of this literature dealing with activity networks mainly employs Monte Carlo simulations to estimate schedule uncertainty for acyclical (non-iterative) networks (e.g., Grey 1995). For iterative projects, Browning and Eppinger's (2002) simulation model estimates cost and schedule risk by weighting each outcome that overruns the budget or schedule by its impact. Similarly, Browning et al. (2002) also use a Monte Carlo simulation model to estimate technical risk by sampling from a distribution of outcomes representing the uncertainties in technical performance parameters and weighting the outcomes that fail to meet requirements by the corresponding loss in customer value. However, while framing cost, schedule, and technical risk estimation in PD projects, these models do not explore the tradeoffs among these factors. Such tradeoffs are critical, because risks in one area are often ameliorated in practice by pushing them into other areas (e.g., by trading time and cost), thus decreasing one area of risk without reducing the overall project risk.

To mitigate unforeseen uncertainties, PD planners may want to estimate process robustness and flexibility/adaptability and their value. Pall (1999) uses the "network of commitments" framework (discussed previously) to design adaptable processes, where a key is to predefine acceptable ranges of interactions (instead of only point values) so robust commitments can be made.[13] Although designing for adaptability, however, this approach does not measure it. Huchzermeier and Loch (2001) measure flexibility in terms of managerial options to delay, abandon, contract, expand, switch, or improve a project. They examine the value of flexibility under five types of uncertainties (payoffs, requirements, budgets, schedules, and performance/quality), beginning with a key insight from options theory: the value of flexibility increases as uncertainty increases (Dixit and Pindyck 1995). In contrast, however, they find that if costs or revenues occur *after* all decisions have been made, then increased uncertainty can reduce the value of flexibility, because this effectively reduces the uncertainty in the "reachable" project outcomes. And if uncertainty reduces the likelihood of flexibility ever being exercised, this also reduces its value. This work could be extended to specific PD processes, however, as it addresses a generic activity network. We could not find an activity network-based model that provides an estimate of the "flexibility index" for PD processes, even though it would seem that a model of what work could be done and when in a project would provide a useful basis for such an estimate.

We notice that many of the aforementioned models address a single project variable such as time, cost, or quality—essentially assuming that others are inconsequential to the results. Some other models explore more than one of these variables together, often to guide tradeoffs among them. For example, many noncyclical activity network models address time-cost tradeoffs, with a basic assumption that expediting activities increases their cost. However, Graves (1989) questions the universality of this assumption, and Brooks (1995) essentially states that going faster costs less. Thus, the full set of time-cost tradeoffs remains unexplored. For cyclical PD processes, Luh, Liu, and Moser (1999) account for the effects of resource constraints and uncertain numbers of activity iterations in determining minimum-time schedules. Browning and Eppinger (2002) show how the structure of an iterative activity network simultaneously affects project time, cost, and risk. Generally, these models show how the *structure* of a PD process can drive project time, cost, etc. Meanwhile, the classical "crashing" models (based on the Project Evaluation and Review Technique and

[13] This process engineering concept is related to the product engineering concept of set-based design (Sobek et al. 1999). Robustness is similarly connected to the concept of *sensitivity* in activity overlapping: a robust process is insensitive to perturbations.

Critical Path Method) show how the *elasticities* of the constituent activities also affect the overall project. Most of the models that address more than one variable focus on time-cost tradeoffs, but these are influenced tremendously by other factors such as design performance (quality) levels and risks. However, we did not find any models that formalized tradeoffs taking into consideration these factors as well.

Our survey of the process models that support estimates of PD project duration, cost, quality, resources, uncertainty, risk, and flexibility leads to several insights. (i) A significant portion of PD project duration and cost stems from iteration and rework (or cycles) in the activity network, but analyzing models that account for iteration is challenging for large networks. (ii) Many of the PD process models that do account for information flow patterns and the resulting activity iterations do not account for resource constraints, and *vice-versa*. (iii) Almost all PD process models assume that the entire set of activities and dependencies can be and are known *a priori*. (iv) There are no activity-network-based models for estimating product quality, and only a few address risk, which we define as the uncertainty weighted by its consequences. (v) Process flexibility can be measured in terms of options, although no process-based "flexibility index" has been developed yet. (vi) Although "speed costs more" is a common assumption, this intuition may not always be true, and the full set of reasons why has not been formalized. (vii) Overall project duration, cost, etc., are the result of both inter-activity (e.g., iteration) and intra-activity (e.g., crashing) effects, although many project management models only address the latter.

We identify several significant opportunities to advance research pertaining to the purpose of estimating and improving project variables. First, traditional assumptions that all activities and their dependencies are known *a priori* and that their durations and costs are independent could be relaxed. However, such an advance may not be possible through mere extensions to existing models: new modeling approaches will probably be required. A significant contribution may be possible through greater synthesis of activity network methods with parametric estimating techniques[14] (Bashir and Thomson 2001). Second, although iterative PD process models often account for information flow constraints, they could also account for resource constraints, which will dramatically alter their results and may also alter many of their insights and recommendations. For example, recommendations to increase activity overlapping (to intensify the exchange of preliminary information) have been made in the absence of consideration of whether the available resources would allow for such concurrency. Third, it seems that process models could be developed to support estimating the quality of an evolving product design. Such a model would lend itself to integration with iterative time-cost models and provide insight into cost-schedule-quality tradeoffs. Fourth, exploration of such tradeoffs might include cost, schedule, and quality *risks*, since one risk area is often mitigated at the expense of another in practice and further since it takes time and money to diminish uncertainties and mitigate risks. Fifth, since process flexibility comes at a price and has a point of diminishing returns, the indicators of such a point would be helpful to identify, along with policies for choosing the optimal level of flexibility. Finally, PD process managers must not only consider tradeoffs among time, cost, quality, risks, resources, and flexibility, they must also remake these decisions over the course of a dynamic project. Existing models do not support these blended decisions, revisited over a rolling horizon. Research could develop integrative models to support balancing project attributes for optimal performance over time.

4.2.5. Allocating PD Resources. Although resources can be reallocated during the course of a PD project, planners must determine an initial allocation. Several models mentioned in the previous subsection have been used to address resource requirements and constraints. At a macro level, resource planning entails shaping the project's budget profile across PD stages. Thomke and Fujimoto (2000) present the benefits and enablers of *front-loading* the profile—i.e., investing in solving design problems and reducing uncertainties earlier. Ahmadi and Wang (1999) and Cohen, Eliashberg, and Ho (1996) find it is optimal to allocate resources to the most efficient and effective PD stage. At a micro level, and accounting for iteration, Joglekar and Ford (2005) and Lee, Ong, and Khoo (2004) map the amount of each type of resource required by each activity, depending on its remaining work, and use this to evaluate process duration. They determine which PD activities and iterations consume disproportionate amounts of resources and use dynamic resource allocation as a project regulator. All five of these models deal only with generic resources, however, whereas PD planners must eventually assign specific individuals, equipment, facilities, etc. to particular activities. These specific resources, used simultaneously on multiple activities and projects, might impose surprising constraints, even when overall resource levels seem adequate. In addition to dealing with such situations that may manifest themselves in practice, an appropriate process model could also enable planners to see the experience, skills, and training

[14] Parametric estimation techniques (e.g., Garvey and Powell 1988) use a regression model of seemingly significant project factors against historical data from similar projects to predict project variables such as time and cost.

required to fill a role on an activity, and they could then quickly ascertain any gaps between the needed and the available resources. Hence, further research is needed to support the allocation of specific resources to specific activities at particular times, and process modeling methodology holds great potential for this purpose. Rather than focusing on the effects of resource constraints alone, this important extension to process models should probably be combined with the others discussed previously to provide insight into their combined effects.

4.3. PD Project Execution and Control Purposes

In addition to supporting PD project visualization and planning, process models can also support ongoing project management (Table 4), first by helping the project manager monitor interim results. Having an agreed-to representation of the work to be done and the interim deliverables to be produced—Pall's (1999) "network of commitments"—is invaluable for driving accountability throughout an organization. In a companion work, Haeckel (1999) develops the metaphor of "management by wire," akin to "fly by wire" in aircraft parlance, to portray monitoring an expansive network of commitments from a project "cockpit" and correcting any breaks. From an information technology network perspective, Ballou et al. (1998) look at the cost, quality, and timeliness of information products as they move between transformers (activities) to determine whether these parameters fall within planned bounds during process execution.

In addition to monitoring commitments kept, managers can use process models to provide other indicators of progress, although progress in PD is difficult to measure. Quantification of progress in PD is difficult because of the difficulty in verifying the quality of interim deliverables (including "information" and "assumptions") and the time lag before final verification and validation are possible (Cooper 1993). In industry, *earned value management systems* (EVMS, e.g., Meredith and Mantel 2003) are popular for tracking deviations from planned schedules and budgets. However, EVMSs ignore iteration, rework (Cooper 1993) and ambiguity and address only cost and schedule, not quality. In another view, by defining the goal of a PD project as the minimization of the risk that the project's outcomes will be unsatisfactory to its stakeholders, Browning et al. (2002) equate progress with uncertainty and risk reduction. Complementarily, Clarkson and Hamilton (2000) define progress as increased confidence in the product design. In contrast to EVMS, these two studies essentially find that progress in PD should be measured as "distance from the finish line" instead of "distance from the starting line" and that this distance becomes clearer as obstructing uncertainties and risks are removed.

Once managers ascertain the current state of a project and compare it with the desired state, they determine if corrective action (re-directing and re-planning) is needed. Process models can be helpful for gauging the impacts of potential changes. A value maximization objective is recommended for evaluating options (Hazelrigg 1998; Browning 2003), and the decision-based design (e.g., Ullman 2001) and decision analysis literatures provide methodological support. Highly iterative and concurrent PD activities may create work for each other as fast as they do it, causing process instability. This effect has been called *design oscillations* (Mihm, Loch, and Huchzermeier 2003) and *design churn* (Yassine et al. 2003), and both models provide strategies for achieving stability for a diverging process or speeding up a slowly converging process. These strategies include speeding up the bottleneck activities (that have the greatest impact on slowing convergence) and delaying responses to changes from highly dynamic activities. Such strategies depend on an analysis of the activity network structure to identify key leverage points. Here, the strategies for corrective action focus on choosing which activities and iterations to do and to what extent.

Closely related are some other PD process models that treat dynamic planning and control as a resource allocation issue. Anderson and Joglekar (2005) reoptimize resource allocation to PD stages over a rolling horizon. Joglekar and Ford (2005) and Lee, Ong, and Khoo (2004) apply techniques from cybernetic control

Table 4 PD Project Execution and Control Purposes

Purpose	Selected references		
★ Are commitments to deliver information and interim results being kept?	(Haeckel 1999)	(Pall 1999)	(Ballou et al. 1998)
★ How much progress has been made?	EVMS (Clarkson and Hamilton 2000)	(Cooper 1993)	(Browning et al. 2002)
★ Given a current state of the project, what is the best direction to go?	(Hazelrigg 1998) (Ullman 2001)	(Muster and Mistree 1988) (Bras and Mistree 1991)	(Browning 2003) (Yassine et al. 2003)
★ How should we dynamically re-plan the project?	(Anderson and Joglekar 2005)	(Joglekar and Ford 2005)	(Lee et al. 2004)

theory to reallocate resources in an activity network. In these models, activity choice is an implicit result of resource allocation. These models deal only with generic resources, however, and further work is needed to account for specific resource constraints, as discussed in Subheading 4.2.4.

The models supporting PD project execution and control provide several insights. (i) Managers should monitor the quality of interim deliverables and commitments kept instead of just activities completed. (ii) Progress equates with confidence in the ability of the hypothesized product design to in fact be the final result, and progress towards a known objective results from removing critical uncertainties (risks). (iii) Corrective actions involve choosing new or iterated activities, and such can be chosen on the basis of their ability to reduce risk and add value. (iv) Analysis of the activity network can reveal points of high leverage for corrective action. (v) Activity selection and activation depend on the allocation of specific resources.

Project execution and control presents several opportunities for PD process modeling research. Research could develop artificially intelligent models that monitor an activity network and appropriately filter the problem areas for the attention of project managers. Such models might provide the basis for *leading indicators* of project problems, such as when the risks of meeting projects goals grow too large. In addition, research could develop improved progress measures for *ambiguous* and *iterative* activity networks. Finally, research could develop process models with sufficient richness to show the impact of changes in stakeholders' values on PD process characteristics. That is, instead of focusing solely on "doing the job right," the models that support project management decisions could also account for "doing the right job" and detecting any deviations from doing so that might emerge.

4.4. PD Project Development Purposes

Process models also serve other purposes from a *project* development and infrastructure standpoint. These include, for example, continuous improvement, organizational learning, training, and compliance. Like visualization (discussed in Subheading 4.1), each of these purposes is probably most properly seen as a means rather than an end. That is, learning, training, compliance, and improvement all have to do with better fulfillment of the other purposes delineated above. Project development purposes are not intended to be ends in themselves, even though they can unfortunately become so when responsibility for them is assigned to a separate group in an organization—e.g., when a process improvement group is so focused on efficiency that it loses sight of higher goals. Except for the purpose of capturing the design process, project development purposes have received little attention in the PD literature. While we focus on these purposes in relation to individual PD projects, they are also of great interest to the multi-project enterprise, which looks for cross-project standards and synergies.

4.4.1. Continuous Improvement Purposes. Continuous improvement methodologies such as Lean, Six Sigma, and Total Quality Management rely heavily on process models. One purpose (Table 5) is to map how work is being done, the "as is" process. However, eliciting the knowledge of PD process participants is notoriously difficult (McCray, Purvis, and McCray 2002). A stream of literature exists—notably in engineering design journals[15] on capturing the design process, using a variety of approaches. An approach associated with Lean manufacturing, value stream mapping, has been ported to the context of PD (McManus 2004). While similar to traditional flowcharting in many respects, value stream mapping adds emphasis on activity attributes such as cycle time, duration, and value-added. Sabbaghian, Eppinger, and Murman (1998) developed a web-based tool to capture distributed process knowledge. Realizing that having a separate group build a PD process model was likely to yield a model without buy-in from those "actually doing the work," the web-based tool distributes the process model data-gathering steps among many people. By changing the model-building paradigm from "a few people taking a lot of time" to "many people taking a little time," this tool enables those actively "doing the work" to model it directly.

Once built, a process model can be analyzed in a variety of ways, depending on the attributes of the real process it has captured. In continuous improvement, key analyses include determining root causes of problems, high leverage points (e.g., critical activities), non-value-adding activities, impacts of potential changes, etc. Overall, the goal is to prescribe a superior "to be" process. Many of the models we have surveyed support such analyses, and some (e.g., Ahmadi et al. 2001; Browning and Eppinger 2002) apply their analyses to recommend improved process structures for real projects.

However, when a particular model accounts for only a subset of the significant factors (e.g., time and cost, but not quality, risk, and resource utilization), its prescriptions are likely to be suboptimal and its advertised improvements a mirage. Thus, future research would be helpful on two fronts. First, we could use more efficient, accurate, and iterative approaches to *building* PD process models. Such developments could also enable improved organizational learning

[15] E.g., *Design Studies*, *Journal of Engineering Design*, and *Research in Engineering Design*

Table 5 Continuous Improvement Purposes

Purpose	Selected references		
★ How is work being done?	Design process capture literature (e.g., Austin et al. 2001)	(Clarkson and Eckert 2005) (Sabbaghian et al. 1998)	(Haque and Pawar 2003) (McManus 2004)
★ How might work be done?	(Ahmadi et al. 2001)	(Browning and Eppinger 2002)	(McManus 2004)

(discussed below). Second, PD process models would provide more helpful process-improvement guidance if they determined "optimality" from a greater number of perspectives than just time and cost.

4.4.2. Organizational Learning and Knowledge Management Purposes. In industry, much project planning and improvement is done "from scratch" for each new project, and the result depends mainly on who is in the meetings and which bits of past documentation are incidentally consulted. For example, many of the process flowcharts created on the walls of conference rooms during improvement meetings (e.g., Lean and Six Sigma's *kaizen* events) are rolled up and put in a corner, never to be thought of again. Not surprisingly, we hear many stories about *organizational forgetting* (e.g., de Holan, Phillips, and Lawrence 2004) and the repetition of past mistakes by new employees. Furthermore, the onerous task of building a rich model of a project's planned process "from scratch" prohibits such a model from being built in most cases. Thus, many projects that could benefit from them do not use rich process models that carry information about a variety of attributes for each of its activities and their relationships, such as, for activities, their estimated cost, duration, inputs, outputs, resources used, entry and exit criteria, etc.

The activity network provides a powerful structure for organizing information about what and how work is done—the "genome" of an enterprise (Browning 2002). Thus, process models can facilitate the buildup and honing of project and enterprise information (Amaravadi and Lee 2005), including the specific purposes in Table 6. In Subheading 4.2.2, we discussed the value of a *standard process* (e.g., Radice et al. 1985; Malone et al. 1999)—also referred to as a *de facto* "network of commitments" in Subheading 4.2.1—for project planning. The act of deciding what and how to model and visualize (Subheading 4.1) a process fosters understanding of that process. More than just "documenting," modeling is the act of sorting out what is known and unknown and can lead to discovery of the "unknown unknowns" ("unk unks" for short). Many of these "unk unks" in a project are actually known to *someone*, but they nevertheless surprise project management because they were not accounted for in the project plans. Building a process model helps expose them. An early benefit of consensus on a standard process model is alignment of the workforce's mental models and vocabularies pertaining to activities and deliverables—a precursor to a viable knowledge network. A standard process that contains the "state of the art" of an organization's knowledge also provides a stable baseline for continuous improvement. Finally, a standard process can also provide benefits in terms of a standard language for information transfer, such as the Process Specification Language (Schlenoff et al. 2000).

Thus, if built upon systematically over time, a process description can become an effective foundation for organizational learning and knowledge management (Davenport, Jarvenpaa, and Beers 1996; Maier and Remus 2002). For example, instead of storing "lessons learned" in a separate database, they could be embedded in a process model as an activity attribute. Model users would thereby be forced to consider past experiences when planning and doing work. And it is not just information about individual activities that matters: as discussed above, it is also information about their emergent, system-level behaviors such as iteration. For instance, Terwiesch and Loch (1999a) suggest combating the engineering change problem by systematically identifying where key couplings lie among activities—i.e., by building up process knowledge. A large, important area for future research could

Table 6 Organizational Learning and Knowledge Management Purposes

Purpose	Selected references		
★ How can we agree on and maintain a common vocabulary of activities and deliverables?	(Radice et al. 1995) (Malone et al. 1999)	(Schlenoff et al. 2000)	(Browning 2002)
★ Where and how should we capture, store, and design process capture literature and access workers' knowledge about work?	Design process capture literature (Brown and Duguid 2000) (Maier and Remus 2002) (Remus and Schub 2003)	(Davenport et al. 1996) (Sabbaghian et al. 1998) (Pilppo et al. 2003)	(Cooper 2003) (Frank 1999) (Bell et al. 2002)

be to bridge the PD and knowledge management literatures (e.g., Cooper 2003). We see process models as a promising structure for this bridge, and we will mention some related research questions at the end of this paper.

4.4.3. Training Purposes. Although underemphasized in the PD process modeling literature, a capable workforce is essential to efficient and effective PD. Activity performance can vary widely depending on the skills and experience of the assigned workers and teams. Process models can help with at least two key purposes: closing skill gaps and organizing real-time delivery of activity instructions and training. If a process is described in terms of the experience, skills, and/or training required to fill each role in each activity, then a skill requirements profile could be compared against the available workers to prescribe optimal staffing assignments. Furthermore, if information about how to accomplish tasks, how to use tools, how to get help, past lessons learned, risks, pitfalls, etc. is attached to each activity in a process model, then the capability exists to push this information to the right people at the right time. We are not aware of research expressly on the use of process models for these purposes, yet this seems to be a large, significant area with many possibilities.

4.4.4. Compliance Purposes. In industry, perhaps nothing has driven the widespread documentation of processes more than the ISO 9000 series of standards. Their requirements for certification specify that a company must, at a minimum, document what it does and then be able to prove it is doing it when audited. Documenting a process is a kind of process modeling, typically using a textual narrative and sometimes supplemented by a flowchart (each of which is a kind of view, as discussed in Subheading 4.1). Unfortunately, the path of least resistance to ISO 9000 certification drives some companies to model their processes in a *purposefully ambiguous* way, so that a wide latitude of actions by the workforce will fall within the bounds of the prescribed process. This situation often leads to misunderstandings among the workforce about the purposes of process models, thereby creating tensions with those who would use process models to *drive out* ambiguity, as discussed in Subheadings 4.1 and 4.4.2. Hence, we see opportunities for further research to address the following questions: What should be modeled about the way work is done, and how much of this model should be subject to audit versus for internal use only? How can a process model be used to convince a stakeholder that a sound and efficient approach to PD is being taken? How can a project internally monitor *process adherence*, which might portend project success, while allowing an appropriate amount of flexibility and adaptation? How can a project balance agility and discipline (Boehm and Turner 2003)?

5. Concluding Remarks

In this work, we have presented a survey of the PD process modeling literature, focusing on activity network-based models to support PD project management. To organize our survey, we used a coherent and fairly complete framework based on four major categories of purposes—visualization, planning, execution and control, and project development—which the models serve. We highlighted the relevant studies, synthesized the key insights, and identified opportunities for further research in each of the four categories. We conclude by highlighting five broad research directions that we feel are especially important for future research in this area.

First, we observe that many of the models place a disproportionate emphasis on *actions* rather than *interactions*. Such models focus on the activities (e.g., the boxes on a flowchart or the bars on a Gantt chart) and their characteristics (such as name, duration, cost, etc.) while giving short shrift to the deliverables (e.g., nominally indicating them with unnamed arrows). However, activity interactions, particularly the flow of deliverables (including information), give rise to a large portion of a PD process's behavior, such as iteration. Although the information processing view of organizations is a helpful motivator for looking at interactivity flows, a majority of models in this stream treat information as a generic quantity, not as a specific item with identifiable attributes that vary and determine the state of a process. PD process models could explicitly consider both activities and the interim *deliverables* that spawn their interactions. Deliverables deserve greater emphasis, because many are overconstraining when one assumes *all* activity input–output relationships imply *firm* precedence relationships. Some dependencies are stronger than others, and while some models (such as numerical DSM models) account for strength of dependency, the identification of unique deliverables with attributes such as maturity, uncertainty, confidence, etc.[16] would be helpful for identifying the potential *process space* in which project planners can seek to optimize the desired balance among cost, duration, technical performance, etc. Deliverables have received more emphasis in recent work on adaptive processes, where the choice and scheduling of activities depends on their required inputs and entry criteria and a project's state.

Second, many of the models focus on local improve-

[16] Browning, Fricke, and Negele (2006) present a PD process modeling framework that balances activity and deliverable characteristics.

ments, like optimizing the relationships between two activities. But it is the project-level improvements that matter, and local optimizations of activity dyads may be sub-optimal from an overall project standpoint. A system-level perspective on both the constituent actions *and* interactions enables a global project analysis. This requires models to determine activity and deliverable criticalities—not merely as a function of the slack in the network—but while accounting for the quality of their outputs, alternative modes of execution, etc. Such determinations might give a truer picture of adjustments to individual activities that will provide the best leverage for the overall project.

Third, although process models hold great potential as an organizing structure for knowledge management, several barriers to realizing this potential exist in that different parts of an organization tend to use disparate models and that the models are cumbersome to maintain. Thus, to better serve the purpose of knowledge management, process modeling frameworks could strive for broader applicability, greater ease of storage and integration, and increased maintainability and reusability.

Fourth, most of the process models assume that all PD activities are known *a priori*. De Meyer, Loch, and Pich (2002) classify projects by their type and amount of uncertainty—variation, foreseen uncertainty, unforeseen uncertainty, and chaos—and suggest that the approach to project management vary accordingly. The approach to process modeling should vary similarly (Lillrank 2002). While the current models are most applicable in cases of "variation" and "foreseen uncertainty"—where over 90% of design activities and relationships can be anticipated from one project to the next (Austin, Baldwin, and Waskett 2000b)—research is needed on process models to support project planning and other purposes in cases of "unforeseen uncertainty" (e.g., Sommer and Loch 2004) and "chaos" (Sommer, Loch, and Dong 2006). In these latter cases, the process might explicitly include activities to *reduce risks* and *discover opportunities*. Natural, adaptive, or emergent processes (Highsmith 2000) such as *bazaar-style development* (Raymond 2001) offer promising avenues for further research.

Finally, as standard processes become more widely used as a basis for project planning and organizational learning, research on tailoring and scaling a standard process for a particular project takes on greater importance. The existing frameworks and models do not provide much guidance for process tailoring and scaling. While pure, mechanistic design of innovative organizations does not work (Dougherty 2001), an appropriate amount of planned process structure yields efficiency (Spear and Bowen 1999; Tatikonda and Rosenthal 2000) by enabling workers to focus their creativity on value-adding actions and coordinate their interactions. Structure can also help prevent the "cutting of corners" and the repetition of past mistakes. Austin et al. (2001) show that some basic process structure is appropriate even for conceptual design. However, the wrong structure over-burdens a project and increases its cost and duration unnecessarily. Therefore, an important research direction would be to explore the right balance between standard processes which can be scaled and tailored and purely innovative processes (e.g., Benner and Tushman 2003).

Acknowledgments

We are grateful to Don Gerwin, Nitin Joglekar, Christoph Loch, Steven Eppinger, Claudia Eckert, and John Clarkson for conversations and comments. An Associate Editor and two anonymous referees also provided helpful guidance that improved earlier versions of this paper.

References

Abdelsalam, H. M. E., H. P. Bao. 2006. A Simulation-based optimization framework for product development cycle time reduction. *IEEE Transactions on Engineering Management* **53**(1) 69–85.

Adler, P. S., A. Mandelbaum, V. Nguyen, E. Schwerer. 1995. From project to process management: An empirically-based framework for analyzing product development time. *Management Science* **41**(3) 458–484.

Ahmadi, R., R. H. Wang. 1999. Managing development risk in product design processes. *Operations Research* **47**(2) 235–246.

Ahmadi, R. H., T. A. Roemer, R. H. Wang. 2001. Structuring product development processes. *European Journal of Operational Research* **130**(3) 539–558.

AitSahlia, F., E. Johnson, P. Will. 1995. Is concurrent engineering always a sensible proposition? *IEEE Transactions on Engineering Management* **42**(2) 166–170.

Alexander, C. 1964. *Notes on the Synthesis of Form.* Harvard University Press, Cambridge, MA.

Altus, S. S., I. M. Kroo, P. J. Gage. 1996. A genetic algorithm for scheduling and decomposition of multidisciplinary design problems. *Journal of Mechanical Design* **118**(4) 486–489.

Amaravadi, C. S., I. Lee. 2005. The dimensions of process knowledge. *Knowledge and Process Management* **12**(1) 65–76.

Anderson, E. G., N. R. Joglekar. 2005. A hierarchical product development planning framework. *Production and Operations Management* **14**(3) 344–361.

Austin, S., A. Baldwin, B. Li, P. Waskett. 2000a. Application of the analytical design planning technique to construction project management. *Project Management Journal* **31**(2) 48–59.

Austin, S., A. Baldwin, B. Li, P. Waskett. 2000b. Integrating Design in the Project Process. *Proceedings of the ICE: Civil Engineering* **138**(4) 177–182.

Austin, S., J. Steele, S. Macmillan, P. Kirby, R. Spence. 2001. Mapping the conceptual design activity of interdisciplinary teams. *Design Studies* **22**(3) 211–232.

Baldwin, C. Y., K. B. Clark. 2000. *Design Rules: The Power of Modularity.* MIT Press, Cambridge, MA.

Ballou, D., R. Wang, H. Pazer, G. K. Tayi. 1998. Modeling information manufacturing systems to determine information product quality. *Management Science* **44**(4) 462–484.

Bashir, H. A., V. Thomson. 2001. Models for estimating design effort and time. *Design Studies* **22**(2) 141–155.

Bassett, M. H., L. L. Gardner, K. Steele. 2004. Dow AgroSciences uses simulation-based optimization to schedule the new-product development process. *Interfaces* **34**(6) 426–437.

Basu, A., R. W. Blanning, A. Shtub. 1997. Metagraphs in Hierarchical Modeling. *Management Science* **43**(5) 623–639.

Belhe, U., A. Kusiak. 1996a. Modeling relationships among design activities. *Journal of Mechanical Design* **118**(4) 454–460.

Belhe, U., A. Kusiak. 1997. Dynamic scheduling of design activities with resource constraints. *IEEE Transactions on Systems, Man, and Cybernetics (Part A)* **27**(1) 105–111.

Belhe, U. D., A. Kusiak. 1996b. Scheduling design activities with a pull system approach. *IEEE Transactions on Robotics and Automation* **12**(1) 15–21.

Bell, D. G., R. Giordano, P. Putz. 2002. Inter-firm sharing of process knowledge: exploring knowledge markets. *Knowledge and Process Management* **9**(1) 12–22.

Benner, M. J., M. L. Tushman. 2003. Exploitation, exploration, and process management: the productivity dilemma revisited. *Academy of Management Review* **28**(2) 238–256.

Bhattacharya, S., V. Krishnan, V. Mahajan. 1998. Managing new product definition in highly dynamic environments. *Management Science* **44**(11) S50–S64.

Bhuiyan, N., D. Gerwin, V. Thomson. 2004. Simulation of the new product development process for performance improvement. *Management Science* **50**(12) 1690–1703.

Boehm, B. 2000. Spiral Development: Experience, Principles, and Refinements. CMU's Software Engineering Institute, Special Report CMU/SEI-2000-SR-008.

Boehm, B., R. Turner. 2003. *Balancing Agility and Discipline: A Guide for the Perplexed.* Addison-Wesley, New York.

Bond, T. C. 1999. Systems analysis and business process mapping: A symbiosis. *Business Process Management Journal* **5**(2) 164–177.

Bras, B., F. Mistree. 1991. Designing design processes in decision-based concurrent engineering. *Journal of Materials & Manufacturing* **100** 451–458.

Brooks, F. P., Jr. 1995. *The Mythical Man-Month: Essays on Software Engineering.* Anniversary edition, Addison-Wesley, Reading, MA.

Brown, J. S., P. Duguid. 2000. Balancing act: How to capture knowledge without killing it. *Harvard Business Review* **78**(3) 73–80.

Brown, S. L., K. M. Eisenhardt. 1995. Product development: Past research, present findings, and future directions. *Academy of Management Review* **20**(2) 343–378.

Browning, T. R. 2001. Applying the design structure matrix to system decomposition and integration problems: a review and new directions. *IEEE Transactions on Engineering Management* **48**(3) 292–306.

Browning, T. R. 2002. Process integration using the design structure matrix. *Systems Engineering* **5**(3) 180–193.

Browning, T. R. 2003. On customer value and improvement in product development processes. *Systems Engineering* **6**(1) 49–61.

Browning, T. R., J. J. Deyst, S. D. Eppinger, D. E. Whitney. 2002. Adding value in product development by creating information and reducing risk. *IEEE Transactions on Engineering Management* **49**(4) 443–458.

Browning, T. R., S. D. Eppinger. 2002. Modeling impacts of process architecture on cost and schedule risk in product development. *IEEE Transactions on Engineering Management* **49**(4) 428–442.

Browning, T. R., E. Fricke, H. Negele. 2006. Key concepts in modeling product development processes. *Systems Engineering* **9**(2) 104–128.

Brucker, P., A. Drexl, R. Möhring, K. Neumann, E. Pesch. 1999. Resource-constrained project scheduling: Notation, classification, models, and methods. *European Journal of Operational Research* **112**(1) 3–41.

Buede, D. M., R. A. Powell. 2001. The Concept of a decision architecture for the systems engineering development process. *Proceedings of the 11th Annual International Symposium of INCOSE*, Melbourne, Australia, July 1–5.

Burns, T., G. M. Stalker. 1961. *The Management of Innovation.* Tavistock, London.

Calantone, R. J., C. A. Di Benedetto. 2000. Performance and time to market: Accelerating cycle time with overlapping stages. *IEEE Transactions on Engineering Management* **47**(2) 232–244.

Carmel, E., S. Becker. 1995. A process model for packaged software development. *IEEE Transactions on Engineering Management* **42**(1) 50–61.

Carrascosa, M., S. D. Eppinger, D. E. Whitney. 1998. Using the Design Structure Matrix to Estimate Product Development Time. *Proceedings of the ASME International Design Engineering Technical Conferences (Design Automation Conference)*, Atlanta, GA, Sept. 13–16.

Casati, F., A. Discenza. 2001. Modeling and managing interactions among business processes. *Journal of Systems Integration* **10**(2) 145–168.

Chakravarty, A. K. 2001. Overlapping design and build cycles in product development. *European Journal of Operational Research* **134**(2) 392–424.

Chen, L., Z. Ding, S. Li. 2005. A formal two-phase method for decomposition of complex design problems. *ASME Journal of Mechanical Design* **127**(2) 184–195.

Cho, S.-H., S. D. Eppinger. 2005. A simulation-based process model for managing complex design projects. *IEEE Transactions on Engineering Management* **52**(3) 316–328.

Chou, S.-C. 2002. ProActNet: Modeling processes through activity networks. *International Journal of Software Engineering and Knowledge Engineering* **12**(5) 545–580.

Christian, A. D., W. P. Seering. 1995. A model of information exchange in the design process. *Proceedings of the ASME International Design Engineering Technical Conferences (Design Theory & Methodology Conference)*, 323–328.

Chung, M. J., P. Kwon, B. T. Pentland. 2002. Making process visible: A grammatical approach to managing design processes. *Journal of Mechanical Design* **124**(3) 364–374.

Chung, P. W. H., L. Cheung, J. Stader, P. Jarvis, J. Moore, A. Macintosh. 2003. Knowledge-based process management—an approach to handling adaptive workflow. *Knowledge-Based Systems* **16**(3) 149–60.

Clarkson, P. J., C. M. Eckert, eds. 2005. *Design Process Improvement: A Review of Current Practice.* Springer.

Clarkson, P. J., J. R. Hamilton. 2000. 'Signposting', a parameter-driven task-based model of the design process. *Research in Engineering Design* **12**(1) 18–38.

Clausing, D. 1994. *Total Quality Development: A Step-by-Step Guide to World-Class Concurrent Engineering.* ASME Press, New York.

Cohen, M. A., J. Eliashberg, T.-H. Ho. 1996. New product development: The performance and time-to-market tradeoff. *Management Science* **42**(2) 173–186.

Conrow, E. H. 2000. *Effective Risk Management: Some Keys to Success.* American Institute of Aeronautics and Astronautics (AIAA), Reston, VA.

Cooper, K. G. 1993. The rework cycle: Benchmarks for the project manager. *Project Management Journal* **24**(1) 17–21.

Cooper, L. P. 2003. A research agenda to reduce risk in new product development through knowledge management: A practitioner perspective. *Journal of Engineering and Technology Management* **20**(1–2) 117–140.

Cooper, R. G. 2001. *Winning at New Products: Accelerating the Process from Idea to Launch.* 3rd ed. Perseus, Reading, MA.

Corbett, C. J., M. J. Montes-Sancho, D. A. Kirsch. 2005. The financial impact of ISO 9000 certification in the United States: An empirical analysis. *Management Science* **51**(7) 1046–1059.

Crowston, K. 1997. A coordination theory approach to organizational process design. *Organization Science* **8**(2) 157–175.

Dahan, E., H. Mendelson. 2001. An extreme-value model of concept testing. *Management Science* **47**(1) 102–116.

Danesh, M. R. 2001. An agent-based decision network for concurrent engineering design. *Concurrent Engineering: Research and Applications* **9**(1) 37–47.

Davenport, T. H., S. L. Jarvenpaa, M. C. Beers. 1996. Improving knowledge work processes. *Sloan Management Review* **37**(4) 53–65.

de Holan, P. M., N. Phillips, T. B. Lawrence. 2004. Managing organizational forgetting. *MIT Sloan Management Review* **45**(2) 45–51.

De Meyer, A., C. H. Loch, M. T. Pich. 2002. Managing project uncertainty: From variation to chaos. *MIT Sloan Management Review* **43**(2) 60–67.

Dixit, A. K., R. S. Pindyck. 1995. The options approach to capital investment. *Harvard Business Review* **73**(3) 105–115.

DoD. 1998. *DoD Integrated Product and Process Development Handbook.* Office of the Under Secretary of Defense (Acquisition and Technology), Washington, DC.

DoD. 2001. DoD Architecture Framework Version 1.0 (Draft). U.S. Dept. of Defense, Instruction 8800.xx.

Dougherty, D. 2001. Reimagining the differentiation and integration of work for sustained product innovation. *Organization Science* **12**(5) 612–631.

Eggersmann, M., S. Gonnet, G. P. Henning, C. Krobb, H. P. Leone, W. Marquardt. 2003. Modeling and understanding different types of process design activities. *Latin American Applied Research* **33**(Special Issue) 167–175.

Elmaghraby, S. E. 1995. Activity nets: A guided tour through some recent developments. *European Journal of Operational Research* **82**(3) 383–408.

Engel, A., M. Barad. 2003. A methodology for modeling VVT risks and costs. *Systems Engineering* **6**(3) 135–151.

Eppinger, S. D. 2001. Innovation at the speed of information. *Harvard Business Review* **79**(1) 149–158.

Eppinger, S. D., M. V. Nukala, D. E. Whitney. 1997. Generalised models of design iteration using signal flow graphs. *Research in Engineering Design* **9**(2) 112–123.

Eppinger, S. D., D. E. Whitney, R. P. Smith, D. A. Gebala. 1994. A model-based method for organizing tasks in product development. *Research in Engineering Design* **6**(1) 1–13.

Fairlie-Clarke, T., M. Muller. 2003. An activity model of the product development process. *Journal of Engineering Design* **14**(3) 247–272.

Finger, S., J. R. Dixon. 1989. A review of research in mechanical engineering design. part i: descriptive, prescriptive, and computer-based models of design processes. *Research in Engineering Design* **1**(1) 51–67.

Ford, D. N., D. K. Sobek. 2005. Adapting real options to new product development by modeling the second Toyota paradox. *IEEE Transactions on Engineering Management* **52**(2) 175–185.

Ford, D. N., J. D. Sterman. 1998. Dynamic modeling of product development processes. *System Dynamics Review* **14**(1) 31–68.

Forsberg, K., H. Mooz, H. Cotterman. 2000. *Visualizing Project Management.* 2nd ed. John Wiley & Sons, New York.

Frank, D. 1999. The importance of knowledge management for BMW *Proceedings of the International Conference on Engineering Design (ICED)*, Munich, Aug. 24–26, 33–40.

Fricke, E., H. Negele, L. Schrepfer, A. Dick, B. Gebhard, N. Härtlein. 1998. Modeling of concurrent engineering processes for integrated systems development. *Proceedings of the 17th Digital Avionics Systems Conference*, Bellevue, WA, Oct. 31-Nov. 6.

Fricke, E., A. Schulz, P. Wehlitz, H. Negele. 2000. A generic approach to implement information-based systems development *Proceedings of the 10th Annual International Symposium of INCOSE*, Minneapolis, July 16–20, 263–270.

Galbraith, J. R. 1977. *Organization Design.* Addison-Wesley, Reading, MA.

Galbraith, J. R., E. E. L. III, Associates. 1993. *Organizing for the Future: The New Logic for Managing Complex Organizations.* Jossey-Bass, San Francisco.

Garvey, P. R., F. D. Powell. 1988. Three methods for quantifying software development effort uncertainty. *Journal of Parametrics*-(March) 76–92.

Gerwin, D., N. J. Barrowman. 2002. An evaluation of research on integrated product development. *Management Science* **48**(7) 938–953.

Girard, P., M. Callot, E. Hugues, H. Kromm. 2002. Performance evaluation of the innovative product design process. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, Yasmine Hammamet, Tunisia, Oct. 6–9.

Goldratt, E. M. 1997. *Critical Chain.* The North River Press, Great Barrington, MA.

Graves, S. B. 1989. The time-cost tradeoff in research and development: A review. *Engineering Costs and Production Economics* **16**(1) 1–9.

Grey, S. 1995. *Practical Risk Assessment for Project Management.* Wiley, New York.

Ha, A. Y., E. L. Porteus. 1995. Optimal timing of reviews in concurrent design for manufacturability. *Management Science* **41**(9) 1431–1447.

Haeckel, S. H. 1999. *Adaptive Enterprise: Creating and Leading Sense-and-Respond Organizations.* Harvard Business School Press, Boston, MA.

Hammer, M. 2001. Seven Insights About Processes. *Proceedings of the Conference, The Strategic Power of Process: From Ensuring Survival to Creating Competitive Advantage*, Boston, Mar. 5–6.

Haque, B. 2003. Problems in concurrent new product development: An in-depth comparative study of three companies. *Integrated Manufacturing Systems* **14**(3) 191–207.

Haque, B., K. S. Pawar. 2001. Improving the management of concurrent new product development using process modeling and analysis. *R&D Management* **31**(1) 27–40.

Haque, B., K. S. Pawar. 2003. Organisational analysis: A process-based model for concurrent engineering environments. *Business Process Management Journal* **9**(4) 490–526.

Harter, D. E., M. S. Krishnan, S. A. Slaughter. 2000. Effects of process maturity on quality, cycle time, and effort in software product development. *Management Science* **46**(4) 451–466.

Hazelrigg, G. A. 1998. A framework for decision-based engineering design. *Journal of Mechanical Design* **120**(4) 653–658.

Herroelen, W. S. 2005. Project scheduling—Theory and practice. *Production and Operations Management* **14**(4) 413–432.

Highsmith, J. A. 2000. *Adaptive Software Development: A Collaborative Approach to Managing Complex Systems.* Dorset House, New York.

Hillson, D. A. 2003. *Effective Opportunity Management for Projects: Exploiting Positive Risk.* Marcel Dekker, New York.

Holman, R., H.-W. Kaas, D. Keeling. 2003. The future of product development. *McKinsey Quarterly*(3) 28–39.

Hong, N. K., S.-G. Hong. 2001. Application of entity-based approach for unified representation of design alternatives for structural design. *Advances in Engineering Software* **32**(8) 599–610.

Hu, J., J. Liu, B. Prasad. 2003. A constraint-driven execution plan for maximizing concurrency in product development. *Concurrent Engineering: Research and Applications* **11**(4) 301–312.

Huchzermeier, A., C. H. Loch. 2001. Project management under risk: Using the real options approach to evaluate flexibility in R&D. *Management Science* **47**(1) 85–101.

Joglekar, N. R., D. N. Ford. 2005. Product development resource allocation with foresight. *European Journal of Operational Research* **160**(1) 72–87.

Joglekar, N. R., A. A. Yassine, S. D. Eppinger, D. E. Whitney. 2001. Performance of coupled product development activities with a deadline. *Management Science* **47**(12) 1605–1620.

Jun, H.-B., H.-S. Ahn, H.-W. Suh. 2005. On identifying and estimating the cycle time of product development process. *IEEE Transactions on Engineering Management* **52**(3) 336–349.

Kara, S., B. Kayis, H. Kaebernick. 1999. Modelling concurrent engineering projects under uncertainty. *Concurrent Engineering: Research and Applications* **7**(3) 269–274.

Khanna, T., M. Iansiti. 1997. Firm asymmetries and sequential R&D: Theory and evidence from the mainframe computer industry. *Management Science* **43**(4) 405–421.

Kline, S. J. 1985. Innovation is not a Linear Process. *Research Management* **28**(2) 36–45.

Krishnan, V., S. D. Eppinger, D. E. Whitney. 1997a. A model-based framework to overlap product development activities. *Management Science* **43**(4) 437–451.

Krishnan, V., S. D. Eppinger, D. E. Whitney. 1997b. Simplifying iterations in cross-functional design decision making. *Journal of Mechanical Design* **119**(4) 485–493.

Krishnan, V., K. T. Ulrich. 2001. Product development decisions: A review of the literature. *Management Science* **47**(1) 1–21.

Kumar, A. V. K., L. S. Ganesh. 1998. Use of Petri nets for resource allocation in projects. *IEEE Transactions on Engineering Management* **45**(1) 49–56.

Kusiak, A. 1999. *Engineering Design: Products, Processes, and Systems.* Academic Press, San Diego.

Kusiak, A., N. Larson. 1995. Decomposition and representation methods in mechanical design. *Journal of Mechanical Design* **117**(3) 17–24.

Lee, S. G., K. L. Ong, L. P. Khoo. 2004. Control and monitoring of concurrent design tasks in a dynamic environment. *Concurrent Engineering: Research and Applications* **12**(1) 59–66.

Lévárdy, V., T. R. Browning. 2005. Adaptive Test Process–Designing a Project Plan that Adapts to the State of a Project. *Proceedings of the 15th Annual International Symposium of INCOSE*, Rochester, NY, Jul 10–14.

Levitt, R.E., J. Thomsen, T. R. Christiansen, J. C. Kunz, Y. Jin, C. Nass. 1999. Simulating project work processes and organizations: Toward a micro-contingency theory of organizational design. *Management Science* **45**(11) 1479–1495.

Lillrank, P. 2002. The broom and nonroutine processes: A metaphor for understanding variability in organizations. *Knowledge and Process Management* **9**(3) 143–148.

Loch, C. H., A. DeMeyer, M. T. Pich. 2006. *Managing the Unknown: A New Approach to Managing High Uncertainty and Risk in Projects.* Wiley, New York.

Loch, C. H., C. Terwiesch. 1998. Communication and uncertainty in concurrent engineering. *Management Science* **44**(8) 1032–1048.

Loch, C. H., C. Terwiesch. 2005. Rush and be wrong or wait and be late? A model of information in collaborative processes. *Production and Operations Management* **14**(3) 331–343.

Loch, C. H., C. Terwiesch, S. Thomke. 2001. Parallel and sequential testing of design alternatives. *Management Science* **45**(5) 663–678.

Lockledge, J. C., F. A. Salustri. 1999. Defining the engine design process. *Journal of Engineering Design* **10**(2) 109–124.

Luh, P. B., F. Liu, B. Moser. 1999. Scheduling of design projects with uncertain number of iterations. *European Journal of Operational Research,* **113**(3) 575–592.

MacCormack, A., R. Verganti. 2003. Managing the sources of uncertainty: Matching process and context in software development. *Journal of Product Innovation Management,* **20**(3) 217–232.

MacCormack, A.D., R. Verganti, M. Iansiti. 2001. Developing products on "internet time": The anatomy of a flexible development process. *Management Science,* **47**(1) 133–150.

Maier, R., U. Remus. 2002. Defining process-oriented knowledge management strategies. *Knowledge and Process Management,* **9**(2) 103–118.

Malone, T. W., K. Crowston, J. Lee, B. Pentland, C. Dellarocas, G. Wyner, J. Quimby, C. S. Osborn, A. Bernstein, G. Herman, M. Klein, E. O'Donnell. 1999. Tools for inventing organizations: Toward a handbook of organizational processes. *Management Science* **45**(3) 425–443.

March, J. G., H. A. Simon. 1993. *Organizations.* 2nd ed., Blackwell Business, Cambridge, MA.

McCray, G. E., R. L. Purvis, C. G. McCray. 2002. Project management under uncertainty: The impact of heuristics and biases. *Project Management Journal* **33**(1) 49–57.

McManus, H. 2004. *Product Development Value Stream Mapping Manual.* Beta version ed., MIT Lean Aerospace Initiative, Cambridge, MA.

Meredith, J. R., S. J. Mantel. 2003. *Project Management: A Managerial Approach.* 5th ed., Wiley, New York.

Michelena, N. F., P. Y. Papalambros. 1995. A network reliability approach to optimal decomposition of design problems. *Journal of Mechanical Design* **117**(3) 433–440.

Mihm, J., C. Loch, A. Huchzermeier. 2003. Problem-solving oscillations in complex engineering projects. *Management Science* **49**(6) 733–750.

Morelli, M. D., S. D. Eppinger, R. K. Gulati. 1995. Predicting technical communication in product development organizations. *IEEE Transactions on Engineering Management* **42**(3) 215–222.

Moser, B., F. Kimura, H. Suzuki. 1998. Simulation of distributed product development with diverse coordination behavior. *Proceedings of the 30th International Seminar on Manufacturing Systems*.

Muster, D., F. Mistree. 1988. The decision-support problem technique in engineering design. *International Journal of Applied Engineering Education* **4**(1) 23–33.

NASA. 1995. *NASA Systems Engineering Handbook.* NASA Headquarters, Code FT, SP-6105.

Negele, H., E. Fricke, E. Igenbergs. 1997. ZOPH–A systemic approach to the modeling of product development systems. *Proceedings of the 7th Annual International Symposium of INCOSE*, Los Angeles, Aug. 3–7, 773–780.

Neumann, K. 1990. *Stochastic Project Networks: Temporal Analysis, Scheduling and Cost Minimization.* Springer-Verlag, Berlin.

Nonaka, I. 1994. A dynamic theory of organizational knowledge creation. *Organization Science* **5**(1) 14–37.

O'Donovan, B. D., P. J. Clarkson, C. M. Eckert. 2003. Signposting: Modeling uncertainty in design processes. *Proceedings of the International Conference on Engineering Design (ICED*), Stockholm, Aug 19–21.

Oppenheim, B. W. 2004. Lean product development flow. *Systems Engineering* **7**(4) 352–376.

Osborne, S. M. 1993. *Product Development Cycle Time Characterization Through Modeling of Process Iteration.* Master's Thesis (Mgmt/Eng), MIT, Cambridge, MA.

Pahl, G., W. Beitz. 1995. *Engineering Design.* 2nd ed., The Design Council, London.

Pahl, G., W. Beitz, K. Wallace. 1984. *Engineering Design.* First English ed., The Design Council, London.

Pall, G. A. 1999. *The Process-Centered Enterprise: The Power of Commitments.* St. Lucie Press, New York.

Paquin, J. P., J. Couillard, D. J. Ferrand. 2000. Assessing and controlling the quality of a project end product: The earned quality method. *IEEE Transactions on Engineering Management* **47**(1) 88–97.

Pentland, B. T. 1995. Grammatical models of organizational processes. *Organization Science* **6**(5) 541–556.

Pentland, B. T. 2003. Conceptualizing and measuring variety in the execution of organizational work processes. *Management Science* **49**(7) 857–870.

Petitcolas, F. A. P., R. J. Anderson, M. G. Kuhn. 1999. Information hiding—A survey. *Proceedings of the IEEE* **87**(7) 1062–1078.

Pich, M. T., C. H. Loch, A. D. Meyer. 2002. On uncertainty, ambiguity and complexity in project management. *Management Science* **48**(8) 1008–1023.

Piippo, P., J. Koivuniemi, H. Kärkkäinen, M. Tuominen, T. Ichimura. 2003. Intranet based system for a product innovation management process. *International Journal of Technology Management* **25**(6–7) 631–642.

PMI. 2004. *A Guide to the Project Management Body of Knowledge.* 3rd ed., Project Management Institute, Newtown Square, PA.

Prasad, B. 1996. *Concurrent Engineering Fundamentals.* PTR Prentice Hall, Englewood Cliffs, NJ.

Presley, A., J. Sarkis, W. Barnett, D. Liles. 2001. Engineering the virtual enterprise: An architecture-driven modeling approach. *International Journal of Flexible Manufacturing Systems* **13**(2) 145–162.

Pritsker, A. A. B., W. W. Happ. 1966. GERT: graphical evaluation and review technique: Part I. Fundamentals. *Journal of Industrial Engineering* **17**(5) 267–274.

Qian, F., Z. Shensheng. 2002. Product development process management system based on P_PROCE model. *Concurrent Engineering: Research and Applications* **10**(3) 203–211.

Radice, R. A., N. K. Roth, J. A.C. O'Hara, W. A. Ciarfella. 1985. A programming process architecture. *IBM Systems Journal* **24**(2) 79–90.

Ramdas, K. 2003. Managing product variety: an integrative review and research directions. *Production and Operations Management* **12**(1) 79–101.

Raymond, E. S. 2001. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary.* Revised edition, O'Reilly.

Reinertsen, D. 1999. Lean thinking isn't so simple. *Electronic Design* **47**(10) 48.

Remus, U., S. Schub. 2003. A blueprint for the implementation of process-oriented knowledge management. *Knowledge and Process Management* **10**(4) 237–253.

Repenning, N. P. 2001. Understanding fire fighting in new product development. *Journal of Product Innovation Management* **18**(5) 285–300.

Roemer, T. A., R. Ahmadi. 2004. Concurrent crashing and overlapping in product development. *Operations Research* **52**(4) 606–622.

Roemer, T. A., R. Ahmadi, R. H. Wang. 2000. Time-cost trade-offs in overlapped product development. *Operations Research* **48**(6) 858–865.

Rogers, J. L. 1999. Tools and techniques for decomposing and managing complex design projects. *Journal of Aircraft* **36**(1) 266–274.

Sabbaghian, N., S. Eppinger, E. Murman. 1998. Product development process capture & display using web-based technologies. *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, San Diego, CA, Oct. 11–14 2664–2669.

Santiago, L. P., P. Vakili. 2005. On the value of flexibility in R&D projects. *Management Science* **51**(8) 1206–1218.

Schlenoff, C., M. Gruninger, F. Tissot, J. Valois, J. Lubell, J. Lee. 2000. *The Process Specification Language (PSL) Overview and Version 1.0 Specification*. National Institute of Standards and Technology (NIST), NISTIR 6459.

SEI. 2002. *Capability Maturity Model® Integration (CMMI^SM), Version 1.1.* Carnegie Mellon University Software Engineering Institute, Pittsburgh, PA.

Shane, S. A., K. T. Ulrich. 2003. Technological innovation, product development, and entrepreneurship in management science. *Management Science* **50**(2) 133–144.

Sim, S. K., A. H. B. Duffy. 2003. Towards an ontology of generic engineering design activities. *Research in Engineering Design* **14**(4) 200–223.

Simon, H. A. 1981. *The Sciences of the Artificial.* 2nd ed., MIT Press, Cambridge, MA.

Smith, R. P., S. D. Eppinger. 1997a. Identifying controlling features of engineering design iteration. *Management Science* **43**(3) 276–293.

Smith, R. P., S. D. Eppinger. 1997b. A Predictive model of sequential iteration in engineering design. *Management Science* **43**(8) 1104–1120.

Smith, R. P., S. D. Eppinger. 1998. Deciding between sequential and parallel tasks in engineering design. *Concurrent Engineering: Research and Applications* **6**(1) 15–25.

Smith, R. P., J. A. Morrow. 1999. Product development process modeling. *Design Studies* **20**(3) 237–261.

Sobek, D. K., A. C. Ward, J. K. Liker. 1999. Toyota's principles of set-based concurrent engineering. *MIT Sloan Management Review* **40**(2) 67–83.

Sommer, S. C., C. H. Loch. 2004. Selectionism and learning in projects with complexity and unforeseeable uncertainty. *Management Science* **50**(10) 1334–1347.

Sommer, S. C., C. H. Loch, J. Dong. 2006. Mastering unforeseeable uncertainty in startup companies: An empirical study. INSEAD, Working Paper.

Song, X. M., M. M. Montoya-Weiss. 1998. Critical development activities for really new versus incremental products. *Journal of Product Innovation Management* **15**(2) 124–135.

Sousa, G. W. L., E. M. V. Aken, R. L. Groesbeck. 2002. Applying an enterprise engineering approach to engineering work: A focus on business process modeling. *Engineering Management Journal* **14**(3) 15–24.

Spear, S., H. K. Bowen. 1999. Decoding the DNA of the Toyota production system. *Harvard Business Review* **77**(5) 97–106.

Sterman, J. D. 2000. *Business Dynamics: Systems Thinking and Modeling for a Complex World.* McGraw-Hill, New York.

Steward, D. 2000. A Very Brief Discussion of Information Driven Business Management: A Problem Solving Approach. Available at: http://www.problematics.com/readings.asp. Accessed February 23, 2007.

Suh, N. P. 1990. *The Principles of Design.* Oxford University Press, New York.

Tatikonda, M. V., S. R. Rosenthal. 2000. Successful execution of product development projects: Balancing firmness and flexibility in the innovation process. *Journal of Operations Management* **18**(4) 401–425.

Taylor, B. W., L. J. Moore. 1980. R&D Project Planning with *Q*-GERT Network Modeling and Simulation. *Management Science* **26**(1) 44–59.

Terwiesch, C., C. H. Loch. 1999a. Managing the process of engineering change orders: The case of the climate control system in automobile development. *Journal of Product Innovation Management* **16**(2) 160–172.

Terwiesch, C., C. H. Loch. 1999b. Measuring the effectiveness of overlapping development activities. *Management Science* **45**(4) 455–465.

Terwiesch, C., C. H. Loch, A. D. Meyer. 2002. Exchanging preliminary information in concurrent engineering: Alternative coordination strategies. *Organization Science* **13**(4) 402–419.

Thomke, S., D. E. Bell. 2001. Sequential testing in product development. *Management Science* **47**(2) 308–323.

Thomke, S., T. Fujimoto. 2000. The effect of 'front-loading' problem-solving on product development performance. *Journal of Product Innovation Management* **17**(2) 128–142.

Thomke, S. H. 1998. Managing experimentation in the design of new products. *Management Science* **44**(6) 743–762.

Tornberg, K., M. Jämsen, J. Paranko. 2002. Activity-based costing and process modeling for cost-conscious product design: A case study in a manufacturing company. *International Journal of Production Economics* **79**(1) 75–82.

Tushman, M. L., D. A. Nadler. 1978. Information processing as an integrating concept in organizational design. *Academy of Management Review,* **3**(3) 613–624.

Ullman, D. G. 2001. Robust decision-making for engineering design. *Journal of Engineering Design* **12**(1) 3–13.

Ulrich, K. T. 1995. The role of product architecture in the manufacturing firm. *Research Policy* **24**(3) 419–440.

Ulrich, K. T., S. D. Eppinger. 2004. *Product Design and Development.* 3rd ed., McGraw-Hill, Inc., New York.

Unger, D., S. D. Eppinger. 2002. Product Development Process Design: Planning Design Iterations for Effective Product Development. MIT Center for Innovation in Product Development, Working Paper.

von Hippel, E. 1990. Task partitioning: An innovation process variable. *Research Policy* **19**(5) 407–418.

Whitney, D. E. 1990. Designing the design process. *Research in Engineering Design* **2**(1) 3–13.

Xiao, R., S. Si. 2003. Research on the process model of product development with uncertainty based on activity overlapping. *Integrated Manufacturing Systems* **14**(7) 567–574.

Yan, H.-S., Z. Wang, M. Jiang. 2002. A quantitative approach to the process modeling and planning in concurrent engineering. *Concurrent Engineering: Research and Applications* **10**(2) 97–111.

Yassine, A., K. Chelst, D. Falkenburg. 1999. A decision analytic framework for evaluating concurrent engineering. *IEEE Transactions on Engineering Management* **46**(2) 144–157.

Yassine, A. A., N. Joglekar, D. Braha, S. Eppinger, D. Whitney. 2003. Information hiding in product development: the design churn effect. *Research in Engineering Design* **14**(3) 145–161.

Yu, B., J. A. Harding, K. Popplewell. 2000. Supporting enterprise design through multiple views. *International Journal of Agile Management Systems* **2**(1) 71–82.

Zachman, J. A. 1987. A framework for information systems architecture. *IBM Systems Journal* **26**(3) 276–292.