Impact of Alternative Flow Control Policies on Value Stream Delivery Robustness Under Demand Instability: a System Dynamics Modeling and Simulation Approach

George W. L. Sousa

Dissertation submitted to the faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
In
Industrial and Systems Engineering

Dr. Eileen M. Van Aken, Co-Chair
Dr. C. Patrick Koelling, Co-Chair
Dr. Kimberly P. Ellis
Dr. F. Frank Chen
Dr. Antonio F. Rentes

September 17, 2004
Blacksburg, VA - USA

Keywords: value stream mapping, delivery performance, lean enterprise design, systems engineering, systems thinking

Impact of Alternative Flow Control Policies on Value Stream Delivery Robustness Under Demand Instability: a System Dynamics Modeling and Simulation Approach

George W. L. Sousa

ABSTRACT

This research explores the effect of proposed management policies and related structures on the dynamics of value streams, particularly under demand instability. It relies on methods from the systems thinking and modeling literature and was designed to fulfill three main objectives.

Objective 1: Provide insight into the causes of problematic behavior in traditional value streams.
Objective 2: Identify modes of demand behavior suitable for pull-based systems operation.
Objective 3: Propose and test alternative value stream management policies and structures.

The achievement of objectives 1 and 3 required the fulfillment of both a hypothetical and a real case. The hypothetical case was designed to describe the problem and improvement alternatives in generic terms, whereas the real case served to contextualize the main generic modeling elements in a real world situation, thus serving as an illustrative example.

The research approach was one based on system dynamics modeling and simulation methodologies that reflect the scientific method. Three alternative policies were created and tested.

Policy 1: a decision rule for altering the number of kanbans in circulation at the protective decoupling inventory during production cycles.
Policy 2: a decision rule for defining the amount of demand to include in value stream schedules.
Policy 3: a decision rule for setting a purposefully unbalanced downstream production capacity.

The results suggest a benefit from the combined use of Policies 2 and 3 in the face of sudden demand peaks. Policy 1 is expected to provide minor benefits but also significantly increase the risk of upstream instability and therefore its use is not recommended. This study provides a causality perspective of the structure of value streams, and gives enterprise engineers new insights into the state-of-the-art in value stream design.

Research Sponsorship

This research was sponsored by AT&T and Danaher Motion via the Enterprise Engineering Research Laboratory at Virginia Tech.

Dedication

This work is dedicated to all enterprise designers and their ability to lead the creation of a better future through the combined use of sensibility, imagination and good planning.

Acknowledgments

Virginia Tech, as an extraordinarily well-crafted educational institution, made it possible to integrate the various elements required for the realization of this research. I found in the Grado Department of Industrial and Systems Engineering the freedom and unique combination of expertise in the fields of production management, systems engineering and systems thinking that inspired me and contributed to shape my professional perspectives.

I would like to thank some great people for their significant contribution to this Ph.D. process. My mentors **Eileen Van Aken** and **C. Pat Koelling** have provided great insight and support, both within and outside the graduate school world. My work as graduate research assistant with Eileen at the Enterprise Engineering Research Lab brought me a very relevant and practical perspective of enterprise design and systems performance, contributing directly to this research. It also made the research effort financially viable. The interaction with Pat has been equally invaluable. His willingness to engage in technical discussions for hours at a time was remarkable. His system's modeling expertise has provided the light needed in many critical decision points.

Kimberly Ellis and **Frank Chen** have contributed to sharpening my understanding of manufacturing systems and bringing the discussion to a technically sound level. Their methodological suggestions have contributed to pave the way not only for this research but also for future efforts. **Antonio Rentes**, my former Masters mentor at University of São Paulo, has continued to support me throughout the Ph.D. program both technically and personally. His expertise in the areas of organizational improvement and lean production have contributed directly to defining the direction of this research as well as in ensuring its applicability in industry.

Luiz Carpinetti, from the University of São Paulo, and **Rick Groesbeck** and **Reinaldo Coelho**, from the Enterprise Engineering Research Lab, have helped me contextualize the use of systems modeling and simulation within the enterprise engineering framework. Our joint engagements in exploring the potential uses of system dynamics modeling tools have been pleasantly insightful. I would also like to thank **John Sterman**, **Jim Hines**, and **Paulo Gonçalves** from the MIT System Dynamics Group. They have shared critical technical insights both at the generic system dynamics methodological level, as well as in the specific modeling of pull production systems.

At last but not least I would like to thank all my family in Brazil. My father **George W. B. Sousa**, my mother **Vera Lúcia Sousa**, my father-in-law **Fernão Castro**, and my mother-in-law **Sônia Castro** all have provided unconditional love and support in many dimensions. In various ways they made this work possible. Despite the geographical distance, they have followed closely the steps of this endeavor – an intense learning experience for us all. As for my wife, **Laura**, who has always faced life with joy and optimism despite the great challenges found along the way, providing me inspiration, support and encouragement at all times, I feel I have no words to express my deepest love and gratitude.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

This research explores the impact of innovative management policies on the performance of value streams. Value streams constitute the building block of lean production systems and have traditionally been vulnerable to demand instabilities.

1.1 Overview

A value stream (VS) is defined by Rother and Shook (1999) as the actions, both value added and non-value added, currently used to bring a product through the main flows essential to every product: (1) the production flow from raw material into the arms of the customer, and (2) the design flow from concept to launch.

In many ways, the concept of value stream aligns with the concept of business process widely used throughout the enterprise engineering literature. Value streams are central components of the lean production (LP) framework. LP has its roots in the Japanese philosophy of just-in-time. As such, value stream structures strongly incorporate the concept of production pull. From a production planning and control perspective, this implies that the use of demand forecasts have a very limited role in triggering production in value streams, i.e., a *pull* mode of operation generically implies that no production takes place until required by actual customer requests.

Most of the available literature in this area qualitatively argues that the effectiveness of *pull-based* systems in general is dependent upon the stability of supplier and internal production processes, as well as the demand environment in which the system operates. With regard to the external demand behavior, however, an unanswered question still is: how unstable is too unstable? In other words, if a LP approach is so appealing for many reasons, how unstable does customer demand need to be to justify the adoption of an alternative production planning and control (PPC) architecture such as *push*-oriented ones?

A typical value stream design is based on a static demand profile and advocates the use of a fixed number of kanbans to regulate the work in process (WIP) inventory. The level of WIP in these systems is controllable and usually set to very low levels – this being one key benefit of this approach. On the other hand, the level of WIP coupled with the available production capacity also affects the ability of the system to respond to undesirable changes in demand. Higher levels of WIP at the right times could prove beneficial for the system in the long run.

The typical behavior of traditional value streams in the presence of demand shocks is illustrated in **Figure 1.1**. Note that even as the demand level varies abruptly, the level of WIP tends to remain relatively constant; decreasing only a little because the inventory location closest to the customer is the one that absorbs most of the shock. On the other hand, backlog (and consequently on time delivery) suffers until the system is eventually capable of returning to its normal mode of operation. However, depending on the magnitude and frequency of the demand shocks, the expected on time delivery performance might never be achieved with this system structure.

Innovative inventory management practices and related policies might have a significant attenuating effect on this condition. For instance, if WIP were allowed to assume a wider range of values, less workstation blocking and starving would be expected, therefore possibly increasing production capacity (Gupta & Al-Turki, 1997). Various other potential improvement alternatives could be tested. Ideally, WIP would increase prior to shocks and then, after absorbing the shock in a timely manner, it would return to its base value. The desired effect would be a "robust delivery capability," i.e., the backlog behavior would remain in high performing equilibrium despite adverse environmental conditions.

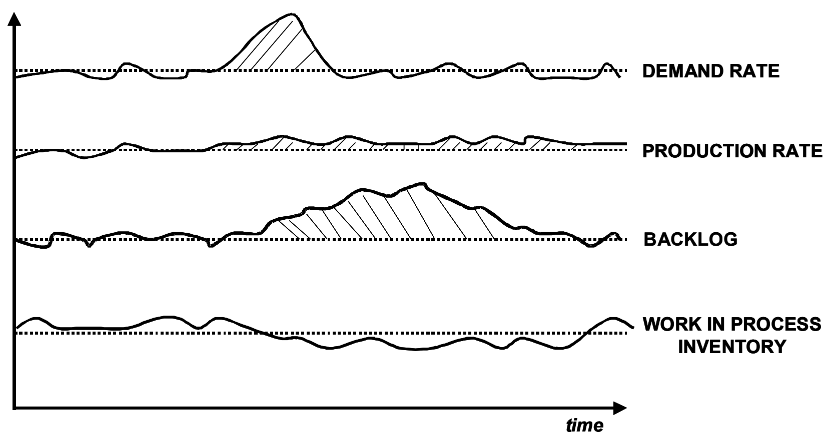


Figure 1.1 *Problematic Mode of Behavior found in Traditional Value Streams.*

Unfortunately, this ideal scenario relies on two unrealistic assumptions: advance knowledge of actual demand and unconstrained production capacity. Given these limitations, what can be done to improve value stream delivery robustness? This research focuses on exploring alternative ways to affect the flow across the value stream, which could individually, or in combination, promote improvement in the system's delivery performance. All along this effort, a conscious decision was to focus on alternatives that are not complex, so they can be harmoniously integrated into the existing, and relatively simple, value stream design guidelines.

The strategy is to focus on the mechanisms of flow regulation in a value stream, i.e., on the conversion of customer requests into production schedules and resource activation along the value stream. A preliminary idea involves the incorporation of a simple flexible kanban system (FKS). The FKS would be capable of altering the number of kanbans circulating in a selected inventory point during production cycles. This would affect the possible ranges of WIP in the system with potentially beneficial effects. In any case, few efforts in general have been reported in terms of quantitatively investigating how alternative flow control policies impact value stream's ability to maintain high delivery performance while responding to demand shocks.

Overall, the nature of this research is understood as a lean enterprise transformation problem. The remainder of this chapter will introduce this research problem in the context of a structured engineering approach, indicating the appropriateness and impact of the selected solution methodology.

1.2 Background

Change is a great constant. The business environment has never been so integrated and volatile. Customers are now more sophisticated and demanding in their preferences. Speed is valued and the use of fast communication technologies is commonplace.

In recent decades, advances in information and logistics infrastructure have improved performance as well as increased the complexity of enterprise operations worldwide by orders of magnitude. As a result, it is not uncommon for customers and suppliers to be located in different continents across the globe and transactions are made in terms of days and not weeks or months, enhancing a competition that is already more fierce than ever before. This highly competitive, integrated and fast-paced environment has forced many firms to develop better ways to cope with uncertainty and instability. Such a need, emphasized by a pressing expectation for continuous improvement, has led enterprises worldwide to engineer themselves in creative ways.

Enterprises have appeared in many types, sizes, and levels of complexity. From the single small business to chains of businesses in an *extended enterprise*, or clusters of businesses in *virtual enterprises*, numerous variations exist. Enterprise operation usually requires an organized cooperation among two or more people, together with other components such as materials, machines, information, knowledge and energy. In all cases, enterprises are human-made undertakings conducted with the purpose of fulfilling customer needs for products and services (Vernadat, 1996).

The creation and operation of enterprises is a phenomenon intimately related to the creation and consumption of products and services (Blanchard & Fabrycky, 1998). In fact, the very existence of an enterprise is contingent upon the provision of specific value added deliverables. Therefore, as long and widely recognized by the industrial engineering discipline, it should be expected that the architecture and life cycle of such products and services provided by an enterprise be intimately related to the architecture and life cycle of the enterprise itself.

The observation of how products/services and enterprises interact reveals complex interrelationships, quite often resembling the famous "chicken and egg" dilemma (see **Figure 1.2**). Sometimes the development of a product/service leads to the creation of an enterprise and sometimes an existing enterprise creates products/services, which in turn leads to the development of new enterprises. In either case, one key imperative from such assertions is valid: given the complexity of today's world, the development of both products/services and enterprises often requires appropriate engineering approaches.

In the current “Systems Age,” enterprises are regarded as complex systems comprised by a network of business processes, the structure of which determines its behavior (Scheer, 1994; Fowler, 1998; Sterman, 2000). Many believe that, despite its complex organic characteristics, such systems can and should be engineered to achieve high levels of performance (Forrester, 1975; Kosanke, Vernadat & Zelm, 1999; Liles, Johnson & Meade, 2002). However, engineering high performing enterprises is an enormous challenge for at least four main reasons.

First, the relationship between product/service architectures and the architecture of enterprises is not obvious. Second, the complex human component brings very unique and usually also poorly understood requirements to the discussion. Third, the current needs for speed, integration, as well as operational reliability in many industries demand the mastering of a sophisticated technological infrastructure, which often imposes severe constraints. And fourth, as other complex open systems, enterprise behavior is particularly vulnerable to external instabilities (Vernadat, 1996).

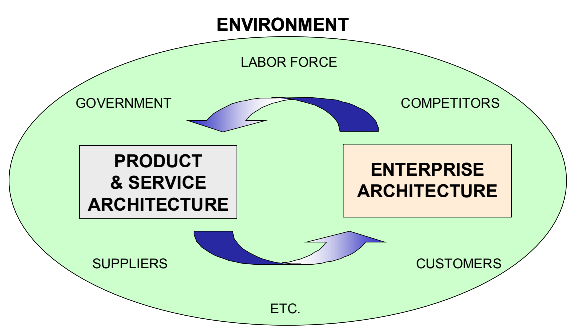


Figure 1.2 *Relationship Between Products/Service Architectures and Enterprise Architectures.*

Considered jointly, these four categories of considerations provide valuable insight into the dynamic relationship between an enterprise and its environment. In fact, as the environment – including products and services but also customers, competitors, suppliers, government, etc. –

changes, an enterprise is required to evolve by changing and adapting its own internal structure accordingly (Kotter, 1996; Rentes, Van Aken & Butler, 1999; Sterman, 2000).

1.2.1 Enterprise Life Cycle

Change requirements imposed by the environment are not always clear, being frequently misinterpreted, taken for granted or ignored, leading to problematic behavior and eventually to enterprise failure at some point. In some special cases, such as in virtual enterprises or project oriented endeavors, the very nature of the need to be fulfilled may require an intentionally short life cycle (Goranson, 1999); however, in most cases, enterprises strive to remain alive for as long as possible. The time period that begins with enterprise creation and ends with enterprise retirement defines its life (Fine, 1998). **Figure 1.3** illustrates these ideas.

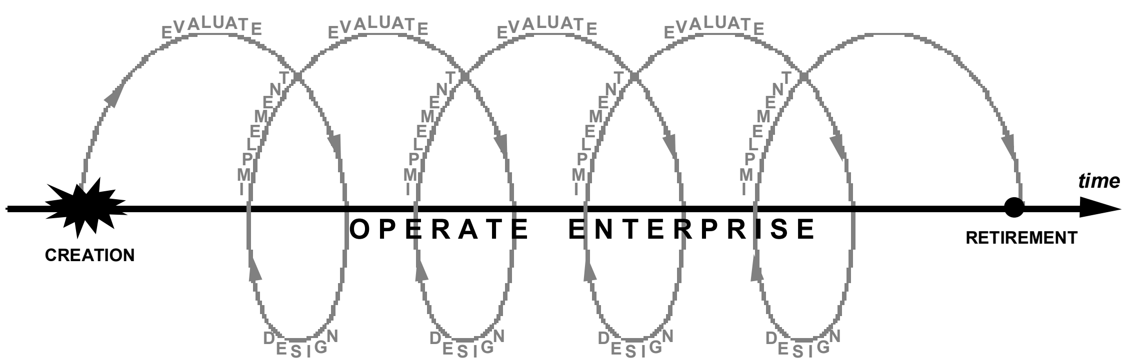


Figure 1.3 *Enterprise Life Cycle.*

During its life, an enterprise, in whole or in part, might get engaged in one or several cycles of transformation, involving evaluation, redesign, and implementation. These cycles are more or less formal undertakings that may take place whenever a significant new need (i.e., changed requirement) or opportunity is identified (Blanchard & Fabrycky, 1998). Various cycles may take place simultaneously targeting various parts of the system. Each one of them involves designing and bringing a new order of things into being.

These cycles of transformation are natural and should be expected in most businesses. Nevertheless, comprehensive enterprise (re)design approaches are still rare in the literature. Partly due to this reason many entrepreneurial attempts as well as improvement efforts in existing organizations fail, and it is not uncommon to identify enterprise problems that can be traced back to poor original designs (Forrester, 1998).

Enterprise redesign involves two important facets: managerial and technical. From a managerial perspective, redesigning an enterprise and implementing a new order of things is intimately related to the concept of leadership: it usually requires the sharing of the creative vision of one or more leaders to get things changed and improved. Along these lines, Kotter (1996) provides a list of 8 common errors and associated consequences encountered in change efforts (see **Table 1.1**).

Kotter (1996) also points out that by any objective measure, the amount of significant, often traumatic change in enterprises has grown tremendously over the past twenty years. Nevertheless, it is expected that the pressure on enterprises to change will be even higher during the next decades. Current common methods adopted in attempts to cope with change include: total quality management, reengineering, right sizing, restructuring, cultural change, and turnarounds among others. The author argues that the main reason why these methods are frequently unsuccessful is because they fail to alter behavior.

Successfully altering the behavior of individuals and organizations is a complex matter. It demands learning and altering their fundamental structural nature; a task that requires experience, profound knowledge, adequate planning, and the right tools (Senge, Kleiner, Roberts, Ross & Smith, 1994).

Table 1.1 *Errors Common to Enterprise Change Efforts and their Consequences. Source: Kotter (1996)*

Common Errors
Allowing too much complacency
Failing to create a sufficiently powerful guiding coalition
Underestimating the power of vision
Under communicating the vision by a factor of 10 (or 100 or even 1,000)
Permitting obstacles to block the new vision
Failing to create short-term wins
Declaring victory too soon
Neglecting to anchor changes firmly in the corporate culture
Common Consequences
New strategies aren't implemented well
Acquisitions don't achieve expected synergies
Reengineering takes too long and costs too much
Downsizing doesn't get costs under control
Quality programs don't deliver hoped-for results

Keating, Oliva, Repenning, Rockart, and Sterman (1999) reinforce these ideas affirming that according to their findings, most improvement programs fail because of the inability to manage them as a dynamic process. This brings the discussion to a more technical facet of enterprise design. From a systems modeling point of view, current enterprise transformation methods seem to lack procedures for generating and testing enterprise designs before they are implemented. In fact, Vernadat (1996) advocates that enterprises should be engineered in a systematic way similar to how software is engineered. Pressman (1995) argues that only recently the software engineering community started to apply systematic design efforts (including testing and evaluation) at the very early stages of the software life cycle, achieving significant, often radical, improvements in results.

In the past, little attention was given to software conceptual design. The majority of development time was spent directly in coding and attempting to remove errors and figuring out missing or redundant subroutines. Quite often such practices led to the creation of software of doubtful quality and effectiveness. Such scenarios seem to resemble very well the current situation of most enterprise development efforts.

As Biemans, Lankhorst, Teeuw, and Wetering (2001) point out, engineers are trained to design systems such as bridges, aircraft, computers, and software in a well-structured manner, however the design of enterprises has not yet matured to this level. Improvements in this area will require a good understanding of customer needs, the nature of business processes, the human component, the supporting technology, the engineering process itself, as well as how enterprise performance and enterprise structure are interrelated.

1.2.2 The Impact of the Enterprise Structure on its Performance

The new order of things or arrangement of system components created by an enterprise transformation effort (in the form of resources, policies, organizational hierarchy, etc.) reflects the system structure (i.e., the overall enterprise architecture) adopted (Scheer, 1994; Vernadat, 1996). This structure can be perceived in the relationships among components and consequently determines the possible modes of behavior the total enterprise system can sustain (Sterman, 2000).

Behavior and performance are two very closely related concepts. Behavior can be interpreted as the observation of one or more measures over time. Among the enormous number of all possible measures, a few are selected according to a paradigm that suits the purpose at hand and adopted as indicators of the system's performance. Therefore, desired performance may be interpreted by means of the desired behavior as described by a few strategically defined measures. For instance, consider the typical enterprise growth behavior (measured in terms of the volume of sales) described by Forrester (1975) in the example provided in **Figure 1.4**.

In this example, Forrester explains that curve A represents a very rare type of enterprise, which simply grows healthy and without obstacles during all its life cycle. More frequently, however, the behavior described by curve B is found, where after an apparent period of success, a sequence of crises leads to bankruptcy or selling of the business. The behavior described in C is also common and represents stagnation. However, among the enterprises that present growth tendencies, it is argued that the most common pattern found is the one described by curve D, where growth is accompanied by repeated crises.

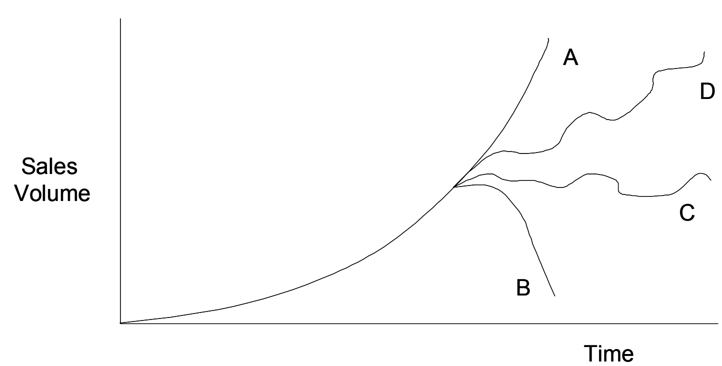


Figure 1.4 *Patterns of Growth in an Enterprise.*
Source: taken from Forrester (1975)

Overcoming crises in order to promote growth is likely to require a new order of things obtained through a transformation effort of some sort. Therefore, the task of transforming an enterprise may be understood as an effort to change the structure of the system so that the new structure is capable of engaging in a desired behavior (see **Figure 1.5**).

Consequently, transformation requires that a precise understanding of the basic purpose of the system (mission), the new strategies selected to fulfill the purpose, and the desired performance, be translated into a design of the new enterprise structure (architecture). The cycle of transformation is completed only when the new design is implemented and the real system is physically modified and made operational under the new configuration. The potential complexity involved in this task is further reinforced by the fact that changes usually need take place while keeping the enterprise operational at all times (please refer back to **Figure 1.3**), which also enhances the need to account for the dynamics involved in the periods of transition.

Conceiving an appropriate enterprise structure is a task that can greatly benefit from past experiences by considering how certain typical architectural types (i.e., enterprise archetypes) are capable of promoting certain typical behaviors (Senge et al., 1994). This idea emphasizes a fundamental systems concept, which provides rational meaning to the design activity: the behavior of a system is a function of its structure (Forrester, 1961; 1998; Fowler, 1999; Sterman, 2000).

In practical terms, this concept reflects the fact that the creative work of enterprise designers can often make use of pre-conceived arrangements or components as the total design solution is synthesized. Likewise, certain key characteristics may be expected from enterprise designs according to their basic configuration, just as happens with the design of other complex systems. These characteristics are often grouped in categories such as technical, economical, or social; and are related to performance indicators such as quality, efficiency, and effectiveness among others (Blanchard & Fabrycky, 1998; Rolstadas, 1998).

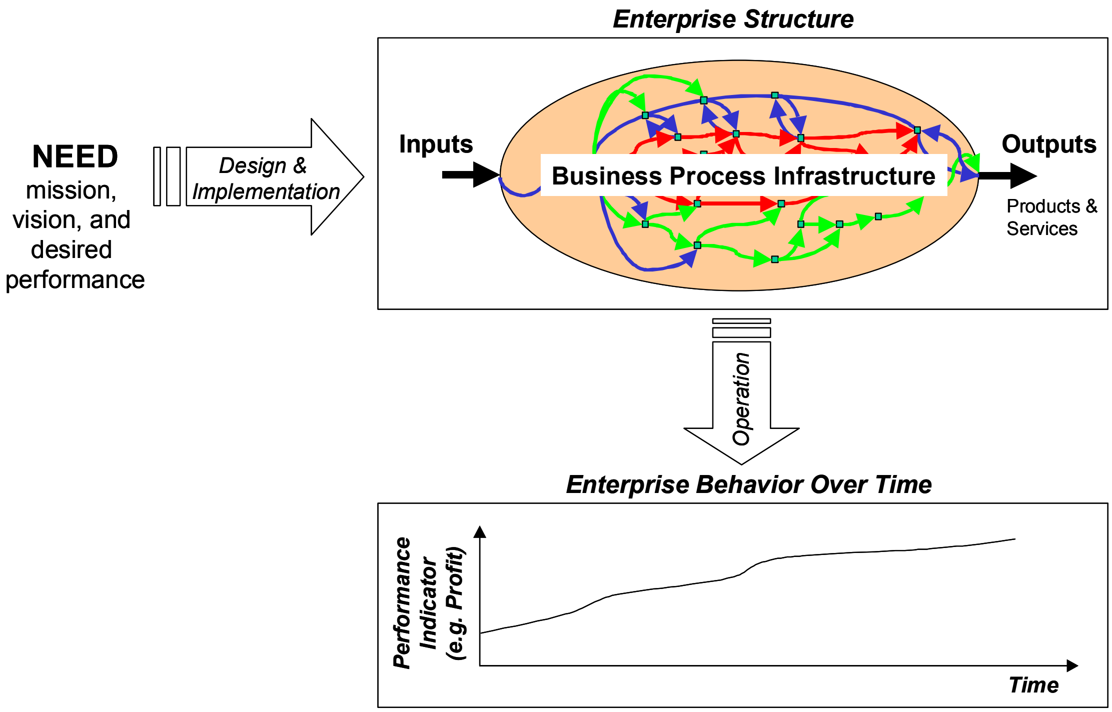


Figure 1.5 *Enterprise Behavior Over Time (Performance) is a Function of the Enterprise Structure (Architecture).*

For the purpose of illustration, consider the case of another complex system: a commercial aircraft. When in operation it is expected that the aircraft will fly from origin to destination on time, safely transporting passengers and goods, despite variations that may occur in terms of changes in visibility, air pressure, wind velocity, temperature, humidity, etc. Similarly, it is also expected that an enterprise will deliver quality products and services on time and at an affordable price, despite variations in customer demand, commodity prices, communication difficulties, supplier delivery performance, quality of raw materials, and resource efficiency. In both cases, the common theme is that the design of the system in question needs to reflect a structure robust enough to ensure the desired performance during system use, despite a range of adverse conditions.

1.2.3 Enterprise Design

Three key ideas are related to any system or subsystem design or redesign effort and therefore are applicable to the enterprise case (Wortmann, Muntslag & Timmermans, 1997).

- To design implies choosing from alternatives or design choices (descriptive of the system structure) by means of performance indicators.
- The designer creates design choices in an attempt to fulfill a purpose and optimize system performance.
- Design choices are related to performance indicators by models of some sort (see **Figure 1.6**).

The designers of complex systems usually break the design problem into parts for reasons of reducing complexity, sharing complementary expertise, as well as enabling collaborative efforts to reduce design lead-time. The enterprise design problem, in particular, is typically broken into complementary dimensions such as: functional, organizational, informational, products/services, resources, and process/control (Vernadat, 1996; Whitman, Huff & Presley, 1998; Scheer, 1999a; Yu, Harding & Popplewell, 2000; Sousa, Van Aken & Rentes, 2001).

Consequently, the design effort (as a process) requires a common language, i.e., a minimally structured form of representation of the various parts that compose the object of study. This means that there is a need for some sort of modeling formalism capable of both representing the various subsystems and components appropriately as well as allowing an integrated representation of the whole so that evaluations of the total system can be made.

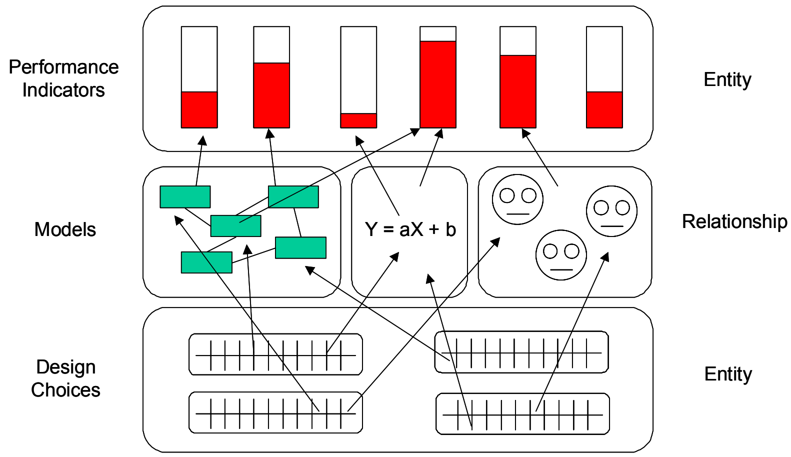


Figure 1.6 *Relationship Between Design Choices and Performance Indicators. Source: adapted from Wortmann et al. (1997)*

Furthermore, in addition to this classification in terms of dimensions, enterprise models are also classified in terms of the stage of the design process in which they belong. As advocated in the systems engineering literature, a typical design effort can be divided into three main stages: (1) initial analysis and conceptual design, (2) preliminary design, and (3) detailed design. This research fits in the first two stages of the enterprise design problem and focuses on the operational dynamics of value streams. As such, it adopts the system dynamics (SD) modeling and simulation language as the primary representation formalism of choice. **Figure 1.7** illustrates these ideas.

The adoption of a SD modeling approach during the conceptual enterprise design stage has two critical advantages. First, it allows for the quantitative assessment of behavior at an aggregate/strategic level while maintaining a clear identification of the units (e.g., specific products) flowing through the system across any organizational boundaries. Second, the nature of this modeling approach, by identifying and making explicit key feedback control loops in the system, is especially appropriate for considerations regarding the dynamic interrelationships among individual enterprise business processes and their effect on the entire system architecture at a strategic level (Fowler, 1999).

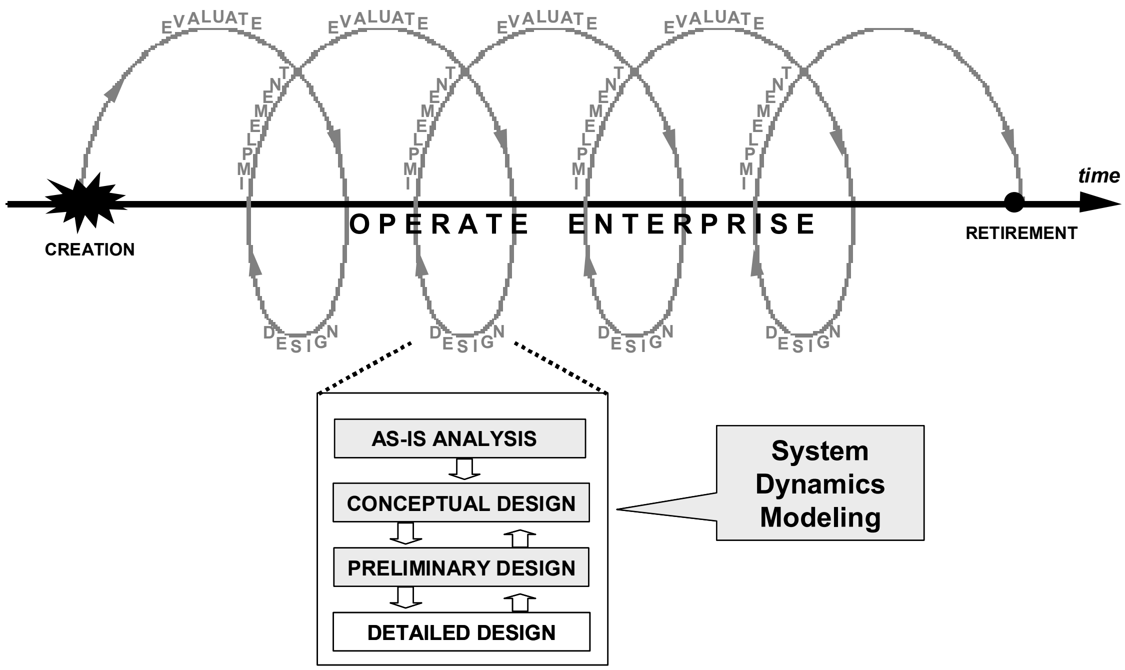


Figure 1.7 *Focus on Conceptual Enterprise Design Using System Dynamics Modeling.*

Interestingly, according to Womak and Jones (1996) these same advantages were also the very weaknesses of the reengineering movement; which, despite having correctly recognized that departmentalized thinking is suboptimal and tried to shift the focus from organizational units (departments) to value-creating processes, has lacked an integrated and quantitative approach at a strategic/conceptual level.

Furthermore, SD models developed and tested at conceptual and preliminary levels can be further deployed throughout the detailed design stages using various process-oriented modeling techniques (e.g., IDEF1/3 – Integrated DEFinition, eEPC – extended Event Process

Chain, and VSM – Value Stream Mapping), allowing the promotion of a qualitative top-down functional decomposition typical of successful system development strategies.

In quantitative terms, as appropriate to the design effort, the continuous simulation promoted by the SD approach can in many cases be converted into discrete-event simulation (Sterman, 2000) using adequate representation formalisms and time lag considerations. This procedure can, on the one hand, enrich the functional decomposition approach by ensuring quantitative consistency as increased level of detail is generated for critical subsystems, and, on the other hand, allow for the integrated assessment of systems and subsystems – a modeling feature critical during bottom-up checks for physical realizability and expected overall performance.

1.2.4 System Dynamics Modeling

Models are simplified representations of reality and serve the purpose of capturing a point of view of interest. By definition no model is complete in itself, which means that a model can always be complemented by other models. Nevertheless, as models filter irrelevant details from reality they enable the manipulation of the perceived complexity (Vernadat, 1996).

Two types of complexity can usually be identified in business environments: combinatorial and dynamic. As described by Sterman (2000; 2001), *combinatorial* or *detail complexity* can be exemplified by the problem of scheduling an airline's flights and crew, which involves finding the best or a good enough solution out of a very high number of possibilities.

Dynamic complexity, on the other hand, may arise even in simple systems with low combinatorial complexity and is related to the interactions among agents over time (Wilding, 1998; Sterman, 2000). Vernadat (1996) defines an agent as an active entity able to perceive, reason, and act as well as to communicate with other agents or with its environment by synchronously exchanging messages to provide a service (i.e., to realize some function). According to this definition, examples of agents may include humans, automated machines, or even software applications.

For instance, the decisions made by managers (i.e., human agents) form feedback loops that operate in any given system. These loops react to the decision maker's actions in ways that are both anticipated and unanticipated. Furthermore, these loops may contain many variables and nonlinearities creating complex dynamics.

System dynamics modeling is especially appropriate to address issues pertaining to dynamic complexity. As a modeling formalism, it consists of a set of tools to describe the structure of systems. It also involves the construction of formal simulation models to assess system behavior (Disney, Naim & Towill, 1997). For instance, fundamental modes of dynamic behavior include exponential growth, goal seeking, and oscillation (Sterman, 2000).

Models developed under this paradigm are usually based on descriptions of feedback loops, stocks and flows, and decision-making rules. Feedback loops represent a chain of causality and can be of two types: positive or negative. If positive, they are self-reinforcing. If negative, they are self-correcting. Stocks are created by accumulating the difference between the inflow to a process and its outflow. Decision rules or policies represent the criteria used by an agent to regulate one or more flows in an attempt to drive the system to a desired state (Sterman, 2000). For instance, the inventory of final products in a manufacturing enterprise (a stock) is affected by the decision rules (policies) embedded in a control cycle (feedback loop). It is increased by the rate (flow) of production and decreased by the rate (flow) of shipments.

1.2.5 Enterprise Production Planning and Control Architectures

Various types of flows take place in enterprise systems. At least two types are widely recognized: (1) material flows and (2) information flows. The first is usually part of the conversion of raw materials or semi-finished parts into final or semi-final products. The information flows on the other hand are less obvious, serving to regulate (i.e., control) the flow of materials as well as to generate services (Towill, 1996a; Mason-Jones & Towill, 1998; Scheer, 1999a). Special interest in this research lies in the dynamic interrelationships between material and information flows, especially when the information flows are part of a production planning and control (PPC) architecture.

A PPC architecture may be interpreted as the structure that makes production control systems out of lower level components such as sensors, databases, control algorithms, decision-makers, and actuators (Wiendahl & Breithaupt, 1999; Little, Peck, Rollins & Porter, 2001). Its main purpose is to regulate the rate of production so that it matches the rate of final consumer sales. This is accomplished through enactment of production control policies by means of implementation through management (or control) processes (Wilding, 1998; Kurstedt, 2000). Consequently, a production control mechanism is, by definition, a required component of any service or manufacturing enterprise (Sipper & Bulfin Jr., 1997).

In practice, this means that management business processes regulate the flow of goods and services along core business processes. Nevertheless, Powell, Schwaninger, and Trimble (2001) emphasize that while a myriad of different approaches have been proposed for specifying and managing enterprise business processes (BP), it has been taken for granted the fact that any well-engineered BP is one in which management establishes measurements of process performance, and influences process performance in a desired direction by using these measurements to control the process.

Control involves, among other things, the regulation of levels of resources available to the process. Thus, the notion of feedback control is a key part of BP engineering efforts. In more technical terms, feedback control basically involves controlling the behavior of a BP by measuring its current state, comparing it to a desired state, and selecting and implementing control actions based on differences between actual and desired states in order to bring the state of the system closer to the goal(s). Despite the pervasiveness of this concept, Powell et al. (2001) make the strong observation that while the science of feedback control is highly evolved in more traditional engineering disciplines (e.g., electronic engineering), in enterprise engineering it is still mostly at the elementary level of analogy.

1.3 Scope of Research

This investigation focuses on incorporating a control perspective into the enterprise design domain. The long-term intent of this research stream is to develop innovative value stream management structures that are more capable of coping with instabilities.

1.3.1 Purpose

The particular purpose for this dissertation is to develop alternative flow control policies and evaluate their impact on value stream delivery performance under external demand instability.

1.3.2 Objectives

Three key objectives have been established for this research:

1. Create a comprehensive explanation of how the behavioral characteristics of traditional value streams emerge as a result of the system structure;
2. Identify and describe modes of demand appropriate for pull-based system operation;
3. Propose and test alternative flow control solutions.

1.3.3 Research Approach

This investigation adopts a research approach based on system dynamics modeling and simulation. System dynamics modeling methodologies have their root in the scientific method and involve the generation and testing of formal dynamic hypotheses.

1.3.4 Research Contributions

Overall, this research aims to contribute to the practice of enterprise design, particularly by enhancing the comprehension of how the structure of value streams relates to their performance. Some specific expected outcomes are listed below:

- A text that connects tools of systems thinking with the systems engineering approach for the explicit purpose of conducting enterprise design;
- An initial theory about why *pull* systems are vulnerable to instabilities, as well as the documentation of causality relationships among key value stream variables (as part of the hypothetical case in **Chapter 4**);
- A system dynamics simulation model that can be used as a building block in future, more sophisticated modeling efforts;
- A classification of demand profiles with insight into the types appropriate for high performance value stream operation;

- Feasible alternative value stream flow control solutions and their estimated impact in a real case.

Next, **Chapter 2** presents a review of the pertinent literature. For more details on research purpose, objectives, and approach, please refer to **Chapter 3**. **Chapter 4** develops and presents the hypothetical case. **Chapter 5** contextualizes the hypothetical findings in a real application in a manufacturing enterprise. **Chapter 6** summarizes conclusions, as well as provides suggestions for future research.

Chapter 2
Literature Review

The literature reviewed in this research covers three broad areas: (1) enterprise engineering, (2) system dynamics modeling and simulation, and (3) production planning and control.

The concept of ***enterprise engineering*** represents the overall context in which this research was conducted, emphasizing the engineering approach, the use of enterprise modeling techniques, and the importance of testing and evaluation in the conceptual and preliminary design phases at the early stages of the enterprise transformation cycle.

System dynamics is the modeling technique of choice in this research. Key concepts are reviewed, including feedback loops, the relationship between structure and behavior of a system, as well as process instability. The modeling methodology is presented and technical concepts pertaining to the simulation aspects, as appropriate to the research, are emphasized.

Conceptual enterprise design using system dynamics modeling and simulation is likely to be a very complex endeavor involving many and diverse facets depending on the problems at hand. In the third area of literature review, ***production planning and control,*** the focus is shifted from the model-based engineering approach to the object being engineered. Attention is placed on the management infrastructure that regulates flows in a production system, at the core of any service or manufacturing enterprise. Two basic control modes – *pull* and *push* – are described, compared, and placed within the generic context of lean production (LP) design and operation guidelines. The concepts of value stream and kanban are reviewed as central components of the

LP framework. The state of the art in control solutions is presented. Last, the vulnerability of lean production systems to demand instabilities is discussed and the concept of robustness is reviewed.

For a summary of literature findings organized according to the three areas of review, please refer to **Appendix A: Classification of Literature Findings**. The discussion on production planning and control (PPC) is presented in this chapter. For details on the enterprise engineering and system dynamics materials, please refer to **Appendix B: Background and Extended Literature Review**.

The PPC literature reviewed in this chapter is divided into two parts. First an overview of the history, main components, hierarchy, and strategies of PPC systems is summarized. Next, a more focused discussion of flow control policies under demand instability is provided. Complementarily, a further detailed review of key elements and practices embedded into value streams structures is presented in **Chapter 4** as part of the hypothetical case development.

2.1 Overview of Production Planning and Control Systems

Converting raw materials into final products is an inherent activity of human civilization (Blanchard & Fabrycky, 1998). As a socio-physical phenomenon involving various interacting components it requires a minimum level of control and coordination. One of the oldest records of organized production activity points out that as far back as 5000B.C. Sumerian priests were already keeping records of inventories, loans, and tax transactions. However, despite major achievements obtained along the following millenniums by the Egyptians, Hebrews, Chinese, Greeks, and Italians, significant production management theories and technologies started to appear only in the early 1700s at the outburst of the European Industrial Revolution. Considered under such a time reference, organized production through complex industrial systems is an extremely recent human endeavor (Sipper & Bulfin Jr., 1997). **Figure 2.1** summarizes the evolution of key production technologies and theories.

The last five decades have marked the progressive consolidation of a systems view of the world. According to the paradigm and terminology adopted in this so-called "systems age," production takes place by means of a production system. A production system is put together in order to attend a predetermined purpose by promoting the conversion of inputs into outputs – therefore providing some sort of value to the ultimate customer. Inputs are generally in the form of raw materials, information, and energy. Production outputs may be realized in the form of goods (tangible) or services (intangible), being normally associated with manufacturing and/or service enterprises.

In service enterprises the generation of production outputs and its consumption by a customer usually occurs simultaneously. In such cases, the customer (or user) "uses" the service enterprise directly in order to have a need fulfilled. For instance, consider the case of a telecommunications enterprise such as AT&T. When a customer places a telephone call, he or she is using AT&T directly. In other words, although the service (i.e., the capability of communicating to one more parties) is intangible, the customer uses AT&T's physical infrastructure directly to place the call. A key implication of this fact is that, in order to maintain high performance levels, enterprise production power has to be very well aligned with demand levels at all times. Hence, the critical need for accurate staffing and other production capacity settings observed in many large-scale service enterprises.

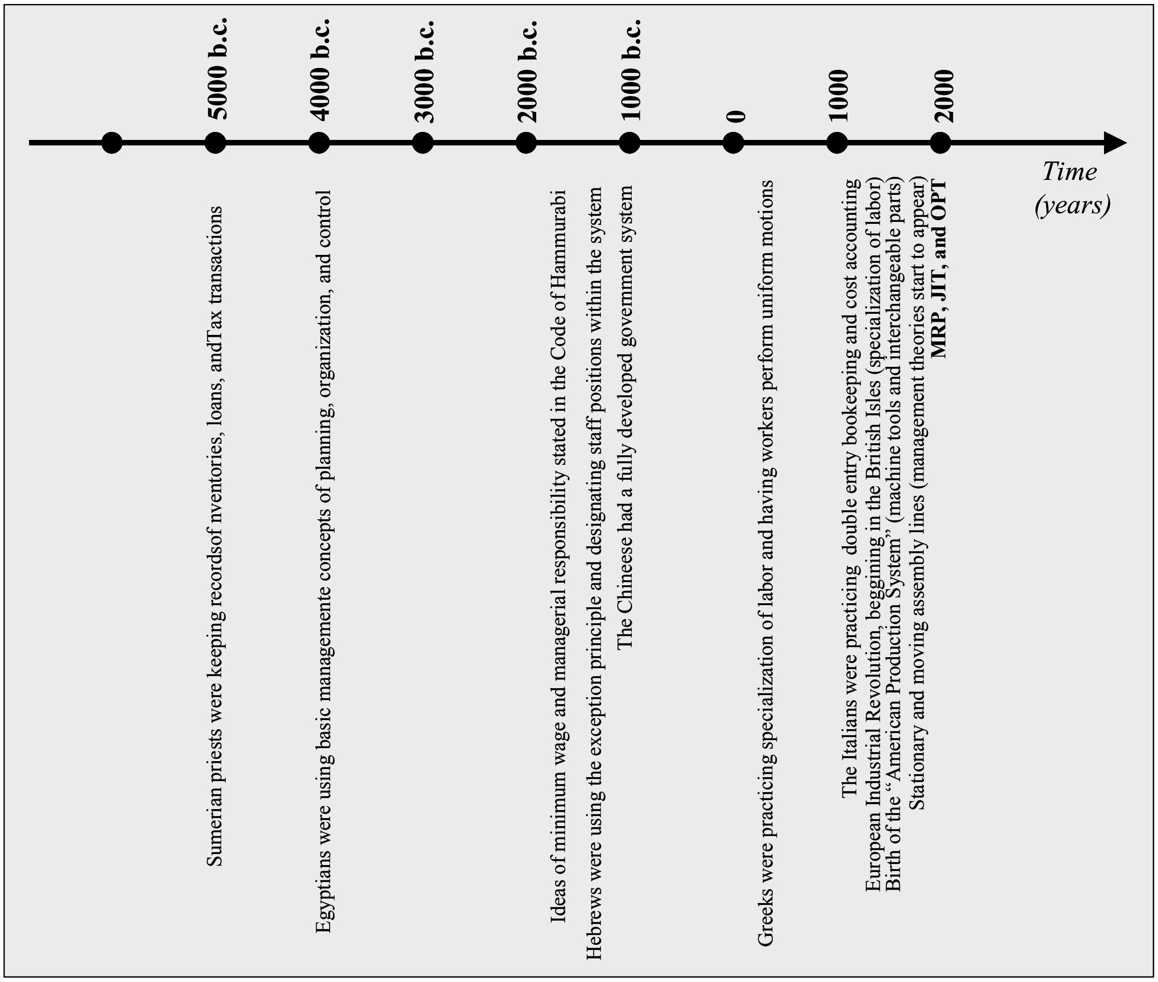


Figure 2.1 *Key Milestones in the Evolution of Production Technologies and Theories. Source: adapted from Sipper & Bulfin Jr. (1997 p.2-3)*

The situation tends to be different in manufacturing enterprises. In these cases, the customer "uses" the enterprise indirectly, meaning that there is not necessarily a real-time coordination between producer and consumer. For instance, consider an auto manufacturer such as Ford. In this case an automobile is sold to a customer, who is in fact ultimately purchasing the service of transportation. The generation of the service of transportation and its consumption happens simultaneously, however, the system capable of providing this service is not the manufacturing enterprise but the automobile itself. In this case, system A (manufacturing enterprise) uses its physical infrastructure to generate system B (the automobile). However, in order to get the ultimate desired service of getting transported, the customer uses the physical infrastructure of system B directly, and not system A. System A is used indirectly. A practical fundamental consequence of these conditions is that many manufacturing enterprises may choose to maintain inventories of finished or semi-finished products; whereas for many service enterprises this is not an option.

Consequently, whether or not enterprise use is direct or indirect imposes distinct requirements and has major implications on the appropriate structure and production strategies. In any case, however, the generic systems concept is maintained, i.e., inputs are transformed into outputs by means of flows promoted by production processes. The rate at which these flows take place is of particular interest, and brings the discussion to the fundamental PPC challenge, i.e., to match the rate of production to the rate of final consumption.

According to control theory, it is unlikely that these rates will be matched without some conscious intervention in the form of a regulatory mechanism. In fact, this is the single most important reason for the existence of a production management system. **Figure 2.2** illustrates a generic regulatory mechanism and its interaction with production processes and the environment. The regulatory component is a control system capable of interfering in the rate of production. In practice, this is accomplished through management processes (e.g., inventory management) that work in the following manner: (1) indicators of system performance are measured and compared to targets, (2) deviations trigger decisions (according to adopted policies) which trigger actions onto the production process in an attempt to achieve the targets (Scheer, 1991; Sipper & Bulfin Jr., 1997; Vollmann, Berry & Whybark, 1997; Sterman, 2000).

In the context of service provision, many enterprises might basically have no choice other than providing the service at the time of consumer choosing. In the context of manufacturing, however, enterprises may be able to select from a number of flow strategies. These strategies are reflected in the decision-making rules embedded in the regulatory infrastructure. According to **Figure 2.2** this decision-making process usually encompasses three stages.

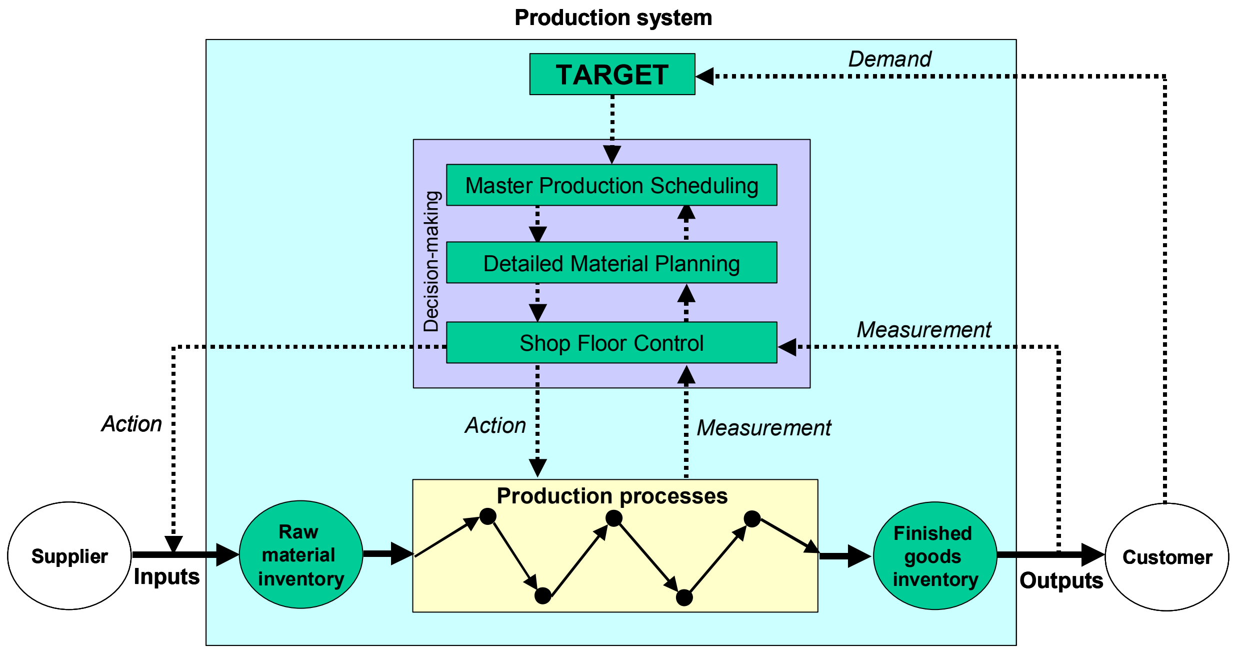


Figure 2.2 Generic Structure of a Production Planning and Control System
Source: Adapted from Scheer, (1991), Sipper & Bulfin Jr. (1997), Sterman (2000), and Vollmann et al. (1997).

STAGE I - Master Production Scheduling (MPS): plans what end items will be produced, in what quantities, and when they will be available. This stage is usually coupled with rough-cut capacity planning. There are three basic MPS strategies.

- **Make to Stock (MTS)**: generates end items in advance of actual demand through the use of forecasting techniques; demand is fulfilled directly from end items inventory; inventory levels are usually high; product variety is low due to physical infeasibility; customer lead time on end items is short.

- **Make to Order (MTO)**: items are produced only as needed according to orders; generally, there are no inventory of end items; product variety may be high; however, customer lead times can be very long depending on the characteristics of the product.
- **Assembly to Order (ATO)**: end items (usually the final stages only) are assembled according to a customer signal (and lower level items are produced in advance of actual demand to an inventory according to a forecast); high variety of end products (high inventory of lower level items); customer lead time usually not as short as in MTS but shorter than the manufacturing lead time.

STAGE II - Detailed Material Planning (DMP): specifies what lower-level items to make, where and when in order to satisfy the MPS. This stage develops and uses detailed capacity plans. Detailed material plans for any of the MPS strategies can also be accomplished in several ways. There are two common planning alternatives depending on the production process's characteristics (Vollmann et al., 1997; Shewchuk, 2003).

- **Time-phased**: uses time-phased records indicating what quantities of what items are due when; typically used with material requirements planning (MRP) approaches; production is usually based on batch (job shop) manufacturing; and materials are also purchased in batch orders.
- **Rate-based:** specified items are produced at specified rates; rates are established for parent items; examples of firms using rate-based planning include repetitive manufacturing, assembly lines, just-in-time (JIT) and other flow systems (see **Table 2.1**).

Table 2.1 Features of Detailed Material Planning Approaches.
Source: Vollmann et al.(1997)

Basis for planning and Control	**Time-Phased**	**Rate-Based**
Control point	Shop/purchase orders	Planning bills
Control unit	Batches	Kanbans
Product level	Material explosion of time-phased net requirements for product components	Material explosion of rate-based requirements for product components
Material planning features	**Time-Phased**	**Rate-Based**
Fixed schedules	No	Yes
Use of WIP to aid planning	High	Low
Updating	Daily/weekly	Weekly/monthly
Inventory netting	Performed	None
Lead-time offsetting	Performed	None
Lot sizing	Performed	None
Safety stock/safety lead-time	Considered	Not considered
Container size	Not considered	Considered
Bill of material	Many levels	Single level

STAGE III - Shop Floor Control (SFC): specify and implement control actions that regulate how the manufacturing facility is run in order to realize the detailed material plans. The three main SFC issues are: scheduling and sequencing, use of alternative routings, and lot splitting (Shewchuk, 2003). In accordance to the two basic detailed material planning options, the two basic approaches for SFC are (Vollmann et al., 1997): (1) time-phased material requirements planning (MRP), and (2) rate-based just-in-time (JIT). MRP is also popularly known as *push,* and JIT as *pull* (see **Table 2.2**).

Table 2.2 *Features of Shop-Floor System Approaches.*
Source: Vollmann (1997)

Basis for planning and Control	**MRP**	**JIT**
Control basis	Work center capacity utilization	Overall product flow times
Unit of control	Shop orders	Kanban cards or containers
Product level	Individual operations scheduled at each work center	Production on an as-required basis to replenish downstream stocks that support end item requirements
Shop floor features	**MRP**	**JIT**
Control of material flow	Work center dispatching rules	Initiated by downstream kanban cards
Sequencing procedure	Due-date oriented dispatching rule	Not an issue
Order tracking	Shop-floor transactions by operation and stocking point	None (paperless system)
Monitoring and feedback	Input/output and shop load reports	Focus on overall result
Order completion	Shop order close-out in stock room	None
Achieving delivery reliability	Batch order status reports	Through flow of material
Lot size	Large	Small
Work-in-process and safety stock	Large	Negligible

2.1.1 Push, Pull, and Bottleneck Systems

Despite disagreement regarding their meaning, the terms p*ush* and *pull* have found widespread use in the literature as a way to distinguish, respectively, MRP-based and JIT-based shop-floor systems. These terms are very useful in providing a generic, simplified, and intuitive understanding of the basic mechanisms behind these alternative paradigms for the control of material flow in manufacturing enterprises.

A more technically accurate discussion, however, would uncover certain key apparent inconsistencies. For instance, Vollmann et al. (1997) argues that this terminology has incited debates over whether MRP is a push or a pull system, whether kanbans are part of a pull system when the company is make-to-stock with inventory, or even if JIT is push-based when the need for an item is exploded into raw materials that are then sent through the factory without any kanban type of replenishment, etc. As advocated by these authors, in order to avoid confusion, the distinction should be simple: push systems allow individual work centers to utilize production capacity (i.e., "to keep busy") without being driven by a specific end item schedule. Pull systems do not.

According to Taylor (1999), push control refers to the production of items at times, quantities, and locations required by a given schedule planned in advance of actual demand. In material control, production orders trigger production activity. Material is pushed through the process to assure adherence to the predetermined master production schedule. Resource utilization is generally a key concern. Push type WIP is used in MRP. Production capacity is typically considered infinite in the generation of the MPS. Consequently there is normally no pre-established basis for WIP inventory control other than the checks and balance procedures of the MRP system itself and, naturally, the physical storage capacity of the workstations buffers. As a consequence, advancement of additional WIP inventory takes place regardless of the current level, which allows inventory to build prior to areas where production capacity is insufficient to handle the load.

Pull control on the other hand refers to the production of items only as demanded for use. In material control terms, this means that material is not moved until a signal is received, i.e., the withdrawal of inventory takes place only as demanded by the customer(s) operations. Pull type

WIP is used within JIT and two conditions are required: (1) output of a product must be level for a reasonably long-time horizon, and (2) mixed-model final assembly must be practiced (Taylor, 1999).

Vollmann et al (1997) argue that MRP was only possible with the advent of high-speed random access computing and point out the fact that not all PPC frontiers are being extended by new technology. For instance, the JIT approach usually drives simplicity and in many cases less computerization. Nevertheless, despite the level of technology required and the terminology employed, MRP-based and JIT-based systems are regarded as two fundamental alternatives for material flow control. Less agreement is found on whether or not bottleneck systems should be considered a third basic option. Considering that bottleneck systems are indeed a third basic option, the excerpt presented next from Sipper & Bulfin (1997 p.531) illustrates its origins:

> *"There are three major approaches to production planning and control: push systems, pull systems, and bottleneck systems. Ironically they all started as production tools and later evolved into PPC systems. The forerunner of push systems was a tool called material requirements planning (MRP), developed in 1974 by Joseph Orlicky of IBM. The forerunner of pull systems was the kanban system introduced in the late 1960s at Toyota by Taichi Ohno. The origin of bottleneck systems can be attributed to Eli Goldratt, an Israeli physicist, who introduced Optimized Production Technology (OPT) in the mid 1970s. (...) Push systems originated in the United States, pull systems in Japan, and bottleneck systems in Israel, a three-continent integration."*

According to Taylor (1999), the bottleneck system is a hybrid pull-push inventory control method used within the Theory of Constraints (TOC) and based on "buffer" management. Buffer management is achieved through what is referred to as drum-buffer-rope. The name comes from the bottleneck defining the schedule (i.e., the drum), "pull" scheduling in nonbottleneck operations (the rope), and buffers at both the bottleneck and finished goods (but not at non-bottlenecks). The idea is to move material as quickly as possible through non-bottleneck work centers until it reaches the bottleneck. At the bottleneck, work is scheduled for maximum efficiency (e.g., large batches), thereafter work again moves at maximum speed to finished goods. What this means in terms of lot sizing is very small transfer batches to and from the bottleneck, with a large process batch at the bottleneck. In fact, JIT operating conditions are used everywhere except at the bottleneck (Vollmann et al., 1997). According to Taylor (1999), the main force behind TOC is the desire for continuous improvement. The improvement effort is based on Goldratt's five steps to constraint management:

1. Identify the system's constraints;
2. Decide how to exploit the system's constraints;
3. Subordinate everything else to the above decision;
4. Elevate the system's constraints; and
5. If, in the previous steps, a constraint has been broken, go back to step one, but do not allow inertia to become the system's constraint.

The ability to identify constraints is central in TOC. In many systems, however, this can be a significant challenge. Fung (1999) presents a detailed discussion concerning the identification of constraints in bottleneck systems, alerting that the bottleneck does not generally coincide with where the greatest congestion in flow is observed.

2.1.2 Factors Influencing the Choice of Candidate PPC Designs

As the overall discussion just presented illustrates, it is widely accepted that any well-designed enterprise system includes a control mechanism through which management decides which aspects of system performance are to be measured and how these measurements are to be used to drive the system towards its purpose. However, it is argued that little is known about the best ways to design such control mechanisms for typical business processes (Powell et al., 2001).

Push and pull are two basic paradigms for production planning and control. The fundamental difference is that push systems initiate production in anticipation of future demand, whereas pull systems initiate production as a reaction to present demand. More specifically, *push* systems control throughput (by controlling work order release) and measure WIP (work in process). On the other hand, *pull* systems control WIP and measure throughput (Sipper & Bulfin Jr., 1997).

During the enterprise design effort, the decision of which PPC architecture to adopt may not be a trivial task; especially when aware that this decision is likely to have a profound impact on the behavior of the total system. Nevertheless, the literature seems to lack explicit quantitative studies to support this design decision at a strategic level (Fowler, 1999).

A number of studies have been conducted focusing on the comparison of push, pull, and push-pull systems from very specific points of view. For instance, Gianesi (1998) argues that, among these three main approaches to production planning and control, MRP is the most suitable for dealing with medium and long term planning, regardless of the type of production process. Taylor (1999) suggests that, from an inventory holding cost point of view, the hybrid push-pull systems present the best performance, followed by pull systems, and then by push systems. Wang and Xu (1997) recommend hybrid push-pull strategy as the best option for mass product manufacturing systems from a combined inventory cost and shortage probability point of view. In this case, the suggested approach is to embed JIT in an MRP system, i.e., to control material input by a push policy and have the remainder of the process work in a pull-controlled way using kanbans.

The complexity involved in this kind of decision arises from the fact that usually there is not a single determinant factor to drive the selection of the preferred design alternative. In fact, this design decision even for simple cases requires the joint consideration of many factors descriptive of the environment as well as the requirements under which the system needs to operate. Razmi, Rahnejat, and Khan (1998) developed a three-dimensional model to support the decision of choosing among push, pull, and push-pull (see **Figure 2.3**). The dimensions represent three important classes of design factors: (1) supplier and lead-time reliability, (2) serviceability, commitment, and costs, and (3) demand fluctuation. The authors denominate it a three-dimensional state-space model. It encompasses many possible environments, providing guidelines for the choice of approach.

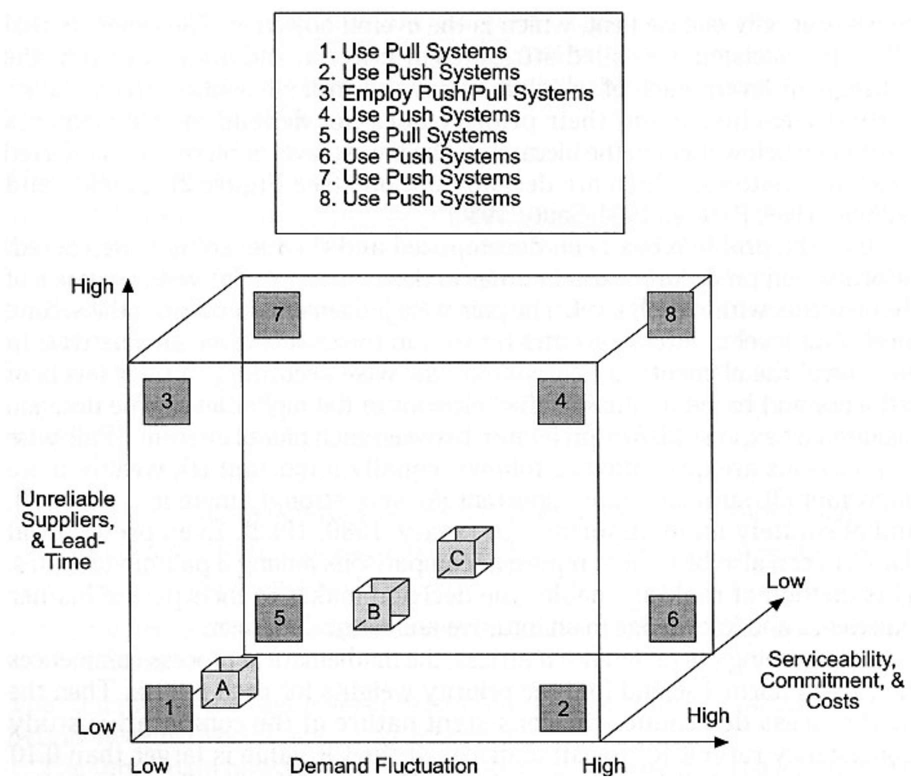


Figure 2.3 *Push versus Pull versus Push/Pull Systems.*
Source: taken from Razmi et al. (1998)

These various views illustrate the complexity involved in the design of the PPC component of high performing enterprises. In summary, there are simply too many possible design configurations, whereas the terms push, pull, and push-pull provide only a very high level indication of the specific system structure in question (Grosfeld-Nir, Magazine & Vanberkel, 2000). The common conclusion shared by many respected experts in the field is that there is not a single best solution that fits the generic case. Each one of the main pre-conceived design paradigms presented has many strengths and limitations. Practical solutions are likely to mix existing concepts and eventually create new ones. The specific requirements at the situation at hand always need to be carefully analyzed so that an informed decision can be made (Sipper & Bulfin Jr., 1997; Vollmann et al., 1997).

2.1.3 Lean Production and the Kanban System

The enterprise design and management approach known as lean production (LP) has its roots in the Toyota Production System and makes extensive use of the pull mode of production control (Womak & Jones, 1996). In recent years it has been popularized by a modeling technique promoted by the Lean Enterprise Institute, denominated *value stream mapping* (Rother & Shook, 1999; LEI, 2003).

LP encompasses a comprehensive set of tools as well as a philosophy. It recognizes the complexity and unique requirements of modern operations and provides an approach geared towards establishing flow, adding value, and eliminating waste. Waste is defined as activities that absorb resources but creates no value to the ultimate client. Overproduction, here meaning producing more, sooner or faster than is required by the customer, is considered the most significant source of waste because, in addition to excess inventory, overproduction is believed to cause all other types of waste. As such, avoiding overproduction is central to LP (Womak & Jones, 1996).

Such a waste reduction mindset combined with concepts of flow and pull control tend to simplify production structures and utilize low cost technology such as the one adopted in most kanban systems. In discrete manufacturing contexts, for instance, production is usually organized according to product families and ideally *one-piece-flow* of finished and semi-finished products throughout the shop floor is implemented; at a rate that matches the average customer purchase rate. In the parts of the system where production cannot flow one piece at a time, work in process inventories (WIP) will naturally accumulate. In such cases *pull* control usually involving supermarkets and kanban systems should be utilized. And, as a last option, if *pull* control cannot be utilized, *push* control should be used (Rother & Shook, 1999).

It needs to be emphasized, however, that achieving *one-piece-flow* requires the alignment of many conditions. Due to physical constraints, in some scenarios it might be considered practically impossible. Consequently, a considerable fraction of the systems classified as *lean* are in fact pull-based systems utilizing kanbans as their main WIP control mechanism.

Pull systems under the rubric of LP have been reported to account for major improvements in enterprise operations all over the world. From a production planning and control point of view, some key advantages of this system include

- shorter lead times in comparison to push systems, which add flexibility to the system to respond to changes in demand (Sipper & Bulfin Jr., 1997);
- reduced inventory in comparison to push systems (Womak & Jones, 1996); and consequent ability to identify problems earlier (Rother & Shook, 1999);

- easy utilization; simple control mechanism (kanban system) that is easily implemented (Fowler, 1999); and
- low cost of supporting technology (Sipper & Bulfin Jr., 1997; Fowler, 1999).

However, the practice of lean production has also received criticism. Among the strongest arguments against the use of this approach are the ones concerning the "blind" adoption of lean techniques upon reliance on its supposed universality and without questioning the adherence to the situation at hand (Goranson, 1999; Cooney, 2002). Considering that LP rests so strongly upon the pull approach (i.e., JIT) to promote control of material flow, some significant weaknesses have been identified.

- It requires a relatively stable product mix and assumes that set-up times are short at every work center (Sipper & Bulfin Jr., 1997).
- Pull systems usually cannot perform lot tracking, i.e., they normally cannot peg lots to specific customers (Sipper & Bulfin Jr., 1997).
- It requires stable demand to operate well and maintain low inventory levels, usually not recognizing well future events (Sipper & Bulfin Jr., 1997) and therefore having very limited flexibility to short-term changes in demand (Corrêa & Gianesi, 1996).

2.1.3.1 Terminology and Principles

The pervasiveness and complexity associated with these concepts makes it difficult to clearly generalize, classify, and distinguish among strengths and weaknesses associated with the lean paradigm. Nevertheless, there are some very specific design guidelines associated with what is nowadays referred to as lean production.

The term *lean production* was coined by a research group at MIT to refer to a set of production system design and operation guidelines with roots in Eiji Toyoda and Taiichi Ohno's Toyota Production System (TPS) (Nightingale, 2002). It was then used in the early 1990's by Womak and Jones (1996) in their book entitled *The Machine That Changed the World,* in reference to the potential benefits of the adoption of this enterprise design and operation approach pioneered by Toyota after World War II. A central contribution of this book was the results of a comprehensive benchmarking study confirming the strength of the LP concepts and practices.

In the following years, another book was put together in order to propose a method to achieve LP. Such a method was synthesized out of past experience from across the globe in successful lean production implementations. This was a key piece of information still missing from previous related work and has since then been referred to as *lean thinking*. Womak & Jones (1996) concluded that *lean thinking* could be summarized in five major guidelines or principles. A summary of each one is provided below. It is important to note that the principles are to be applied in the order presented.

PRINCIPLE I: Precisely Specify Value by Specific Product

Clearly specifying value is central in *lean thinking.* The idea is that the ultimate customer is the only one who can define value. Value has meaning when described in terms of a specific product (good or service) that attends a specific customer need at a specific time. After all, providing value as defined by the ultimate customer is the reason for existence of the entire production system.

PRINCIPLE II: Identify the Value Stream for Each Product

Once value has been defined, the value stream can be identified. A value stream is defined as the set of all specific actions required to bring a specific product through the three critical sets of steps of any business: (1) the *problem-solving task,* i.e., the product development process, (2) the *information management task,* encompassing order taking through detailed production scheduling and delivery, and (3) the *physical transformation task,* i.e., the actual production process, from raw materials to the hands of the customer.

PRINCIPLE III: Make Value Flow Without Interruptions

Given that value has been precisely specified and the value stream for a specific product or product family fully identified (and activities that promote waste as much as possible eliminated) it is time to concentrate on the flows across the system. The idea is to avoid accumulations (e.g., inventory) along the processes and let value be created without interruption. In contrast to the batch-and-queue mode of thinking, promoting flow usually requires breaking organizational paradigms, concentrating on value-added processes and change over challenges from one product to the next in order to systematically reduce the size of production lots. The idea is that the smaller the lot sizes, the closer the system is to producing strictly the necessary and therefore not generating waste.

PRINCIPLE IV: Let the Customer Pull Value From the Producer

As flow is established across the system, the objective is to let the customer pull the product from the producer and not the producer to push the product to the customer. Ideally, production is to be pulled "one piece at a time" across the entire system. If this is not possible and accumulation is required in one or more points than appropriate kanban like mechanisms should be put in place in order to trigger the use of resources only when necessary and not allow inventory accumulations to exceed certain levels. This practice can have a tremendous simplifying impact in the operation of certain systems; ultimately meaning that producing to a forecast is not necessary and that production schedules can be sent to a single point in the process.

PRINCIPLE V: Pursue Perfection

This last principle refers to the need for a continuous effort towards improving the system drawing from the synergistic effects of the previous four steps.

2.1.3.2 The Kanban System: a Core Component of Pull Production

The literal translation of the Japanese word *kanban* is visible notation or visible plaque. The term is also commonly known as plaque or card (Resende & Sacomano, 1997). A kanban system enables the communication from customer to producer to resume production. It is a key component that enables the ability to "pull" in many JIT production scenarios. The number of kanbans in use at each decision-making point in the process matters because each *kanban* corresponds to a specific inventory quantity. Consequently, the kanban system can be interpreted as a management system that works as a regulator of inventory levels. This means that in a strictly kanban controlled pull production there is limit for the maximum amount of inventory levels at any point in time.

There are three basic types of kanban systems (Rentes, 2002): (1) signal kanban, (2) one-kanban system, and (3) two-kanban system. **Figure 2.4** illustrates the two-kanban case. Sipper and Bulfin Jr. (1997 p.546) provide a brief description of this example.

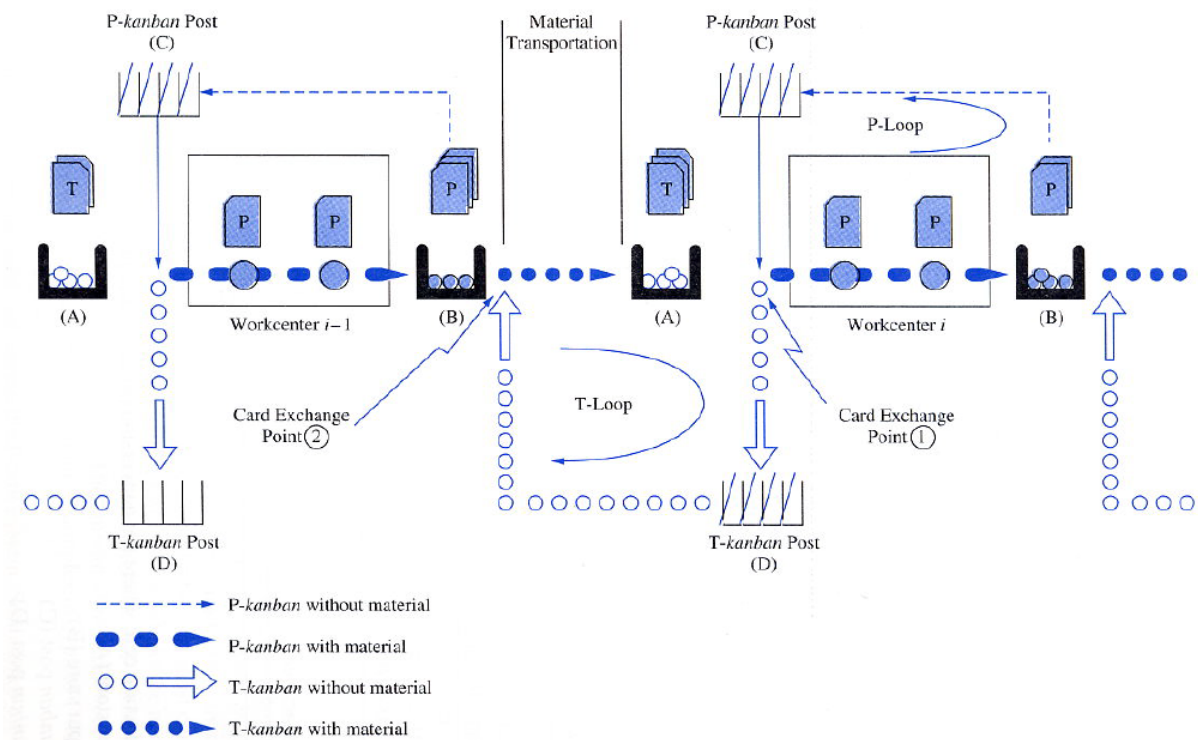


Figure 2.4 *The Kanban System.*
Source: taken from Sipper & Bulfin (1997 p.548)

"Upstream workcenter (i-1) supplies downstream workcenter i. Each workcenter has five components: production cell (where the conversion process take place), input store (A), output store (B), P-kanban post (C), and T-kanban post (D).

The system has two control loops, a P-loop to control cell operation and a T-loop to control material transfer between workcenters. Parts are stored in containers. Each container holds a fixed amount of product that a P-kanban authorizes to produce, or a T-kanban authorizes to move. Each container in the input store (A) has a T-kanban attached. Similarly, each container in the output store (B) has a P-kanban attached. To understand how the system operates, we discuss each loop separately.

P-loop. *When a predetermined number (batch) of P-kanbans is accumulated at the P-kanban post (C) of workcenter i, it signals workcenter i to produce a* *batch. P-kanbans are removed from the post to the card exchange point (1) at the input store (A). There, the T-kanban is removed from each container and replaced by a P-kanban. The T-kanbans are replaced in the T-kanban post (D). The number of kanbans in this exchange is equal to the number of P-kanbans on the post. Production starts and each container has its P-kanban attached. Upon completion, the finished batch is placed in output store (B), its P-kanban is detached and again placed on the P-kanban post (C). The P-kanban post makes the kanbans visible and shows the queue of work to be performed in the cell.*

***T-loop.** When a predetermined number of T-kanbans is accumulated, they are removed from the T-kanban post (D) of workcenter i and taken to the card exchange point (2) of workcenter (i-1). The P-kanbans are removed from each box and replaced by T-kanbans. The P-kanbans are put on the P-kanban post of workcenter (i-1) and the containers with a T-kanban are transported to the input store (A) of workcenter i. The quantity trigger for T-kanban removal is sometimes replaced by a time trigger where T-kanban pickup is performed at fixed time intervals.*

(...) There are three major guidelines for kanban systems: there is no material container in the system without a kanban attached to it, only a P-kanban authorizes production, and only a T-kanban authorizes transportation. These guidelines force all workcenters to be nearly synchronized."

There are a number of methods to determine the number of kanbans needed in a JIT system. The original method was developed by Toyota and is still largely used nowadays (Sipper & Bulfin Jr., 1997 p.552). It considers the following variables:

n : Number of P and T kanban sets for a given part;
D : Demand per unit time, usually the leveled daily demand;
L : Average lead-time for the kanban;
t_p : Average processing time per container;
t_w : Average waiting during the production process plus transportation time per container;
C : Container capacity, in units of products;
α : A safety coefficient (usually not over 10 percent).

Lead-time is taken as the summation of actual processing time plus average waiting and transportation times:

$$L = t_p + t_w \tag{2.1}$$

And the number of kanban sets is calculated based on the average number of units (and consequently containers and kanbans) required to fulfill demand during the lead time:

$$n = \frac{DL(1+\alpha)}{C} \tag{2.2}$$

2.2. Flow Control Policies under Demand Instability

There seems to be a consensus in the literature regarding the vulnerability of pull-based systems to unstable internal and external demand and supplier conditions. The following excerpts reinforce this perception.

> *"Many companies are interested in implementing just-in-time (JIT) manufacturing philosophies in response to increased competitive pressures on manufacturing. At the shop floor level, one application of JIT is through the introduction of Kanbans (or cards) so as to control the in-process inventory. Traditionally, it has been argued that Kanban systems work well when the shop floor environment is fairly stable (Moeeni, Sanchez & Vakharia, 1997 p.2821)".*

> *"A kanban system is not for everybody. It works best when flow is uniform and the product mix is highly stable (Sipper & Bulfin Jr., 1997 p.550)".*

> *"Since JIT was only meant to operate in a deterministic environment, its performance is seriously affected by variations in processing times and demand (...) the JIT was designed for a constant processing time and smooth and stable demand environment, hence its performance is optimum in that environment (...) (Gupta & Al-Turki, 1997 p.133)".*

> *"The kanban system has no adaptability for such sudden and large variations in demand (Monden, 1981 p.46)".*

Fowler (1999) suggests that familiar concepts such as JIT (i.e., *pull*) are actually special cases of generic feedback control principles while MRP (i.e., *push*) is a classical example of feedforward control. He argues that each of these control modes has its strengths and limitations and that the complexities of modern operations often requires combinations of both. Other authors have reinforced the notion that pull-based production is suitable for environments where demand is relatively stable (Sipper & Bulfin Jr., 1997; Mason-Jones, Naylor & Towill, 2000; Cooney, 2002). In fact, Fowler (1999) argues that it might be impossible to engineer a production system capable of producing simultaneously responsive yet stable behavior using pull (i.e., feedback) control.

In essence, these observations emphasize that there is a need for a minimum level of stability if a pull system is to perform well. However the precise limits of demand stability suitable for this mode of operation have not been sufficiently addressed in the literature. In general, there is not a single solution to this problem. It is known, however, that various practices are utilized, often in an ad-hoc manner, in an attempt to minimize the undesirable effects of supply and demand instability. Some of these key practices, which may be used in isolation or in creative combinations, are summarized here.

2.2.1 Increase Production Capacity through Overtime Work

One of the most common and well-known practices is to utilize overtime work as a means to increase production capacity in the short term. While the supply of inventory lasts, this practice works as an expansion of the base production capacity enabling demand peaks to be absorbed. As observed by Rees, Philipoom, Taylor, and Huang (1987 p.201):

> *"For JIT to work, demand must be fairly constant. Minor fluctuations in the demand are handled by adjusting the length of the workday. Thus, even if the daily demand fluctuates, the hourly demand rate should stay constant."*

This can be a costly approach considering that overtime work hours are likely to be considerably more expensive than normal work hours. As such, firms tend to quit this practice and acquire permanent resources should the frequency of demand peaks increase enough to justify it economically.

2.2.2 Recover Unused Capacity through "Bank of Hours"

In this case, total production capacity is fixed over a given planning horizon. However the fraction of the worker capacity that do not get consumed over time is saved for future use by means of a bank of hours, which is an accumulation of the capacity not utilized over time. Basically, workers are allowed to leave the shop floor once they have met the daily production target and their remaining (i.e., not utilized) capacity is recorded. Therefore, at any given time, the firm could have a quantity greater or equal to zero worth in worker hours that can be utilized in addition to the base worker capacity.

2.2.3 Maintain Stock of Bestseller Products to Absorb Demand Shocks

Some firms are willing to pay for maintaining finish product inventories for bestsellers. These inventories act as "shock absorbers" at the occurrence of demand peaks, which enables the system to keep producing the other lower demand products while unusually high demand orders for bestsellers are also being fulfilled. At times when demand levels go down, capacity is used to replenish bestsellers inventories. This practice is aligned with the MTS production strategy and is well described in the PPC literature.

2.2.4 Maintain Delivery Time Buffers

Another practice is to have customers agree on a certain delivery lead-time longer than what actually is necessary. This extra time acts as a safety component, enabling the producer to better handle unexpected large orders and manage supplier delivery problems without undermining customer satisfaction regarding on-time delivery (Shewchuk, 2003). This practice is aligned with the concept of safety lead-time. However it is of limited use in scenarios where at least one strong competitor is present and time-based competition is critical.

2.2.5 Manipulate the Frequency of Kanban Transfers

Sometimes kanbans may wait at the kanban posts longer than desired because of the adoption of a certain fixed review frequency. This can cause interruption of production and the consequent loss of production capacity. If the fixed interval for kanban revision is decreased it is possible that capacity gets better utilized. Thus, in this case, the number of kanbans in the system remains fixed but the frequency of kanban transfers is increased. Monden (1981 p.46) describes the adoption of this practice at Toyota:

> *"The number of Kanbans tends to be fixed despite variations in demand. Toyota's experience shows that a 10-30% variation in demand can be handled by changing only the frequency of Kanban transfers without revising the number of Kanbans."*

2.2.6 Manipulate the Number of Kanbans in the System During Production Cycles

Another practice is to change the number of kanbans in the system during production cycles. It is argued hypothetically (Rees et al., 1987; Gupta & Al-Turki, 1997; Gupta, Al-Turki & Perry, 1999) that this practice – by dynamically manipulating the level of WIP in the system – can promote better capacity utilization, as work centers tend to *starve* and *block* less often. As such, the timing of the insertion/removal of cards tends to influence the ability of the system to respond to instabilities in demand and processing time. More sophisticated variations of this practice can also be used as advanced kanban control mechanisms, and have been referred to as "flexible kanban system" or FKS; in contrast to the "traditional kanban system" or TKS where the number of kanbans remains fixed.

Flexible kanban systems (FKS) are a recent attempt to increase delivery robustness of pull-based systems in situations of demand and/or supply instability. Although the FKS literature does not specifically contextualize the discussion in terms of value stream applications, this concept has served as original insight for the direction of this research. A brief review is provided in this section.

As described by Monden (1981) the original Toyota Production System guidelines did not advocate changes in the number of kanbans during production cycles. In order to cope with changes in demand due to variability around the estimated average, Toyota tended to manipulate the frequency of kanban transfers instead. The type of situation when Toyota would in fact change the number of kanbans in face of large variations in demand is explained in the following excerpt. These situations involve not only changing the number of kanbans but also imply a possible intervention in other aspects of the systems structure, such as layout rearrangements or alterations in production capacity.

> *"The case of seasonal changes in demand, or the case of an increase or decrease in actual monthly demand over the predetermined load or the preceding month's load. For these cases, the number of Kanbans must be increased or reduced, and at the same time, all the production lines must be rearranged. That is, the cycle-time of each workshop must be recomputed and correspondingly the number of workers in each process must be changed. The kanban system has no adaptability for such sudden and large variations in demand. In order to cope with the bottom and the peak in variation of demand during the year, top management has to make a decision either to level the sales volume for the whole year, or construct a flexible plan for rearranging all the production lines corresponding to seasonal changes during the year (Monden, 1981 p.46)."*

However, as the following excerpts suggest, there is need for better understanding the impact of managing the number of kanbans during (i.e., not only in between) production cycles.

> *"The number of kanbans is generally held fixed during a production cycle. It is, however, well known that supervisors, from time to time, on an ad-hoc basis, increase or decrease the number of kanbans depending on whether the system is experiencing shortages or build up. Even so, no techniques have been reported to systematically manipulate the number of kanbans (...) (Gupta & Al-Turki, 1997 p.134)."*

> *"The Kanban system has been developed by a Japanese automobile manufacturer as an original system of JIT ordering system, and the alternatives such as the constant work in process (CONWIP) system or the concurrent ordering system have been proposed (...). However, in most of the previous literature on JIT ordering systems, stable changes in demand have been assumed, and the influence of unstable changes in demand has never been analyzed. Recently, product life cycles become shorter and shorter due to the diversification of customer needs, and the duration of stationary demand has also shortened. Therefore, not only stable changes in demand, but also unstable changes, should be considered in designing an ordering system (Takahashi & Nakamura, 2002a p.702)."*

It is argued theoretically by some experts that a FKS would be able to improve performance of the system given a certain base configuration established in each redesign and implementation cycle referred to by Monden's excerpt. Considering that research in this area is not abundant and that no reports of actual usage of FKS in practice have been found in the literature, some questions remain open at this point.

The term *flexible* in flexible kanban system is suitable because it refers to one specific system structure theoretically capable of better absorbing stable and unstable changes in processing times and demand. A more advanced discussion regarding the extension of this concept in terms of *agility* characteristics could be also be appropriate, where agility is interpreted as the ability to create and switch among alternative system structures according to the current needs. Vernadat (1996) offers both the definitions of flexibility and agility for systems in general but specifically illustrates them in the enterprise case.

For illustration purposes, consider the case of a sports utility vehicle. Agility in this example might mean the ability to quickly create and switch among alternative suspension systems, perhaps one for highway conditions and another one for off-road conditions; whereas flexibility would refer to the intrinsic ability of each particular suspension system to keep the vehicle stable under adverse conditions.

Rees et al. (1987) proposed an algorithm to adjust the number of kanbans using estimated values of lead-time. Gupta and Al-Turki (1997) proposed an algorithm to adjust the number of kanbans during production cycles taking into account stochastic processing times and variable demand. This algorithm was later enhanced to include the consideration of preventive maintenance interruptions and the effect of sudden material handling breakdown (Gupta & Al-Turki, 1998a; Gupta & Al-Turki, 1998b). A systematic methodology to manipulate the number of kanbans was then presented by Gupta et al (1999) and referred to as "flexible kanban system." Moore and Gupta (1999) introduced the use of stochastic colored Petri nets to model traditional and flexible kanban systems. Takahashi and Nakamura (1999) proposed a reactive JIT ordering system for unstable changes in demand via manipulation of buffer size. This concept was then expanded to multi-stage JIT production, inventory, and transportation systems (Takahashi & Nakamura, 2000b; 2000a) and then further enhanced into a decentralized concept (Takahashi & Nakamura, 2002a; Takahashi & Nakamura, 2002b). Tardif and Maaseidvaag (2001) also provide suggestions for an adaptive approach to controlling kanban systems. Although these studies all focus on JIT production, considerations are provided for generic pull scenarios and none of them contextualizes the investigation in terms of the value stream concept.

2.3 Engineering Robustness to Demand Instability in Value Streams

This research focuses on a specific type of pull-based system known as value stream, particularly in accordance with the guidelines advocated by value stream mapping theory prescribed by the Lean Enterprise Institute (Rother & Shook, 1999).

The literature review effort suggests that little quantitative research has been conducted on the topic of value stream delivery robustness under demand instability. A recent text from Art Smalley (2004), also sponsored by the Lean Enterprise Institute, provides related quantitative guidelines for calculating the size of protective inventories in unstable demand conditions. This text also recognizes the need for establishing formal flow control policies to regulate not only the use of these inventories but also to translate customer orders into production schedules in unstable situations.

No previous studies specific focused on value stream characteristics or weaknesses using a system dynamics approach have been found. Gupta and Gupta (1989) present a system dynamics model of a JIT-kanban system. The focus is on the behavior of a single station within a generic production line; key value stream characteristics such as flow control across the system via a single point in the process (i.e., the pacemaker) are not considered. The objective of this simulation study was to determine the relationship of the number of kanbans and container size to the production efficiency under various scenarios.

O'Callaghan (1986) presents a system dynamics model of a three-stage JIT transfer line. The model emphasis is on production scheduling and smoothing. The model is used to examine the response of the production system to small shocks, such as small changes in demand. However, a generic pull system, where customer orders arrive at the stage closest to the customer and propagate upstream via kanbans is assumed. Key value stream components are not identified explicitly.

Despite the lack of literature specifically focused on value stream dynamics, some generic insights from the system dynamics as well as enterprise engineering literature regarding suggestions for how to increase robustness of a system were identified. System robustness can be interpreted as the ability of a system to stay in stable state when subjected to perturbations (Vernadat, 1996). Naturally, the state variables of interest and the degree of stability appropriate for any given situation needs to be defined according to the nature of the case. Recall the example of a sports utility vehicle. One could argue that it is robust (from the point of view of the suspension system) if passengers are able to have a comfortable ride without significant bumps or shakes, even when the vehicle is driven in reasonably bad road conditions.

Disney et al. (1997) explain that robustness is a function of the system structure, i.e., a property arising from the internal arrangement of system components in face of external disturbance. From a material flow control point of view, it is suggested that the addition of feedback loops combined with the reduction of time delays can increase the robustness of the system and lead to better performance. Towill (1996a) confirms this assertion, pointing out that the reduction or elimination of time delays is well known among system dynamicists as a preferred route to achieve better dynamic behavior.

Matson and McFarlane (1999) use the term *responsiveness* to refer to the ability of a production system to achieve its operational goals in the presence of supplier and customer disturbances, where disturbances are those sources of change that occur independently of the system's intentions. Interestingly, it is emphasized that such disturbances are likely to be perceived at the supply and customer interfaces of a production system.

Powell et al. (2001) suggest that, regardless of the demand environment, a control process based on system backlog is generally more robust than the alternatives in the sense that adequate performance is achieved over a broader range of control parameters. Additionally, it is also suggested that proportional control by itself is inadequate to provide effective performance and that differential control is a necessary adjunct.

Given the current emphasis in the today's business arena on competitive factors such as quality, cost, innovation, and timeliness, one particular useful definition of delivery robustness is: the systematic ability of the system to deliver products on time despite undesirable significant variations in demand and supplier rates.

Chapter 3
Research Methodology

3.1. Research Purpose

The long-term intent of this research stream is to develop innovative value stream structures that are better capable of coping with externally as well as internally generated instabilities. The particular purpose for this dissertation is:

To develop alternative flow control policies and evaluate their impact on value stream delivery robustness under external demand instability

3.2. Research Questions

Three critical research questions support and clarify the research purpose:

- **Why do traditional value streams require stability in demand and processing times to perform well?**
- **What types of external demand profiles are appropriate for the operation of a pull-based system?**
- **Under what conditions can alternative flow control systems make a significant beneficial difference in the performance of value streams?**

3.3. Research Objectives

Three research objectives, one related to each research question, have been defined. As explanation of what is meant by each one, key research outcomes supporting the objectives are also identified:

I. **Create a comprehensive explanation of how the behavioral characteristics of traditional value streams emerge as a result of the system structure:** compilation of an initial theory of how the key elements of traditional value streams interact to generate the dynamics observed in the literature (to be accomplished by means of a textual explanation illustrated with causal loops and stocks and flows, as well as appropriate simulation results); identification of high leverage points in the system structure which may serve as a target for structural changes aiming improvement in performance in this research as well as in future work. The pursuit of this objective generated the AS-IS hypothetical model described in **Chapter 4** (sections **4.1** through **4.4**) as well as the AS-IS real model described in **Chapter 5** (sections **5.1** through **5.4**).

II. **Identify and describe modes of demand appropriate for pull-based system operation:** review of potential types of demand profiles (e.g., exponential growth/decay, goal-seeking, overshoot and collapse, oscillation, overshoot and oscillation, etc.); classification of profiles in terms of appropriateness for high performing pull system operation; identify which profiles, if any, can not at all be handled effectively by a pull system. Please refer to **Appendix D** for the results addressing this objective.

III. **Propose and test alternative flow control solutions:** specification of alternative flow control policies; evaluate alternatives under selected modes of demand instability and compare performance against the traditional value stream baseline. The pursuit of this objective generated the TO-BE hypothetical model policies and related structures described in **Chapter 4** (section **4.5**) as well as the TO-BE real model policies and related structures described in **Chapter 5** (section **5.5**).

3.4. Research Process

The definition of research provided by Leedy and Ormrod (2001), serves as a guide for the methodological considerations presented next. Research is considered the systematic process of collecting and analyzing data in order to increase our understanding of the phenomenon about which we are concerned or interested.

The particular research methodology intended for this case is indirect experimentation by means of formal computer simulation models, as directed by a system dynamics modeling approach. Research strategies based on computer modeling are not uncommon. In fact, Carley (1999) argues that computational modeling is increasingly being used to do theory development. In such cases, procedures such as computer simulation, numerical estimation, and emulation models have been used to describe complex systems and generate and test a series of hypotheses about the behavior of these systems under different scenarios.

Objective II is fulfilled through a classification effort based on literature review as well as logical interpretation of possible relationships between demand and production rates under conditions of interest in this study.

Objectives I and **III** are achieved through a parallel effort involving a systematic modeling approach consisting of ten steps as illustrated in **Figure 3.1**. The approach involves dividing the effort into two main phases: (I) identifying and dealing with a generic hypothetical scenario first in order to ensure alignment with basic pull system characteristics; and then (II) applying the findings in the generation and evaluation alternative scenarios for a real manufacturing enterprise.

According to the system dynamics modeling paradigm, the system structures reflected in computer models constitute dynamic hypotheses upon which the research effort relies. The hypotheses are *dynamic* because they must provide an explanation of the dynamics characterizing the problem in terms of the underlying feedback and stock and flow structure of the system. They are *hypotheses* because they are always temporary i.e., subject to revision or abandonment as more learning from the modeling process itself as well as from the real world is achieved (Sterman, 2000 p.95).

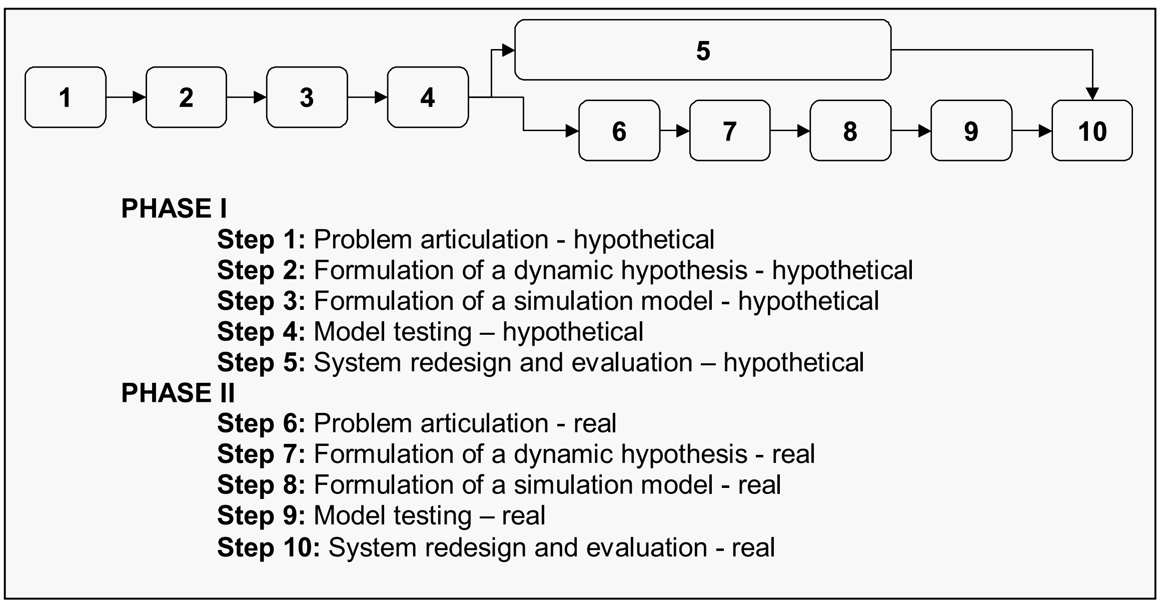


Figure 3.1 *Research Process.*

Some steps in the proposed research process illustrated in Figure 3.1 are to be conducted concurrently. In particular, Step 5, involving the hypothetical conception of alternative flow control systems, is to be carried simultaneously with Steps 6, 7, and 8. The reason is that the learning gained from these early steps in Phase II is expected to provide valuable insights in terms of practical characteristics and limitations in real value streams – thus, potentially enriching the quality of the generic solutions proposed in Step 5. Nevertheless it should be emphasized that the purpose in Step 5 is to identify exploratory solutions capable of improving value stream operation and not necessarily on optimizing performance.

Steps 9 and 10 are to be focused, respectively, on finalizing formal tests of the dynamic hypothesis descriptive of the real system, and on customizing the generic solutions proposed hypothetically for the particular real case at hand. Just like in Step 5, it is expected that Step 10 will also involve the comparison of proposed alternative structures against a base (i.e., current) system structure. These comparisons will provide quantitative estimation of the impact of adopting one or more proposed improvements.

3.5. Research Process Background and Steps

This section contextualizes this research process within a perspective of learning and improvement cycles, as well as provides a detailed description of each step in the process.

3.5.1 System Dynamics and Learning Cycles

System dynamics is suitable for analyzing the behavior of systems by combining theory, methods, and philosophy. It is most useful to understand how policies affect behavior. A system dynamics modeling project starts from a problem to be solved or an undesirable behavior that is to be corrected or avoided. It uses concepts from the field of feedback control to organize information into a computer simulation model (Forrester, 1998). According to Sterman (2001 p.10):

> *"System dynamics is a method to enhance learning in complex systems. Just as an airline uses flight simulators to help pilots learn, system dynamics is, partly, a method for developing management flight simulators (often based on formal mathematical models and computer simulations) to help us learn about dynamic complexity, understand the sources of policy resistance, and design more effective policies. (...) it is grounded in the theory of nonlinear dynamics and feedback control developed in mathematics, physics, and engineering. Because we apply these tools to the behavior of human as well as technical systems, system dynamics draws on cognitive and social psychology, organization theory, economics, and other social sciences."*

3.5.1.1 Policy Resistance

Policies and decisions are conceptually very distinct from one another although, according to Forrester (1975), they are often confused in the literature. Policies are those rules that guide decisions. A policy treats the general case and usually partially defines how specific decisions under that policy are to be made.

Many authors recognize that policy resistance is nothing new and it has long been acknowledged that people seeking to solve a problem often make it worse (Senge et al., 1994; Sterman, 2001). Policies may create unanticipated side effects. As pointed out in Sterman (2000 p.5):

> *"Our attempts to stabilize a system may destabilize it. Our decisions may provoke reactions by others seeking to restore the balance we upset (...) these unexpected dynamics often lead to policy resistance, the tendency for interventions to be delayed, diluted, or defeated by the response of the system to the intervention itself."*

Furthermore, Sterman (2001) argues that policy resistance arises because, despite the amazing potential of the human mind, the complexity of the world overwhelms our understanding. Human mental models are limited, internally inconsistent, and unreliable. Consequently, our ability to understand the unfolding impacts of our decisions is poor. Humans usually take actions that make sense from a short-term and narrow-minded perspective, but due to the imperfect understanding of complexity, some of these very decisions often return to cause problems in the long run.

3.5.1.2 Systems Thinking

In other to overcome such problems, many believe that the solution lies in *systems thinking,* i.e., the conscious effort to understand the system as a whole (Senge et al., 1994; Fowler, 1999; Ritchie-Dunham & Anderson, 2000). With a holistic view of the world, it is believed that we would be able to learn faster and more effectively, identifying the high leverage points in systems, and avoid policy resistance. Ideally, a systemic perspective would enable us to make decisions taking into account the long-term best interests of the system as a whole. The field of system dynamics provides an organizing framework for promoting systems thinking and analyzing how policies and decisions interact (Keough & Doman, 1992; Sterman, 2001).

3.5.1.3 Dynamic Complexity

It is believed that most people think about complexity in terms of the number of components in a system or the number of possibilities one must consider in making a decision (Sterman, 2001). For instance, the problem of optimally scheduling an airline's flight and crews is highly complex, and the complexity in this case lies indeed in finding the best solution out of a very large number of possibilities. Problems of this nature have high levels of what is known as *combinatorial* or *detail complexity.*

However, when dealing with policy resistance it is also important to consider another type of complexity referred to as *dynamic complexity,* i.e., the counterintuitive behavior of complex systems that arises from the interaction among agents over time. This type of complexity is significantly distinct from the previous one, and arises because systems are (Sterman, 2001):

- Constantly changing;
- Tightly coupled;
- Governed by feedback;
- Nonlinear;
- History-dependent;
- Self-organizing;
- Adaptive;
- Characterized by trade-offs;
- Counterintuitive; and
- Policy resistant.

Among the many underlying elements of dynamic complexity, the ones people usually find most problematic are feedback, time delays, stocks and flows, and nonlinearity. A brief discussion of each one of these topics is presented next:

3.5.1.3.1 Feedback

One cause of policy resistance is our tendency to interpret experience as a series of events like "the temperature is too low" or "the inventory is too high". This open-loop view of the world leads to an event-oriented, reactionary approach to problem solving where we assess the state of affairs and compare it to our goals and the gap between the situation we desire and the situation we perceive defines our problem. Without an understanding of the feedback processes that create these outcomes as consequences of our own decisions, we are likely to see new crises as more evidence confirming our view that the world is unpredictable, unpleasant, and uncontrollable – that all we can do is react to events (Fowler, 1999; Sterman, 2000; 2001).

3.5.1.3.2 Time Delays

Time delays between taking a decision and its effects on the state of the system are common and particularly critical. These delays create instability and increase the tendency of systems to oscillate. Consequently, decision makers, by ignoring these delays, often continue to intervene to correct apparent problems or discrepancies between the desired and actual state of the system long after sufficient corrective actions have been taken to restore the system to equilibrium (Towill, 1996a; Mason-Jones & Towill, 1998; Fowler, 1999; Sterman, 2000).

3.5.1.3.3 Stocks and Flows

Stocks and flows, also referred to as the accumulation and dispersal of resources, are vital to the dynamics of complex systems. An enterprise's inventory of final products is increased by the flow of production and decreased by the flow of shipments. The literature suggests that only since the past decade or so has the strategic management community begun to consider the role of stocks and flows explicitly, as the resource-based view of the firm has grown in popularity (Ritchie-Dunham & Anderson, 2000; Sterman, 2001).

3.5.1.3.4 Nonlinearity

Effect is rarely proportional to cause, and what happens locally in a system (i.e., near the current operating point) often does not apply in distant regions (i.e., other states of the system).

Nonlinearity often arises from the basic physics of systems or from the interaction of multiple factors in decision-making (Sterman, 1991; 2000; 2001).

3.5.1.4 Attribution of Error

Research shows that few mental models incorporate any feedback loops. Experiments in causal attribution show that people tend to assume each event has a single cause and often end their search for explanations when the first convincing cause is found (Sterman, 2001). Although a fundamental principle in system thinking emphasizes that the structure of a system guides its behavior, in many cases the tendency to blame people instead of the system they are part of is so strong that psychologists call it the *fundamental attribution error*.

3.5.1.5 Learning in Complex Systems

Increasing the understanding of the true causes of undesirable behavior in a system requires a process of learning (Senge et al., 1994). To learn is defined by the Oxford dictionary (Hornby, 1995) as: "to gain knowledge or skill by study, experience or being thought; to become aware of something through information or observation." True learning implies enhancement in one's mental models and a consequent ability to consider complementary perspectives/explanations of a phenomenon.

Attempting to learn about a system that one is a significant part of can be a very challenging task. When considering the context of enterprise systems, where the human component is not only a user but also its designer, a comment provided by Sterman (2000) is worth repeating here: "We are all passengers on an aircraft we must not only fly but redesign in flight."

Successful approaches to learning about complex systems require: (1) tools to elicit and represent the mental models we hold about the nature of difficult issues; (2) formal methods and simulation approaches to test and improve our mental models, design new policies, and practice new skills; and (3) methods to sharpen scientific reasoning skills, improve group processes, and overcome defensive routines for individuals and teams (Senge et al., 1994; Sterman, 2000; Groesbeck, 2001).

As far back as in the early 1960's Forrester (1961) proposed a management laboratory approach affirming that the information-feedback concepts of system behavior, mathematical models of dynamic interrelationships, and the digital computer to simulate system interactions make experimental industrial systems design possible following the same steps common to other laboratory design approaches. At the time, Forrester specifically proposed an enterprise design approach based on the following steps: (1) identify the goals, (2) describe the current situation, (3) create a mathematical model based on the previous description, (4) simulate the model, (5) interpret the results of the simulation, (6) revise/change the system structure and policies, (7) repeat the experiments for the candidate designs.

Figure 3.2 illustrates current ideas about requirements for successful learning in complex systems – which are in fact strongly aligned with Forrester's original ideas. In particular, it emphasizes the relationship between **real world** and **virtual worlds**, i.e., the connection between reality and formal models and simulations in which decision-makers exercise decision-making skills, conduct experiments and play. And this can be done by means of physical models, role-plays, or computer simulations.

Sterman (2001) supports that the ideal learning process has a double-loop nature, resembles the scientific method, and involves continuous experimentation in both the virtual world and the real world. Feedback from both informs the development of mental models, formal models, and design of experiments for the next iteration. This paradigm is closely related with the concept of cycles of transformation discussed in **Chapter 1**.

As such, the concept of double-loop learning lies at the core of System Dynamics theory, and is also directly related to the ability of enterprise systems to evolve and adapt to new conditions. As illustrated in **Figure 3.2**, virtual worlds, especially by means of computer-aided tools, play a key role in today's practical modeling efforts.

In the context of enterprises, learning by experimentation with both virtual worlds and reality itself also brings the discussion to a very important and fundamentally related point: the duality of system operation vs. system design (Keough & Doman, 1992). The following excerpt emphasizes this point (Sterman, 2000 p.84):

> *"Jay Forrester often asks, Who are the most important people in the safe operation of an aircraft? Most people respond, The pilots. In fact, the most important people are the designers. Skilled, well-trained pilots are critical, but far more important is designing an aircraft that is stable, robust under extreme conditions, and that ordinary pilots can fly safely even when stressed, tired, or in unfamiliar conditions. In the context of social and business systems, managers play both roles. They are pilots, making decisions (who to hire, what prices to set, when to launch the new product) and they are designers, shaping the organizational structures, strategies, and decision rules that influence how decisions are made. The design role is the most important but usually gets the least attention. Too many managers, especially senior managers, spend far too much time acting as pilots – making decisions, taking control from subordinates – rather than creating an organizational structure consistent with their vision and values and which can be managed well by ordinary people.*
>
> *Today, designing a new aircraft is impossible without modeling and simulation. Managers seeking to enhance their organizational design skills, however, continue to design by trial and error, by anecdote, and by imitation of others, though the complexity of their organizations rivals that of an aircraft. Virtual worlds provide an important tool for managers in both the operation and especially the design of their organizations."*

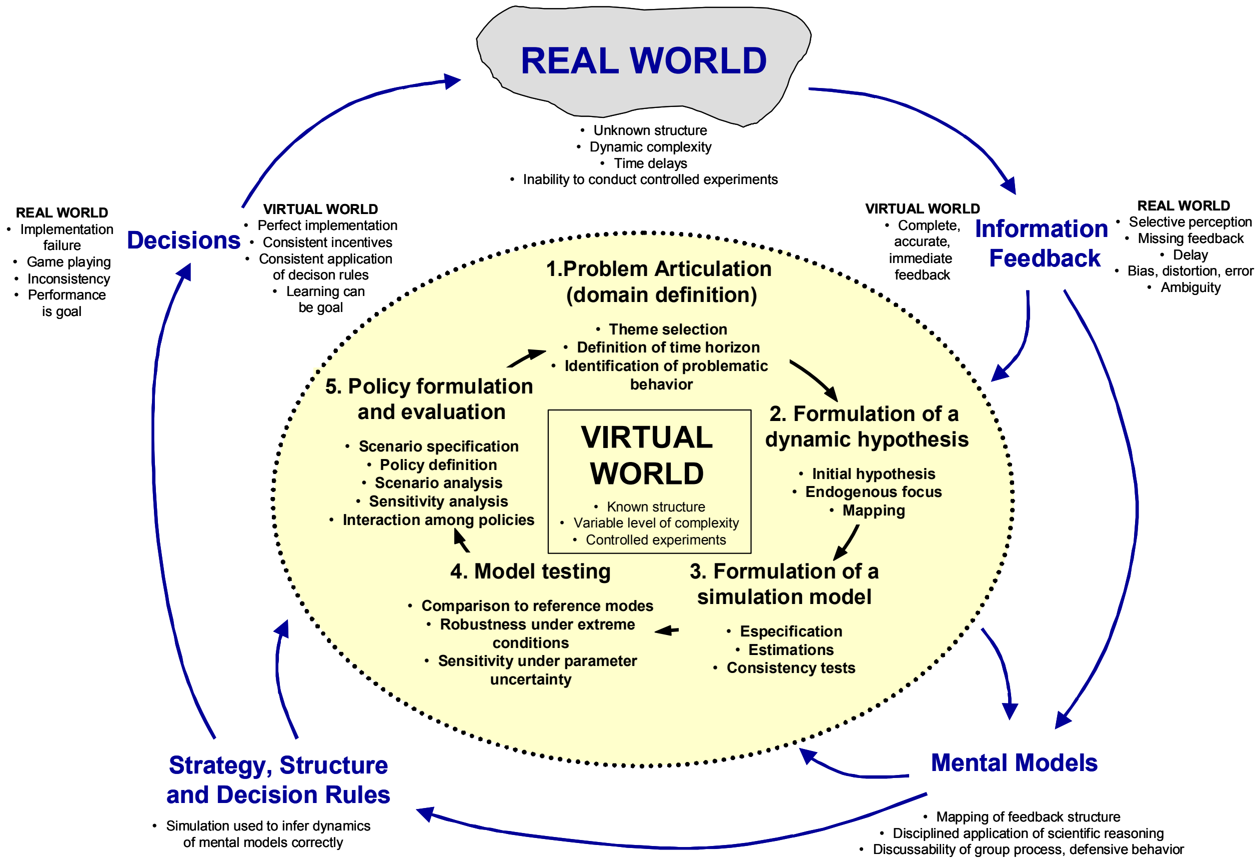


Figure 3.2 *Idealized Learning Process.*
Source: adapted from Sterman (2000)

3.5.2 Description of Research Process Steps

The research approach described in this chapter constitutes a methodology to generate and use models in the virtual world. These steps reflect a logic that represents indirect experimentation, i.e., experimentation with a model of the real system and not with the real system itself (Forrester, 1961). A description of each of the steps is provided in this section. Additional methodological details are discussed in **Chapter 4** as the hypothetical model is presented. More details on system dynamics modeling tools are provided in **Appendix B** (section **B.2**),

3.5.2.1 Problem Articulation

A system dynamics modeling project starts with the identification of a problem or undesirable behavior that needs to be corrected. The problem identification guides the overall effort, therefore, being crucial for the successful accomplishment of the undertaking. A holistic problem definition may require the cooperation between individuals in various fields of expertise. However, considering that complementary points of view are likely to perceive complementary aspects of the problem, reaching an agreement on what actually constitutes the problem might be a challenge in itself.

In the case of this research, there has been great care in the definition of a problem based on what seems to be a shared perception across the industrial and systems engineering literature regarding vulnerabilities of pull-based systems. Nevertheless, this step requires the specification of a theme, key variables and concepts, an appropriate time horizon, as well as a dynamic problem definition by means of behavior over time graphs (BOT). In the hypothetical case, the BOTs will need to reflect an interpretation of the fragmented arguments that have been provided in the literature. In the real case, these graphs will ideally reflect historical performance.

3.5.2.2 Formulation of a Dynamic Hypothesis

Next, an explanation of why the problem arose is created involving the available theories of the causes of the problematic behavior. Initially, a textual description of the causality relationships between key variables is created in order to generate an initial hypothesis. This hypothesis will then provide the initial inputs to a map of the system structure described by means of causal loop diagrams with stocks and flows, as well as other tools as appropriate such as the subsystems diagram, and the policy structure diagram.

As the iterative process of capturing and evaluating the system structure expressed in the map takes place, keeping a boundary chart updated can be helpful in monitoring which variables remain endogenous to the model, which variables are exogenous, and also which potentially relevant variables have been excluded from the model. The goal is to create a description of the problematic system structure that explains important variables as endogenous consequences of the system structure. Therefore, all exogenous variables are candidates to be incorporated endogenously; each one of them needs to be carefully analyzed to make sure important feedbacks do not go trough them.

The collection of elements created in this step reflects a dynamic hypothesis attempting to describe the problem. This hypothesis then needs to be tested. In certain simple situations, perhaps mental tests are appropriate for the purpose at hand. However, even for systems of moderate complexity, there is usually a need for more advanced approaches.

The system structure captured in causal loop diagrams with stocks and flows can be mathematically interpreted as a system of differential equations. Depending on the complexity of this system of equations, it can be solved analytically. Often, analytic solutions are not possible and the model needs to be calculated through numerical integration procedures. Therefore, it is expected that the dynamic hypothesis at this point will be converted into a computer simulation model so that it can be efficiently evaluated.

3.5.2.3 Formulation of a Simulation Model

A computer simulation model serves two purposes. One purpose is to enable the efficient numerical calculation of models of considerable size and complexity. The other is to provide a way to test the dynamic hypothesis without having to rely on the feedback obtained from the real world; i.e., without simulation even the best conceptual models can only be tested and improved based on the learning feedback through the real world (Sterman, 2000) – which could prove too time consuming, too costly, too complex, or perhaps considered practically impossible.

The creation of a simulation model involves two basic parts: (1) the mathematical specification of structure and policies, and (2) the estimation of parameters, initial conditions and behavioral relationships. This mathematical specification can be partially derived directly from the stock and flow structure, where the stocks are integrals of the difference between inflows and outflows, and the flows themselves are derivatives.

However, to complete this mathematical specification, the analytical relationship among the remaining variables (including the description of policies) has to be defined. This is necessary because the causal loop diagrams only provide identification of causality in terms of whether it is positive or negative. Analytical relationships between variables reflect assumptions regarding physical properties of the system as well as decision rules used by agents to regulate flows. Uncovering and specifying decision rules used by human agents can be expected to be a significant challenge if the degree of informality and lack of standardization involved is high.

Once the mathematical equivalent of the system structure has been created, the identification of what parameters, initial conditions, and behavioral relationships need to be estimated becomes clear. For instance, all exogenous variables constitute parameters that require estimation. These will act as fixed conditions during each simulation run (even if specified as stochastic variables). Likewise, all stocks will require an initial value, in order to define the initial system state. Additionally, some of the more complex causality relationships among variables might require the estimation of some sort of causality factor according to the adopted assumptions. Once all of this is completed, the model can be executed and tested.

3.5.2.4 Model Testing

The purpose of model testing is to increase confidence in the model, ultimately leading to the acceptance of the underlying dynamic hypothesis. By definition, no model can establish truth about the phenomenon under analysis. This is so because a model is a simplified representation of reality and, therefore, no matter how elaborate it is, it will never capture the full complexity of reality. Consequently, testing refers to evaluating whether or not the model is good enough for the modeling purpose at hand.

Various model testing procedures are available including: boundary adequacy, structure assessment, dimensional consistency, parameter assessment, extreme conditions, integration error, behavior reproduction, behavior anomaly, family member, surprise behavior, sensitivity analysis, and system improvement (Sterman, 2000). Behavior reproduction tests are one of the most utilized. Plotting the simulation results against real historical data in a graph representing behavior over time is particularly insightful.

The model tests performed at this step will culminate with either the acceptance or rejection of the proposed dynamic hypothesis. If accepted, the modeling effort can move to the next step. If rejected, further model development (in the form of revision of earlier steps) and testing is necessary.

3.5.2.5 System Redesign and Evaluation

The entire modeling effort performed to the point of accepting the dynamic hypothesis serves the purpose of developing an explanation of the causes of problematic behavior and identification of high leverage points in the system in order to promote improvement.

The attention is now shifted to the generation and evaluation of alternative system structures and policies capable of correcting the problem. This is where the creative insights of a designer are transformed into candidate design solutions. This goes beyond changing the values of parameters, potentially involving entirely new strategies, structures and decision rules.

In fact, the mental act of generating design alternatives is a phenomenon about which very little is known. Current knowledge regarding how behavior is generated as a function of system structure is likely to be intensively used during this process. Sterman (2000) argues that according to the understanding that the feedback structure of a system determines its dynamics, the generation of alternative designs will likely involve changing the dominant feedback loops by redesigning the stock and flow structure, eliminating time delays, changing the flow and quality of information available at key decision points, or fundamentally reinventing the decision processes of the actors in the system.

Once ideas have been generated, there are various established procedures to refine and select among a given pool of basic candidate designs. In any case, it is highly recommended that the interaction of different policies be assessed because, given the non-linear nature of complex systems, sometimes policies interfere with one another, reinforcing themselves and generating substantial synergies (Sterman, 2000).

In the case of this research the intention is to follow the guidelines suggested by Law and Kelton (2000) to compare the performance of the candidate designs against a standard (i.e., against the performance of the model representing the traditional value stream in the hypothetical case; or the performance of the model representing the current state in the real case) in order to determine whether or not each particular design alternative is capable of promoting improvement. The alternatives considered capable of promoting improvement in system performance are selected as feasible candidates. Next, the feasible candidates are ranked and the best one selected.

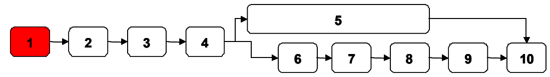
The variable that serves as a performance reference for this selection procedure is *system backlog*. There is particular interest in high performance as well as in robustness of the system regarding delivery characteristics. Overall, this same sequence of steps will be used for both the hypothetical and real cases. A fundamental difference is the availability of real data; which enables more advanced model tests, especially the ones related to behavior reproduction.

Chapter 4
Hypothetical Case and Results

This chapter presents an investigation of a hypothetical case focused on the study of value stream dynamics. In particular, it describes an effort for the search of innovative structures and policies capable of providing value streams with increased robustness under demand instability.

The structure of a generic door-to-door value stream is uncovered and formalized by means of a causal loop diagram with stocks and flows, and the underlying differential equation system. This system description serves as a baseline for analytical analyses, as well as the development and quantitative evaluation of improvement alternatives. The overall effort is guided by a system dynamics modeling methodology, as reflected in the first five steps of the adopted research process. The steps and sub steps of this research process are prescribed by Sterman (2000).

1. Problem articulation
2. Formulation of a dynamic hypothesis
3. Formulation of a simulation model
4. Model testing
5. System redesign and evaluation



4.1. Problem Articulation

Problem articulation is critical. Sterman (2000) argues that a clear purpose is the single most important ingredient for a successful modeling study, as an inadequate problem definition could put the entire modeling effort in question. Additionally, Keating (2000) affirms that many management failures can be traced back to one simple and fundamental flaw: solving the wrong problem with precision.

In light of these considerations, significant effort has been devoted in this study to defining the problem and explaining the reasons why it is a problem. As such, first an overall problematic theme is presented. Next, key related variables and concepts are identified and an appropriate time horizon for the analysis is established. Finally a dynamic problem definition is formalized by means of graphs of behavior over time (BOT).

4.1.1. Theme Selection

This study is focused on enterprise structures that are designed according to lean production design guidelines. These guidelines advocate the organization of material and information flows according to what is known as "value streams."

A key characteristic of the lean production approach is the emphasis on "pulling" production, i.e., producing only to actual customer demand as opposed to relying on forecasts. Experts in pull production argue that, despite numerous advantages, pull systems are not suitable for unstable demand environments. More details are provided in **Chapter 2.**

As one approaches the problem from this perspective, some natural questions are: *Why* do pull streams behave in this manner? *What* can be done to eliminate or improve this condition? In order to search for answers to these questions, some key variables and concepts are identified next.

4.1.2. Key Variables and Concepts

Key variables:

- Demand rate
- Production rate
- On time delivery
- Backlog
- Production capacity
- Work in process inventory

Key concepts:

- Production pull
- Demand and production leveling
- Kanban
- Heijunka box

4.1.3. Time Horizon

The magnitude of time horizon relevant to this study is on the order of one or a few months. As explained exemplified by Toyota's case (Monden, 1981), this is typically the frequency with which many pull systems make revisions and adjustments to production capacity.

4.1.4. Dynamic Problem Definition

This section converts into a graphical representation what is typically described in qualitative terms in the literature. The typical behavior of a pull system in the presence of a demand shock follows a pattern described in **Figure 4.1**. Consider the following situation: the system is initially operating in an environment with stable demand rate, backlog is under control (perhaps slightly oscillating around the target), and the level of WIP is between the minimum and maximum values allowed by the system structure.

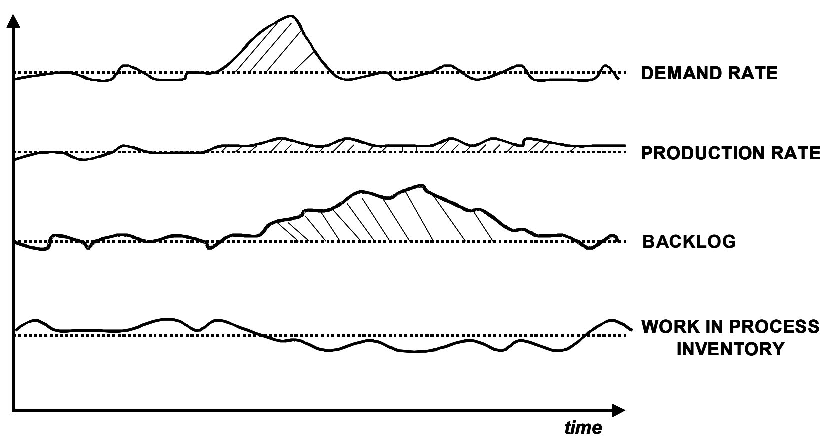


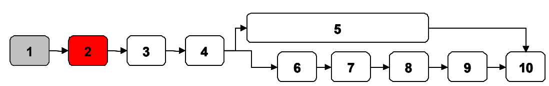
Figure 4.1 *Problematic Mode of Behavior found in Traditional Pull Systems.*

Suppose that the amount of resources allocated to production remains fixed over time and it is set at just about the right amount fulfill the average demand. The processes are under control, the quality of the product is good, the cost is predictable, and it gets delivered on time. In other words, overall the customer is satisfied.

Now imagine that the demand rate temporarily jumps abruptly to 2 or 3 times over its usual level. This is beyond the expected demand variability. Current production capacity suddenly becomes short and the production rate cannot cope with this increased demand rate. As a result, backlog starts to build up and the time to fulfill new orders start to increase. This condition increases the pressure on the production resources, which might manage to slightly increase their utilization. The level of WIP reduces a little as the PDS absorbs part of the shock. However, because resources are scarce, the small increase in production rate is still not enough to cope with the extra quantity being demanded and backlog continues to build up.

As backlog increases, on time delivery indicators show an increasing deterioration in performance. This problematic condition reaches a maximum and then starts to improve gradually. Depending on the demand behavior in the following periods, the system may eventually return to its original high performing condition but the long lasting effects of a quick demand spike may be felt long after it has passed, causing disruption to the business and potentially meaning loss of customers.

4.2. Formulation of a Dynamic Hypothesis



This initial understanding of the problem serves as input for the creation of a more formal explanation: a dynamic hypothesis. A dynamic hypothesis is a working theory of how the problem arose. The hypothesis is dynamic because it must provide an explanation of the dynamics that characterize the problem in terms of underlying feedback and stock and flow structure of the system. Additionally, it is a hypothesis because it is always temporary, subject to revision or abandonment as more learning is gained from the modeling process and from the real world (Sterman, 2000).

The effort of generating the dynamic hypothesis is here divided into three parts. First, an initial attempt to qualitatively explain the cause(s) of the problem is conducted. Next, this initial theory is converted into a map of the pertinent system structure by means of causal loops with stocks and flows. Along this mapping effort, a conscious effort to try to keep the important aspects of the problem endogenous to the description (and not as outside factors) is made. A model boundary chart is presented, classifying all variables in the map as either endogenous to the model (i.e., totally explained by the causality description created) or exogenous (i.e., as parameters to be estimated).

4.2.1. Initial Hypothesis Generation

This subsection describes an initial theory about why pull systems behave in the way the literature describes them. In fact, it constitutes a personal interpretation as well as a synthesis of fragmented explanations provided by various other authors. As such, this preliminary explanation will serve as a starting point for the more formal mapping effort later on.

It is understood that an ideal pull system is one capable of producing and delivering exactly the quantity demanded at exactly the time requested. The inability of real systems to do so consistently over time is described in terms of four main lines of thinking: (1) capacity acquisition delays, (2) limited supply availability, (3) lack of protective inventory, and (4) uneven flow.

4.2.1.1 Capacity Acquisition Delays

In any production system, there is always a delay between the request of a certain quantity and its delivery by the system. This delay reflects the fact that a physical process must take place as a result of a request; and, by definition, any process consumes time. This is true even if this process means simply withdrawing the desired quantity from a pool of finished products and handling it to the customer. The delay in this case is likely to be insignificant relative to the macro dynamics of the system. In many cases, however, this necessary physical process refers to considerably more complex operations and the dynamics involved cannot be ignored.

This means that there is always a delivery lead-time to be accounted for even if production capacity is not a constraint. Additionally, in situations where production capacity is initially found to be a constraint and later on adjusted appropriately as a result of this realization, the process of acquiring production capacity in itself also adds to the delivery delay. If the summation of these delays results in a time lag bigger than what the customer is willing to wait, then there is a problem.

If capacity could always be adjusted continuously as demanded and there was unlimited supply availability, this delivery delay would always be minimal and proportional to the combination of quantity demanded, time to acquire capacity and process cycle time. In such a scenario, inventories could be kept close to the customer in order to ensure a short customer delivery time. However, there would be no purpose in maintaining inventories as a means to absorb differences between the demand rate and the production rate. In other words, under these conditions the production rate would always be able to match the demand rate, but with a certain time lag equivalent to the minimum delivery time possible.

Many real systems, however, do not present such ideal characteristics, not only because capacity is usually limited but also because changes in required capacity (when possible) do not happen continuously over time; instead they tend to take place at fixed time intervals. This practice, associated with pressures for not acquiring or discarding resources unnecessarily contributes to increase the rigidity of the system. The result in general is a system that operates well under stable conditions but may not respond fast enough to changes in customer activity regarding the quantity demanded.

4.2.1.2 Limited Supply Availability

If the assumption of unlimited supply is relaxed, then another important factor is added to this discussion. In this case, in addition to the effects related to the capacity acquisition process just described, it is likely that there will be situations when the supplier system (and not the target system) will be limiting the flow and therefore undermining the ability of the target system to fulfill customer demand.

4.2.1.3 Lack of Protective Inventory

Inventories or accumulations can emerge simply as result of unintended differences in rates of flows between adjacent processes (Sterman, 2000). In pull systems, however, there is a rigid control over inventory levels. One use of inventories is related to a strategic decision to buffer the system against uncertainty in rates of flow. This can be referred to as “buffer inventory”. Another use of inventories is related to ensuring uninterrupted flow in the presence of batch processes along a value stream. Inventories in this second case are a necessary feature of the kanban mechanism and is here referred to as “cycle inventory” (Smalley, 2004).

In any case, according to the lean production (LP) philosophy, there is a tendency to keep inventory levels at a minimum. Consequently, the ability to maintain uninterrupted flow through the system is constantly being challenged in the face of unstable demand and processing times. In particular, the amount of buffer inventory, coupled with supply availability and capacity constraints, determine the ability of the system to respond to unexpected demand surges. However, the effectiveness of a buffer inventory strategy over time is related to the ability of replenishing it between demand surges.

4.2.1.4 Uneven Flow

The lack of synchronization in processing times along a value stream directly impacts the variability of the production rate (Gupta et al., 1999). The higher the lack of synchronization, the higher the variability of the system, the lower the ability to predict its output. This is undesirable not only because of the difficulty in managing such a system but also because lack of synchronization may also mean loss of production capacity.

In situations where batch processing (as opposed to one-piece-flow) takes place, this problem could become particularly critical considering that bigger amounts of inventory could become blocked. Under batch processing conditions, kanban mechanisms are a creative way to control inventory levels and at the same time authorize production in upstream process(es) as needed. The number of kanbans at each kanban controlled inventory point, places a cap in the inventory quantity possible.

The number of kanbans is usually calculated according to average demand and average lead-times for replenishment, plus a safety factor. However, under the guidelines of LP, there is a constant effort to eliminate waste in the form of overproduction, which means that a key objective is to keep the smallest number of kanbans in use, as an effort to maintain the lowest possible work in process (WIP) inventory in the system. If the number of kanbans, and consequently the inventory level is too low, blocking and/or starvation of processing stations could occur. It is for these reasons that, in order to ensure synchronization and avoid blocking/starvation, pull systems require strict control of processing times.

Consequently, here is a dilemma. On the one hand it is desired that the pull system will be able to quickly react to various levels of demand rates effectively. On the other hand, at every production capacity configuration adopted, the system is required to maintain synchronization, which requires adjustment time as processes are brought into control. Thus, it is expected that the pull system be at the same time responsive as well as stable; conditions some experts like Fowler (1999) suggest cannot be achieved simultaneously.

4.2.2 Mapping

The mapping effort provides a means to integrate relevant aspects from the various complementary facets of the theory just presented. By adopting a causal loop diagram with stocks and flows formalism, a higher degree of precision can be obtained in this discussion. The scope and level of aggregation will become clearer and a more accurate description of the system structure can be obtained, further enhancing the understanding about the causes of the problem.

4.2.2.1 Scope and Level of Aggregation

In order to start the mapping effort, a more formal baseline than the one presented so far needs to be established. The purpose of this enhanced baseline is to provide a clearer indication of the scope of the system addressed in this study as well as the level of detail believed appropriate to the problem under consideration.

In order to do so, this investigation focuses on the concept of door-to-door value stream, presented in **Figure 4.2** as the basic building block of lean enterprises. It is acknowledged, however, that the total value stream often extends beyond the boundaries of a single organization. The expression “door-to-door” is intended to clearly indicate the scope of the system, and refers to production flow inside a particular plant or enterprise. A major reason for this choice is that it clearly establishes a management domain, facilitating the precise identification of responsibility, accountability, and control.

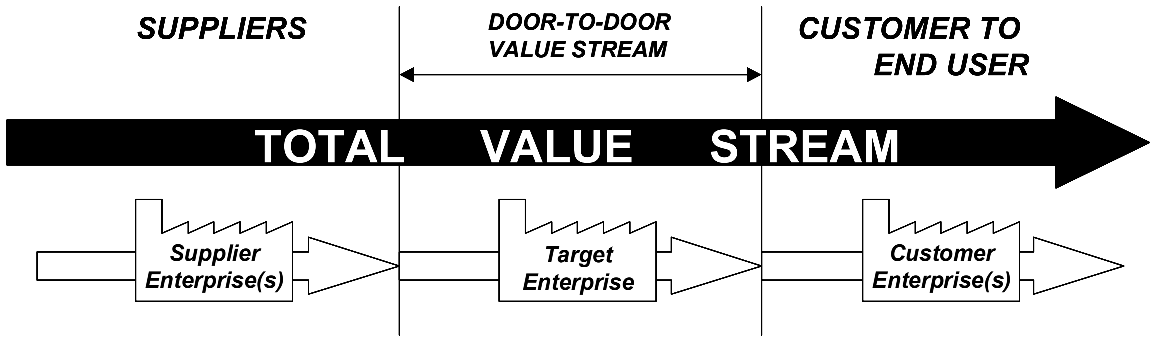


Figure 4.2 *Door-to-door Value Stream.*
Source: adapted from Rother and Shook (1999)

Additionally, by focusing on a specific product family, a specific value stream of interest within the target discrete manufacturing lean enterprise can be identified. This is so because, by definition, a value stream serves to promote all the flow of materials required to deliver units of a certain product family to customers (Rother & Shook, 1999). Furthermore, if it is assumed that there is no significant difference among product family variants, the unit flowing through the system, for the purpose of this study, could be simply referred to as "product family unit" without further distinction.

4.2.2.2 Structure of a Generic Door-to-Door Value Stream

Based on the intended scope and level of detail, a closer look at a door-to-door value stream structure is likely to reveal that it is composed of multiple processing steps (see **Figure 4.3** – for clarification purposes, one processing step ends and another one starts when there is the physical possibility of inventory accumulation between them). Ideally in a lean system parts will flow from one step to the next in a one-piece-flow configuration, meaning that, at maximum, just the unit being worked on is kept as inventory between steps. Additionally, a distinctive characteristic of value streams is that flows takes place only as a result of a customer pull, which is deployed in a cascade effect throughout the processing steps in the system.

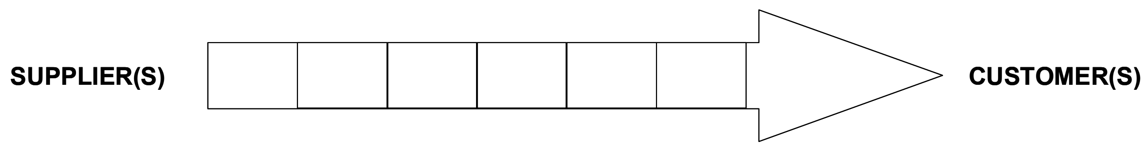


Figure 4.3 *Steps in a Door-to-door Value Stream.*

Furthermore, approaching the system from a control perspective reveals that in every value stream (just like with production systems in general) there is a constant effort to match the rate of production to the rate of demand. In this case, this is accomplished in the following way: information about customer orders is interpreted and sent to the pacemaker process, i.e., a single processing step in the value stream where production is scheduled. In summary, the pacemaker is the one processing step controlled by the outside customer's orders, which ends up dictating the production pace of the whole system (Rother & Shook, 1999).

In this study, for simplification purposes, it is considered that the pacemaker process lies within the door-to-door value stream of interest. There are situations in industry where the pacemaker actually falls outside this boundary and is, for instance, located in the customer's value stream. In these situations, the boundary of analysis needs to be adapted and the concept of *extended enterprise* (Vernadat, 1996 p.4,8) incorporated so that a proper domain that includes the pacemaker is delimited. Given the inclusion of the pacemaker, the concepts here presented remain valid independent of where the frontier between the firms involved is located.

The pacemaker is a very important component in understanding the value stream structure and dynamics. Upstream from the pacemaker material is pulled. Downstream from the pacemaker, the movement of parts needs to occur in a flow, i.e., units are "pushed" to the next production step as they get ready. **Figure 4.4**. Illustrates this concept.

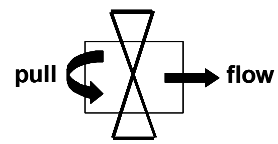


Figure 4.4 *Pull versus Flow at the Pacemaker.*

Consequently, the position of the pacemaker process actually determines what parts of the door-to-door value stream occur in a pull mode and what parts occur in a flow. As illustrated in **Figure 4.5**, these two distinct parts of the system are here referred to, respectively, as "upstream processes" and "downstream processes". In a leveled value stream, the customer orders are carefully filtered so that the pacemaker receives a steady stream of work instructions according to the system production capacity. This is achieved via a mechanism known as load leveling box or heijunka box (Rother & Shook, 1999; Smalley, 2004). Such stability allows the value stream to obtain important benefits of pull production.

However, a value stream also needs to be able to absorb demand instabilities. Again just like any production system, there are two main ways to do so. These refer to the isolated or combined use of (1) protective production capacity, and (2) protective inventory. In this case, protective production capacity refers to available production capacity beyond the necessary to fulfill average customer demand and its function is to enable the absorption of variability beyond the average demand rate. The amount of protective production capacity in upstream and downstream processes need not be necessarily the same. In any case, LP strives to remove waste as much as possible and carrying any sort of protection is likely to be questioned.

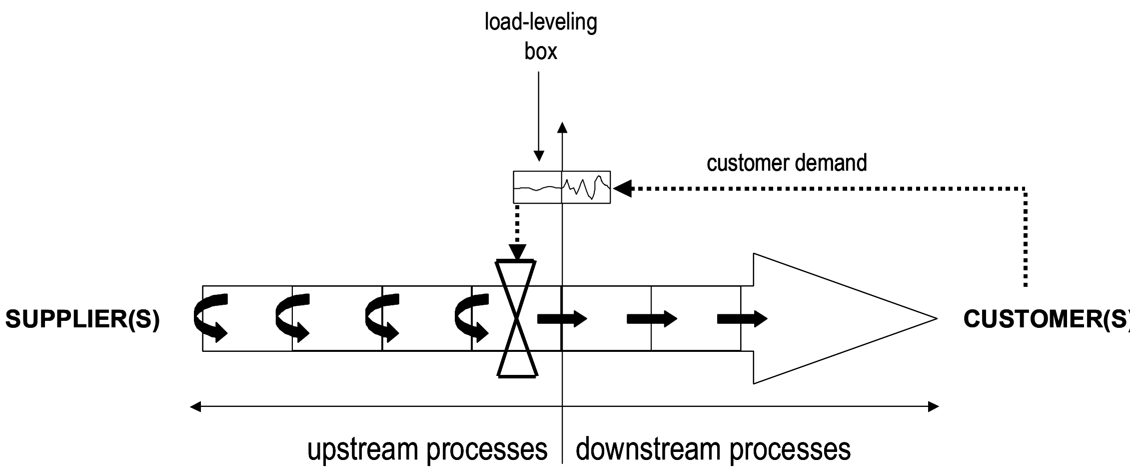


Figure 4.5 *Upstream versus Downstream Processes.*

The protective inventory refers to a conscious decision to carry a certain amount of inventory to buffer the system against demand (as well as supply) uncertainty. The protective inventory, coupled with downstream protective capacity, is a strong determinant of the ability of the system to respond to temporary demand shocks. It is important to note, as previously indicated, that value streams keep a very strict control over any inventory carried in the system. In LP terminology, this means that protective inventory needs to work as a "supermarket", which is in fact a kanban-controlled inventory.

The use of supermarkets along value streams is what makes it possible in many cases to send the production schedule only to a single production step in the system. The kanban systems regulating the supermarkets establish a cap for each supermarket size, and at the same time transfers the scheduling instruction to the immediate upstream step, promoting the upstream flow of information in a cascade effect.

However, it should be emphasized that the protective supermarket as well as any other supermarkets (i.e., supplied parts and intermediary supermarkets) have to be positioned somewhere in the upstream processes. In other words, as a result of the value stream control structure, by definition no supermarkets or pulls can take place in downstream processes. **Figure 4.6** illustrates a protective decoupling supermarket (PDS) located immediately next to the pacemaker.

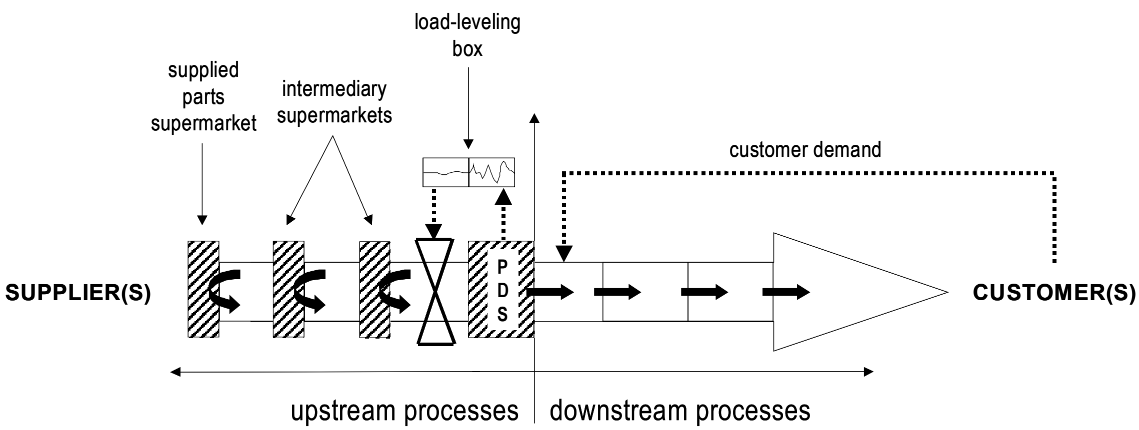


Figure 4.6 *Value Stream Structure with Protective Decoupling Supermarket.*

In a value stream that utilizes protective inventory, the customer orders are sent to the downstream step immediately next to the PDS. As the PDS level decreases, kanban signals are sent to the pacemaker process via the load-leveling box, which promotes a steady disbursement of work instruction to the pacemaker ensuring leveling. Ideally, customer rate and production rate are the same. When customer rate surpasses the upstream production capacity, the difference causes an accumulation of orders here referred to as "demand overflow."

Finally, it should also be noted that there is the possibility of inventory accumulation downstream in the system should a lack of synchronization occur among the downstream processing steps. In such situations inventory could emerge naturally as a result of net differences between rates of flow in adjacent processing steps (a situation that could occur for example if a downstream resource fails). However, since inventory accumulation between downstream steps is not desired, a common practice to avoid inventory overflow downstream is to utilize a FIFO lane with a set maximum limit of units in queue - an approach referred by some as CONWIP or constant work in process (Spearman, Woodruff & Hopp, 1990; Sipper & Bulfin Jr., 1997 p.591). Consequently, in the event of a break down, downstream inventories would be allowed to grow only so much before flow is interrupted along the immediate preceding step, and eventually along the entire value stream in a cascade effect.

4.2.2.3 Value Stream Mapping Notation

In value stream mapping (VSM) notation (Rother & Shook, 1999), the generic structure and components of a value stream just described is illustrated in **Figure 4.7**. Note that in order to obtain the simplest representation as a baseline, there is only one upstream processing step and only one downstream processing step, i.e., here the only possible inventory points are the supplied parts supermarket and the protective inventory supermarket. More complex value streams can be obtained by "exploding" the upstream and/or the downstream processes.

This completes the introductory description of the structure of a generic door-to-door value stream. Next, more advanced causality considerations are incorporated by gradually converting this description into a causal loop diagram with stocks and flows.

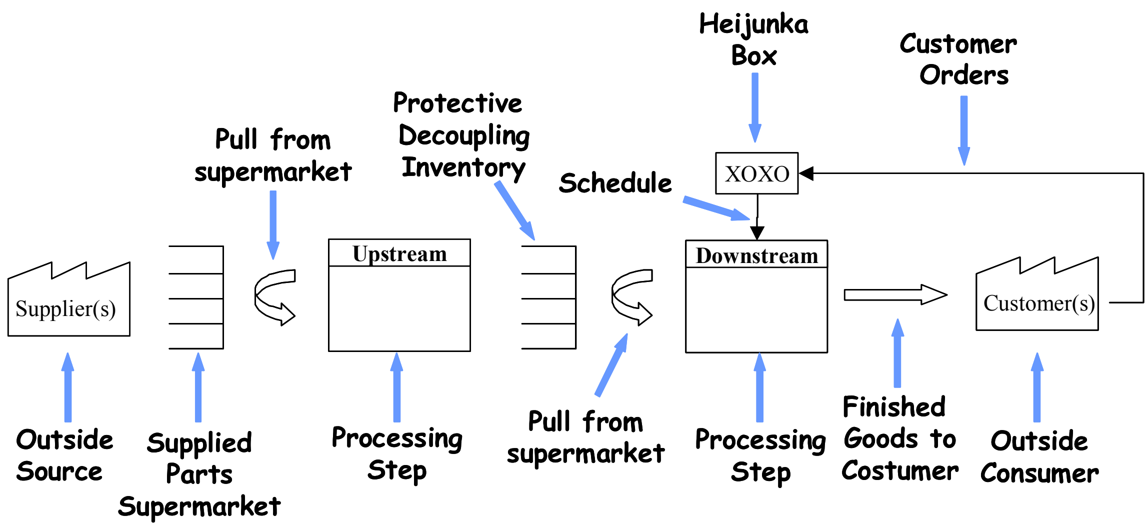


Figure 4.7 *Generic Value Stream Structure in VSM Notation.*

4.2.2.4 Causal Loop Diagram with Stocks and Flows

First, consider the policy diagram in **Figure 4.8**. It shows some key relationships taking place in a generic door-to-door value stream at a total system level. The *Production Planning and Control (PPC)* subsystem affects the upstream and downstream flows (and therefore the *Protective decoupling inventory (PDI)* level, as well as the ability to meet customer demand) in various ways.

In any value stream, the PPC subsystem initially translates the actual customer demand rate into a production schedule for the downstream processes. As the *PDI* level decreases, signals are sent to the pacemaker, located in the upstream processes, via a load leveling box. As the *downstream production rate (DPR)* takes place, demand is fulfilled; therefore affecting back the *PPC* subsystem in its function of providing a production schedule for future planning periods. The *PPC* subsystem also allocates production power to the upstream and downstream processes taking into account the demand history, which consequently places a cap on the maximum production rate the value stream is capable of generating.

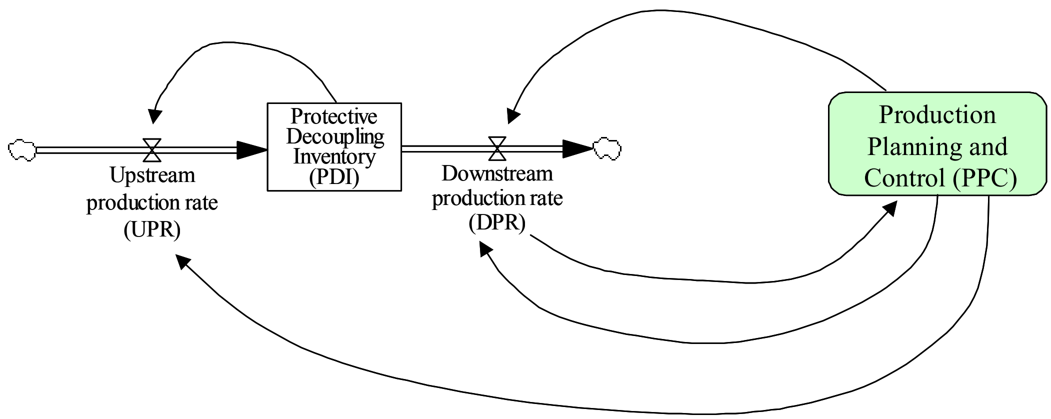


Figure 4.8 *Policy Diagram for the Door-to-door Value Stream.*

In order to explore more details of the system structure, the policy diagram is exploded into a full causal loop diagram with stocks and flows representative of the generic value stream baseline defined in **Figure 4.7**. This will be accomplished by first directing attention to the *PDI* stock, and then gradually adding details around it.

Simply put, the *PDI* stock accumulates the net difference of the upstream and downstream rates. In discrete parts manufacturing, the *PDI* level is regulated by a kanban mechanism that at the same time establishes a target inventory level and communicates the need for replenishment to the upstream processes. When the *PDI* level goes below the target level, a signal (i.e., a kanban) corresponding to the gap between desired and actual inventory levels is sent to the immediate upstream process. This mechanism creates a balancing feedback loop as represented in **Figure 4.9**: when the *PDI* level decreases (increases) *adjustment for PDI* increases (decreases) and *upstream production rate (UPR)* increases (decreases). Note that the *UPR,* by itself, can only increase the *PDI*. On the other hand, the *downstream production rate (DPR)* can only decrease the *PDI.*

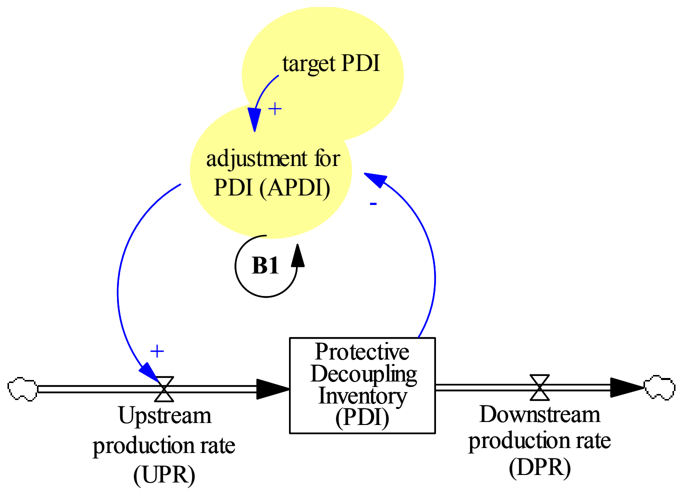
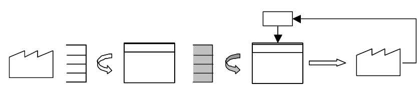


Figure 4.9 *PDI Adjustment Loop*  .

DPR is set according to a *desired DPR,* as shown in **Figure 4.10**. Furthermore *DPR* can only reduce the *PDI* down to zero, i.e., in any real system, the *PDI* is always nonnegative. In order to ensure this condition, a reinforcing loop R1 is inserted in the model: when *PDI* decreases to zero, *DPR* must also decrease to zero. *Desired DPR* is set according to interpretation of actual customer demand. Before exploring this aspect of the system, a closer look on the system production power relationships is provided.

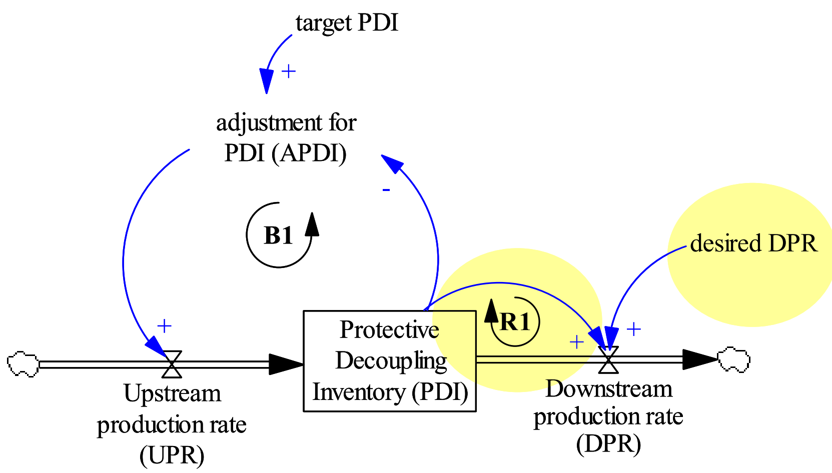
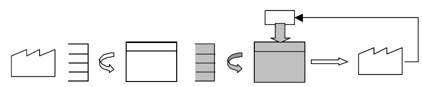


Figure 4.10 *PDI's Nonnegative Condition*  .

In value streams, the production power or capacity (i.e., the amount of work that can be performed over a certain time interval) can be defined according to a combination of two variables: base capacity and protective production capacity factor (see **Figure 4.11**). For instance, the *downstream production capacity (DPR)* can be defined by the *base downstream production capacity (BDPC)* plus a *protective downstream production capacity factor (PDPCF*), where the protective capacity can be defined as the capacity that surpasses the amount needed to attend the historical average demand. The same idea applies for the *upstream production capacity (UPC).*

***Assumption 1**: The base capacity is set according to the average historical customer demand observed during an estimation period.*

The downstream capacity defines the maximum downstream production rate or, in other words, the minimum downstream cycle time. When *average demand rate in previous estimation* *period (ADPEP)* increases (decreases), *base downstream production capacity (BDPC)* increases (decreases), *downstream production capacity (DPC)* increases (decreases). When the *protective downstream production capacity factor (PDPCF)* increases (decreases), DPC increases (decreases). The same logic is valid for the upstream process capacity (see **Figure 4.11**).

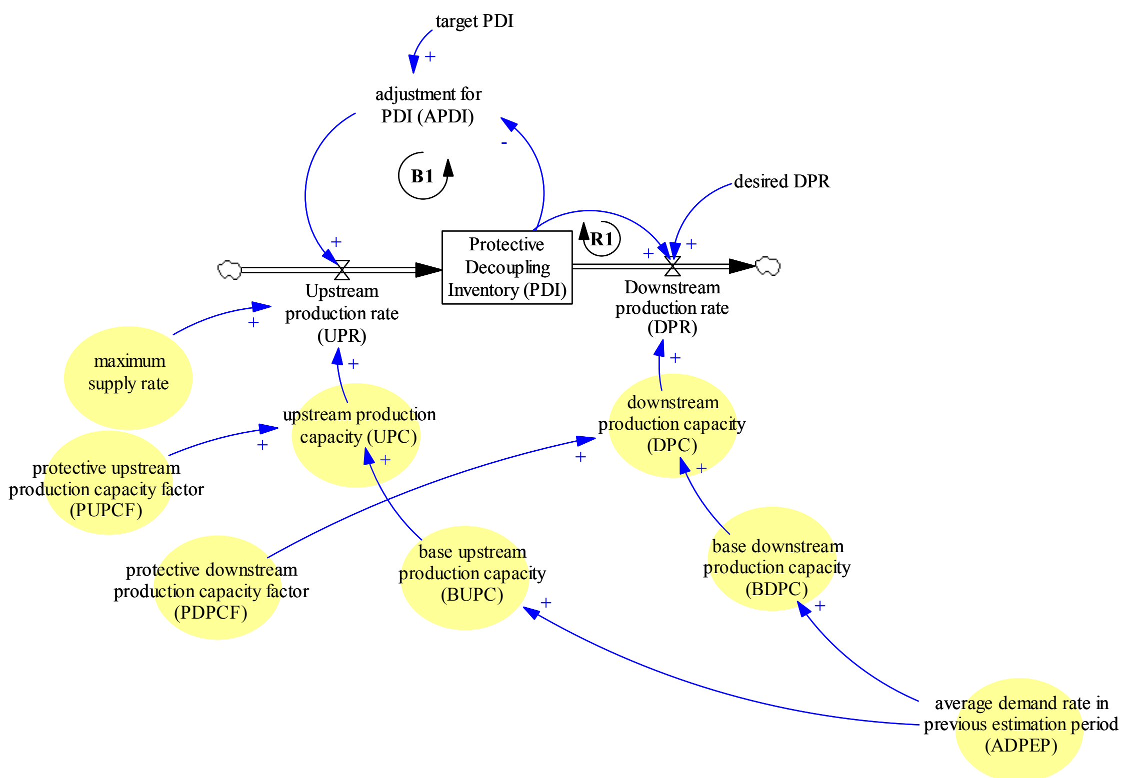
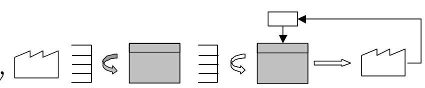


Figure 4.11 *Production Capacity*  .

Assumption 2*: The supplier value stream production capacity equals the upstream production capacity (UPC). Supply availability is unlimited but maximum supply rate equals to UPC.*

The concept of estimation period referred to earlier corresponds to a period of time used to calculate the average historical demand in order to estimate the production capacity needed at the next estimation period.

Assumption 3: *Value stream production capacity is adjusted only at the beginning of each estimation period.*

Typically, as illustrated in **Figure 4.12**, an estimation period may contain various planning periods (PP). At the beginning of each PP, customer demand is allocated to the value stream in the form of a production schedule. For instance, a common practice seems to be to treat each work shift as a PP, and set the estimation period to a month or so.

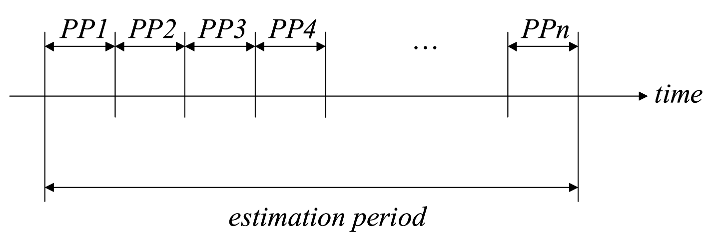


Figure 4.12 *Estimation vs. Planning Periods.*

The customer demand allocated to the downstream production processes is captured through the stock variable *Demand to be fulfilled in current planning period,* as illustrated in **Figure 4.13**. A second balancing feedback loop can be identified in the following manner: when *Demand to be fulfilled in current planning period* increases (decreases), *desired DPR* increases (decreases), *DPR* increases (decreases), *fulfillment rate in current planning period* increases (decreases), which decreases *Demand to be fulfilled in current planning period.* The implicit target in this balancing loop is to drive *Demand to be fulfilled in current planning period* to zero. Note that *fulfillment rate in current planning period* can only decrease *Demand to be fulfilled in current planning period.*

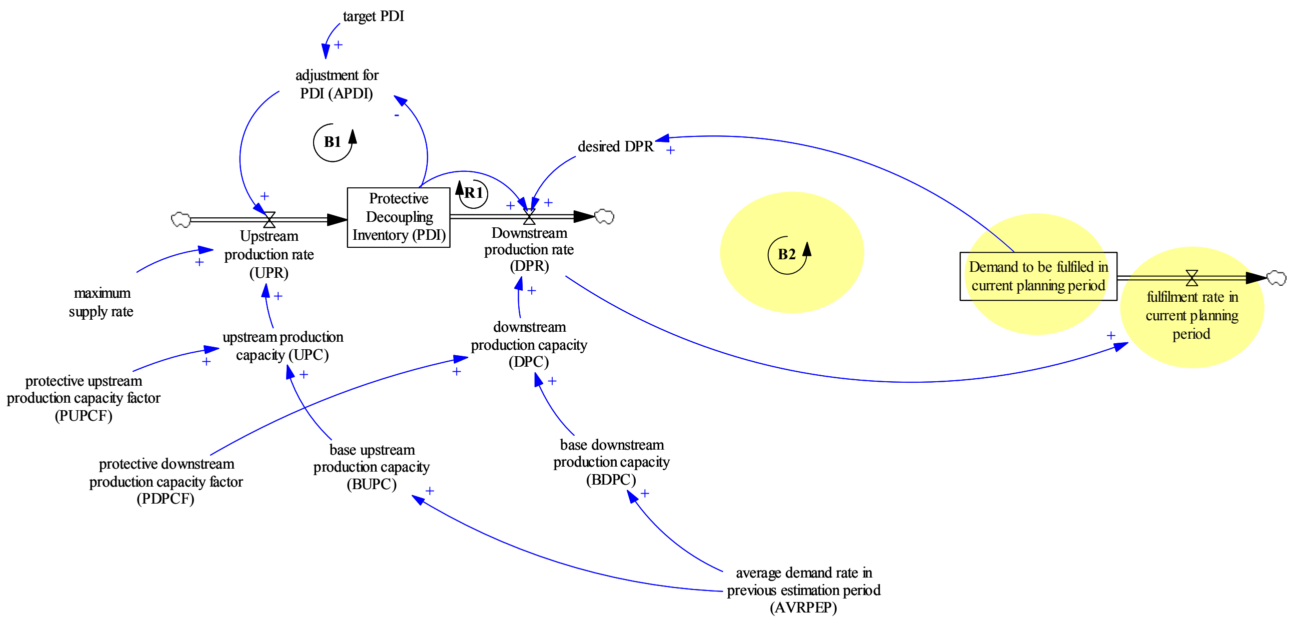
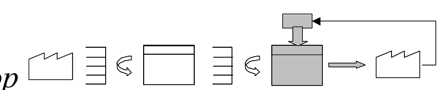


Figure 4.13 *Demand Fulfillment Loop*  .

As illustrated in **Figure 4.14**, *Demand to be fulfilled in current planning period* is increased by *rate of demand allocation to next planning period.* This quantity flows from a *Demand incurred in current planning period,* which is increased by the *demand rate.* What happens is that at the beginning of each PP, *Demand to be fulfilled in current planning period* is increased a certain quantity so that any amount remaining from the previous PP plus the allocated amount is less than or equal to a *demand quantity for a planning period.*

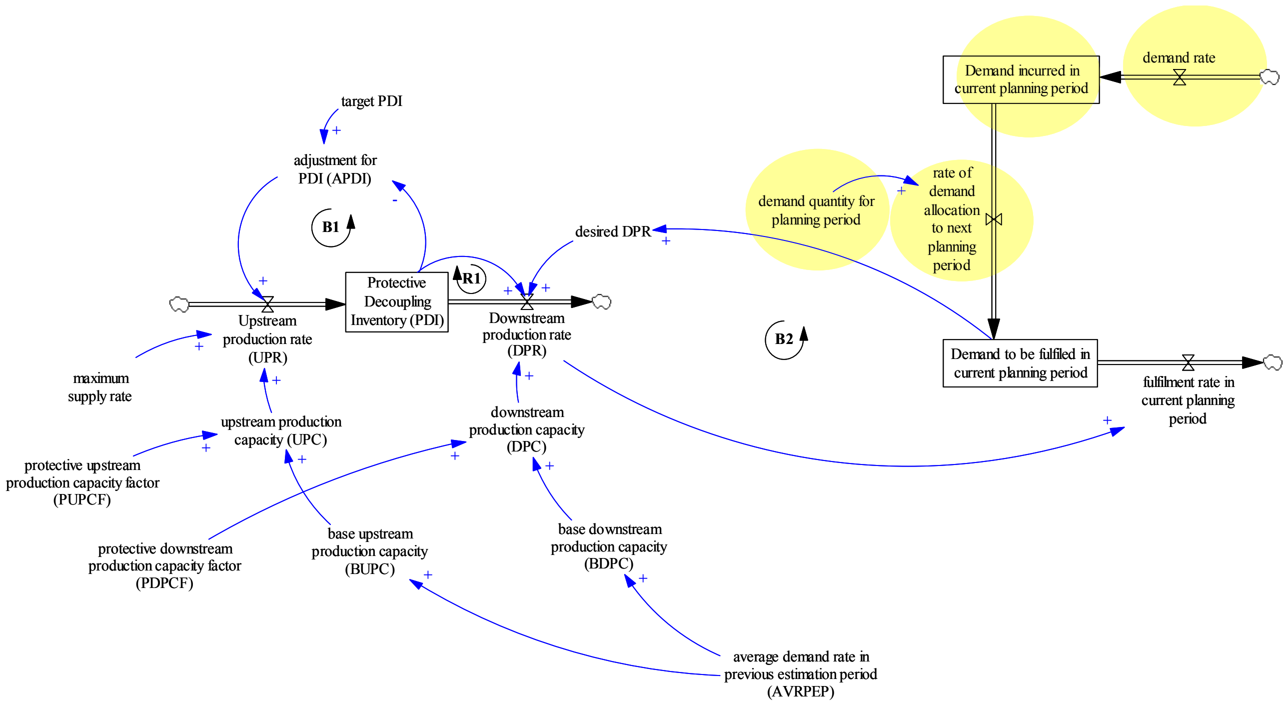
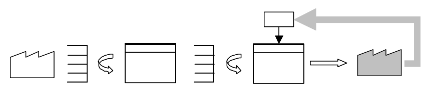


Figure 4.14 *Demand Allocation*  .

The *demand quantity for a planning period* corresponds to the estimated amount of demand the value stream can absorb in a PP. It is calculated by multiplying the *available production time in current planning period* by the *estimated production capacity of the system.* The *estimated production capacity of the system* is considered the minimum between *UPC* and *DPC* (see **Figure 4.15**).

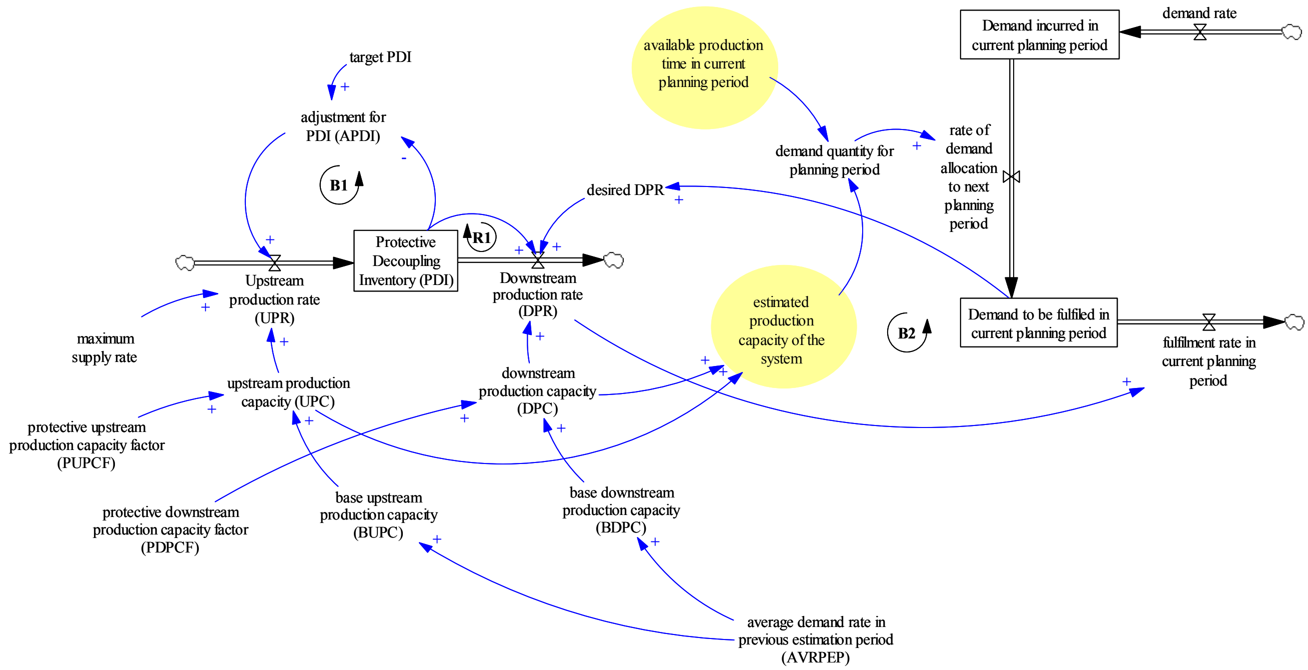
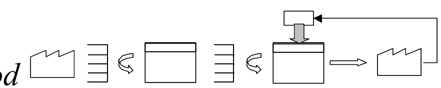


Figure 4.15 *Demand Allocation to a Planning Period* .

Furthermore, in order to better capture the discrepancies between the *demand rate* and the *fulfillment rate in current planning period* the variable *Demand overflow* is also included in the model. As such, at the beginning of each PP, any excess demand is allocated to this stock. Likewise, at the beginning of each PP, quantities from the *Demand overflow* inventory will be allocated to production first, followed by quantities from the *Demand incurred in current planning period. Demand overflow* is increased by *rate of demand overflow* and decreased by *rate of demand overflow allocation to current planning period* (see **Figure 4.16**).

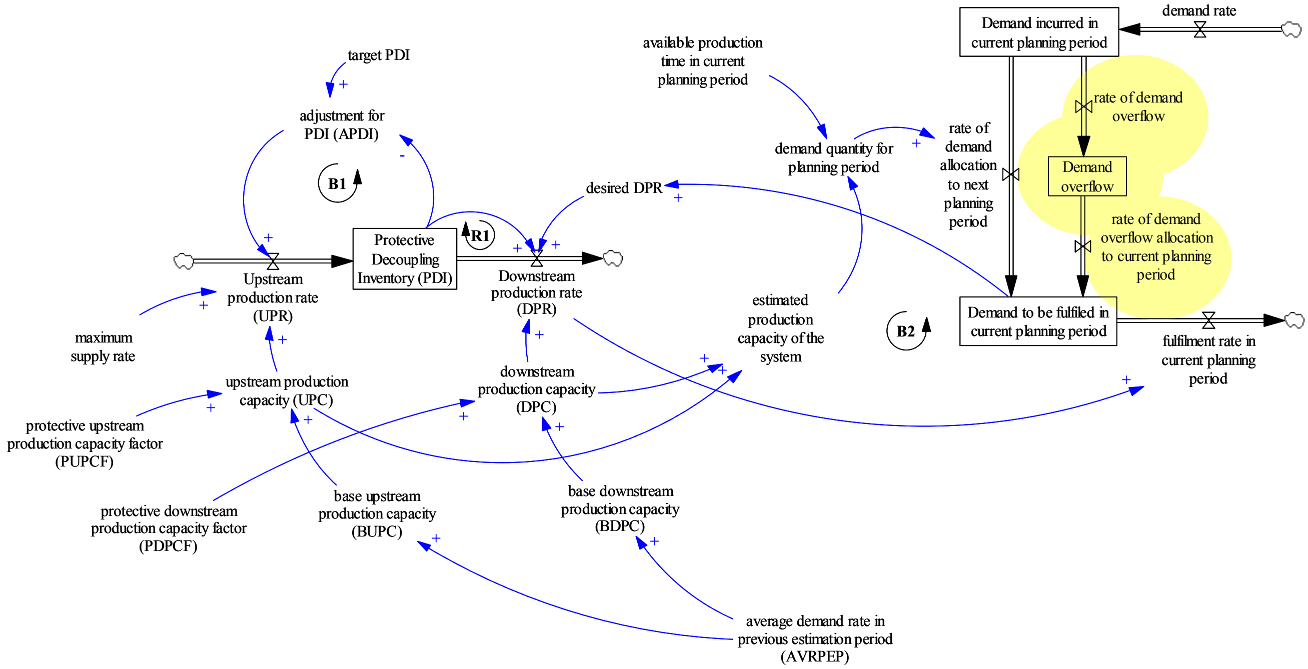
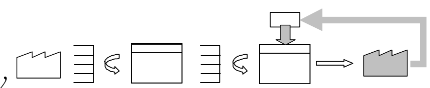


***Figure 4.16** Demand Overflow*  .

For planning purposes, the actual *demand rate* is usefully measured as *takt time*, which is defined as the inverse of the *demand rate*. As suggested in **Figure 4.17**, at the beginning of each PP, takt time is calculated by dividing the *available production time* by the *Demand incurred in current planning period.*

***Assumption 4**: The performance of the system does not significantly affect the demand rate, i.e. for the purpose of this study, it is appropriate to model demand rate as an exogenous variable.*

***Assumption 5**: The delivery lead-time has already been negotiated with the customer; as such, demand quantities accounted in the model are due immediately.*

The *takt time* provides a notion of desired production pace for the entire value stream. Ideally the downstream and upstream cycle times will equal the *takt time,* and therefore the system production rate will equal the *demand rate* and the system will be in dynamic equilibrium. This condition relates to the fundamental goal of matching production and demand over time. Furthermore, when *Demand to be fulfilled in current planning period* is greater than zero, *desired DPR* equals *estimated production capacity of the system*.

Assumption 6: *Work does not extend to fulfill all available time, i.e. if half a day's worth of work is sent to the floor at the beginning of a PP, production takes place at full pace during the first half of the day and then stops, instead of at half pace during the entire PP.*

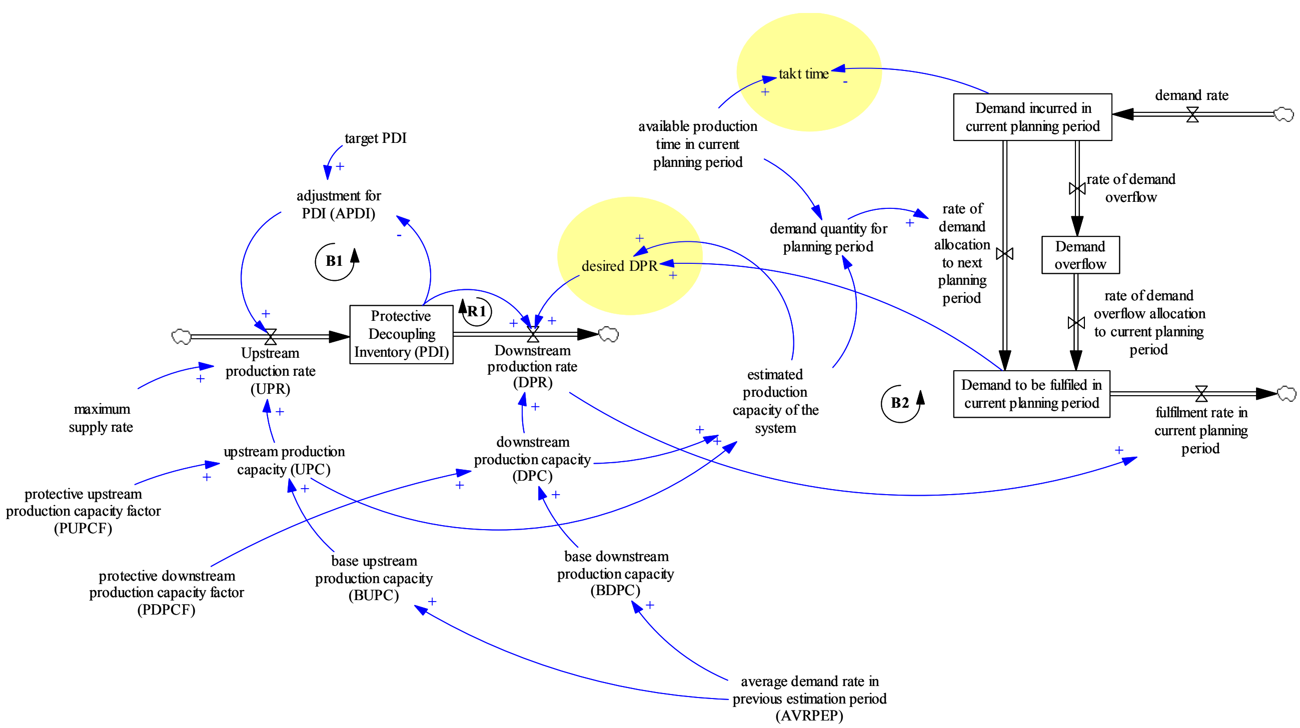
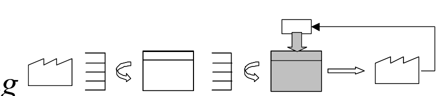


Figure 4.17 *Takt time and Demand Leveling*  .

4.2.2.5 About "Discrete" Flows

The model structure as presented to this point provides a representation of generic value stream structures in continuous terms. In other words, in order to better reflect the discrete-parts manufacturing situations of interest in this study, the insertion of "discrete" flows (both in terms of materials and information) is needed to enable more precise considerations, particularly in what involves the kanban mechanism captured in loop B1. The overall causal structure is similar to the one described to this point. Some additional assumptions follow.

> ***Assumption** 7: the amount of protective capacity in both upstream and downstream processes equals one standard deviation of demand, as measured in the previous estimation period.*
>
> ***Assumption 8**: the size of the Protective Decoupling Inventory is set according to the guidelines suggested by Smalley (2004).*
>
> ***Assumption 9**: there is no advance notice on demand; all demand incurred is due immediately.*

4.2.2.6 Model's Generic Properties

This hypothetical model developed here is intended to be a reference model. Because it captures the main elements that define a value stream, it is a generic representation. If necessary, the upstream and downstream steps might be interpreted as aggregate representations of more complex upstream and downstream process networks. This means that as more detail is added in the description of the upstream and/or downstream processes (e.g., inclusion of more steps and supermarkets), more precision can be obtained in estimating the downstream and/or upstream rates (see **Figure 4.18**).

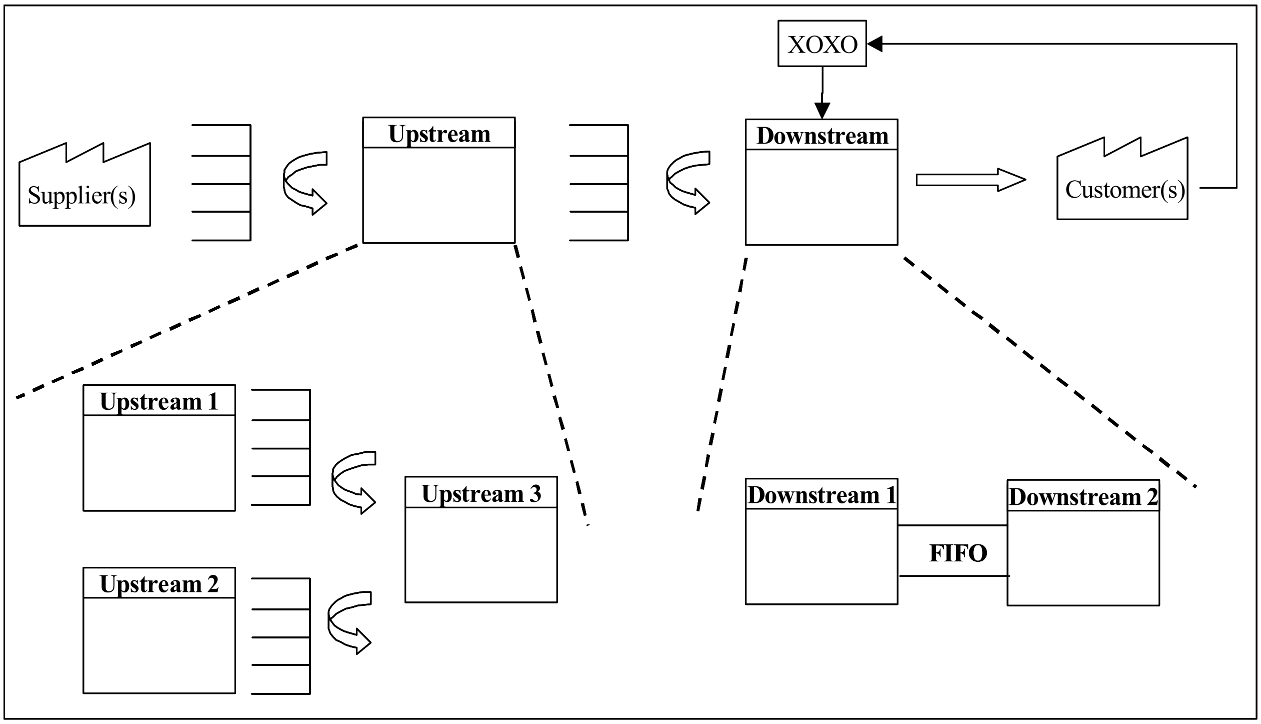


Figure 4.18 *The aggregate nature of the hypothetical model.*

As an important observation, the hypothetical system dynamics model developed in this chapter can be used to represent value streams that are either entirely replenishment pull (i.e., hold a finished or semi-finished goods supermarket for all product family variants) or sequential pull (i.e., producing all product family variants to order from the beginning), but not mixed pull (i.e., some product family variants on replenishment pull and others on sequential pull). The ability to represent mixed pull situations will require adaptation and further development of the model structure.

4.2.3. Endogenous Focus

The purpose of the model is to create an explanation of the problem derived as much as possible from endogenous relationships among the variables identified. Therefore, boundary decisions were constantly being made along the creation of the dynamic hypothesis, that is, choices were made in terms of what variables remained endogenous (i.e., completely described by other variables in the model), what variables remained exogenous (i.e., variables kept in the model but whose behavior need to be assumed), and what variables were excluded (i.e., variables with potential relevance that have not been considered as part of the dynamic hypothesis at this point). A model boundary chart (**Table 4.1**) was created and updated along the modeling process for the purpose of portraying this classification.

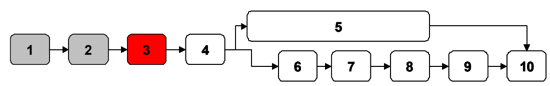
Table 4.1 *Model Boundary Chart for Hypothetical Map.*

Endogenous	**Exogenous**	**Excluded**
IUUP: Items in use at upstream process	SPP: size of planning period	Demand forecast
UPR: upstream production rate	ADRPEP: average demand rate in previous estimation period	Known demand that is not due immediately
RWPDI: rate of withdrawal from PDI	DR: demand rate	
IUDP: Items in use at downstream process	PUPCF: protective upstream production capacity factor	
PDI: Protective decoupling inventory	PDPCF: protective downstream production capacity factor	
DPR: downstream production rate	MSR: maximum supply rate	
RDS: rate of delivery from supplier	SDDPEP: standard deviation of demand rate in previous estimation period	
AKHB: add PDI kanban to heijunka box	time	
HB: Heijunka box	dt: time step	
AKSF: assign kanban to shop floor	NKpitch: Number of kanbans in a pitch	
APDIK: Assigned PDI kanbans	KS: kanban size	

SOF: supplier order fulfillment	one kanban
UCUP: Units completed by upstream process	NSDDBS: Number of standard deviations of demand rate used for the calculation of buffer stock
RUCU: Rate of UCUP clean up	EUPS: expected upstream process shortfall
DDPR: desired DPR	
UPC: upstream production capacity	
BUPC: base upstream production capacity	
DPC: downstream production capacity	
BDPC: base downstream production capacity	
EPCS: estimated production capacity of the system	
ADQPP: average demand quantity for a planning period	
ATPP: Available production time in current planning period	
TT: takt time	
clock time	
DFCPP: Demand to be fulfilled in current planning period	
FRCPP: fulfillment rate in current planning period	
DO: Demand overflow	
RDO: rate of demand overflow	
RDOCPP: rate of demand overflow allocation to current planning period	
RDAPP: rate of demand allocation to next planning period	
DICPP: Demand incurred in current planning period	
RDASF: rate of demand allocation to shop floor	
DBWO: Demand being worked on	
QOC: Quantity on order from external customer	

Sterman (2000) argues that narrow model boundaries are a critical source of policy resistance. As such, the number of exogenous inputs should be small, and each candidate for exogenous input must be carefully scrutinized to consider whether there are in fact any important feedbacks from the endogenous elements to the candidate. Should this be the case, the boundary of the model must be expanded and the variable must be modeled endogenously. Consequently, as the boundary of the model expands in the search for a better explanation for the problem, exogenous variables might become endogenous, new exogenous variables may be created and variables once excluded might start to be considered as part of the model. As an observation, by definition the modeling process never ends; a model can never be perfect, it can only get good enough for the decisions it is intended to support.

4.3. Formulation of a Simulation Model



In order to quantitatively evaluate the dynamics captured by the map created, a simulation model was derived from it and then executed in a computer. Formulating a simulation model involves specifying the analytical relationships between the variables identified, creating a system of differential equations. In this study, this equation system was later on solved through numerical integration by means of a software package named Vensim® (2002).

4.3.1. Specification of Structure and Decision Rules

Part of the mathematical equivalent of the map can be interpreted directly from the connection between stocks and flows (i.e., a stock is interpreted as an integral of the difference between inflows and outflows to that stock, and each of the flows is interpreted as a derivative of the quantity flowing in reference to time). The remaining relationships between variables required to fully describe the flows – which include the policies i.e., decision rules – were specified analytically according to the available value stream mapping theory described previously. Once this was accomplished, then parameters and initial conditions were set, including a choice of time step for the numerical integration procedure.

4.3.2 Model Equations

$$PDI(t) = \left[\int_{t_0}^{t} (UPR(t) - RWPDI(t))dt \right] + PDI(t_0) \tag{4.1}$$

$UPR(t)$:
IF $IUUP(t) > 0$
THEN $UPR(t) = UPC(t)$
ELSE $UPR(t) = 0$ (4.2)

$$IUUP(t) = \left[\int_{t_0}^{t} (RDS(t) - UPR(t))dt \right] + IUUP(t_0) \tag{4.3}$$

$RDS(t)$:
IF $QOS(t) > 0$
THEN $RDS(t) = UPC$
ELSE $RDS(t) = 0$ (4.4)

$$QOS(t) = APDIK(t) \cdot KS \tag{4.5}$$

$$APDIK(t) = \left[\int_{t_0}^{t} (AKSF(t) - SOF(t))dt \right] + APDIK(t_0) \tag{4.6}$$

$$SOF(t) = \frac{RDS(t)}{KS} \tag{4.7}$$

$$UPC(t) = BUPC(t) + PUPCF \cdot SDDPEP \tag{4.8}$$

$$BUPC(t) = ADPEP \tag{4.9}$$

$RWPDI(t)$:
IF $DFCPP(t) \geq KS \cdot onekanban$
AND $PDI(t) \geq KS \cdot onekanban$

THEN $RWPDI(t) = (KS \cdot onekanban)/dt$
ELSE $RWPDI(t) = 0$ (4.10)

$$HB(t) = \left[\int_{t_0}^{t} (AKHB(t) - AKSF(t))dt \right] + HB(t_0) \tag{4.11}$$

$AKHB(t)$:
IF $RWPDI(t) > 0$
THEN $AKHB(t) = \dfrac{onekanban}{dt}$
ELSE $AKHB(t) = 0$ (4.12)

$AKSF(t)$:
IF ($UCUP(t) \geq pitch$ **AND** $HB(t) \geq 1$**)**
THEN $AKSF(t) = \dfrac{onekanban}{dt}$
ELSE $AKSF(t) = 0$ (4.13)

$$UCUP(t) = \left[\int_{t_0}^{t} (UPR(t) - RUCU(t))dt \right] + UCUP(t_0) \tag{4.14}$$

$RUCU(t)$:
IF $AKSF(t) <> 0$
THEN $RUCU(t) = \dfrac{pitch}{dt}$
ELSE $RUCU(t) = 0$ (4.15)

$$IUDP(t) = \left[\int_{t_0}^{t} (RWPDI(t) - DPR(t))dt \right] + IUDP(t_0) \tag{4.16}$$

$DPR(t)$:
IF $IUDP(t) > 0$
THEN IF $DPC(t) \geq desiredDPR(t)$
  THEN $DPR(t) = desiredDPR(t)$
  ELSE $DPR(t) = DPC(t)$
ELSE $DPR(t) = 0$ (4.17)

$$DPC(t) = BDPC(t) + PDPCF \cdot SDDPEP \tag{4.18}$$

$$BDPC(t) = ADPEP \tag{4.19}$$

$$desiredDPR(t) = DPC(t) \tag{4.20}$$

$$DICPP(t) = \left[\int_{t_0}^{t} (DR - RDAPP(t) - RDO(t))dt \right] + DICPP(t_0) \tag{4.21}$$

$RDAPP(t) =$
IF MODULO(*clock time, SPP***)**=0
THEN IF $DICCP(t) > 0$
    THEN IF $DFCCP(t) + DO(t) \geq DQPP(t)$
        THEN $RDAPP(t) = 0$
        ELSE IF $DICCP(t) \leq DQPP(t) - DFCPP(t) - DO(t)$
            THEN $RDAPP(t) = DICPP(t)/dt$
            ELSE IF $DICPP(t) > DQPP(t) - DFCPP(t) - DO(t)$
                THEN $RDAPP(t) = \left[DQPP(t) - DFCPP(t) - DO(t)\right]/dt$
                ELSE $RDAPP(t) = 0$
    ELSE $RDAPP(t) = 0$
ELSE $RDAPP(t) = 0$ (4.22)

$$ADQPP(t) = \frac{KS(t) \cdot onekanban}{ESCTKQ(t)} \cdot ATPP(t) \tag{4.23}$$

$$EPCS(t) = \mathbf{MIN(}UPC(t), DPC(T)\mathbf{)} \tag{4.24}$$

$$ATPP(t) = SPP \tag{4.25}$$

$$DFCPP(t) = \left[\int_{t_0}^{t} (RDAPP(t) + RDOCPP(t) - RDASF(t))dt \right] + DFCPP(t_0) \tag{4.26}$$

$RDOCPP(t) =$
IF MODULO(*clock time, SPP***)**=0
THEN IF $DO(t) > 0$
    THEN IF $DO(t) \leq DQPP(t) - DFCPP(t)$
        THEN $RDOCPP(t) = DO(t)/dt$
        ELSE IF $DO(t) > DQPP(t) - DFCPP(t)$
            THEN $RDOCPP(t) = \left[DQPP(t) - DFCPP(t)\right]/dt$
            ELSE $RDOCPP(t) = 0$
    ELSE $RDOCPP(t) = 0$
ELSE $RDOCPP(t) = 0$ (4.27)

$$RDASF(t) = RWPDI(t) \tag{4.28}$$

$$DO(t) = \left[\int_{t_0}^{t} (RDO(t) - RDOCPP(t))dt \right] + DO(t_0) \tag{4.29}$$

$RDO(t)$:
IF MODULO(*clock time, SPP***)**=0
THEN IF $DO(t) + DFCPP(t) \geq DQPP(t)$
    THEN $RDO(t) = DICPP(t) / dt$
    ELSE IF $DO(t) + DFCPP(t) + DICPP(t) \geq DQPP(t)$
        THEN $RDO(t) = [DO(t) + DFCPP(t) + DICPP(t) - DQPP(t)] / dt$
        ELSE $RDO(t) = 0$
ELSE $RDO(t) = 0$ (4.30)

$$DBWO(t) = \left[\int_{t_0}^{t} (RDASF(t) - FRCPP(t))dt \right] + DBWO(t_0) \tag{4.31}$$

$$FRCPP(t) = DPR(t) \tag{4.32}$$

$$QOC(t) = DR(t) - FRCPP(t) \tag{4.33}$$

4.3.3 Estimation of Parameters and Initial Conditions

A summary of estimated parameters and initial conditions is presented in this section. Parameters are the exogenous variables identified during the mapping effort. They serve as constants during the simulation, whereas the initial conditions correspond to the initial values of the stocks in the model. **Table 4.2** displays the parameters.

Table 4.2 Parameter Estimates.

Parameter	Estimated [unit]
$NSDDBS$: number of standard deviations of demand rate used for the calculation of buffer stock	$NSDDBS = 2$
$EUPS$: expected upstream process shortfall	$EUPS = 0$
$SDDPEP$: standard deviation of demand rate in previous estimation period	$SDDPEP = 0.1$ [piece/min]
$PUPCF$: protective upstream production capacity factor	$PUPCF = 1$
$PDPCF$: protective downstream production capacity factor	$PDPCF = 1$
$ADRPEP$: average demand rate in previous estimation period	$ADRPEP = 1$ [piece/min]
SPP : size of planning period	$SPP = 480$ [min]
DR : demand rate	$DR = 1$ [piece/min]
KS : kanban size	$KS = 10$ [pieces/kanban]
$onekanban$	$onekanban = 1$ [kanban]
$NKpitch$: number of kanbans in a pitch	$pitch = 1$
t : time	t : computer clock time [min]
dt : time step	$dt = 0.03125$ [min]

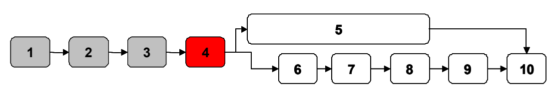
Table 4.3 displays the initial conditions. They establish the state of the system at the beginning of the simulation. They are set in such a way as to attempt to start the simulation in balanced equilibrium. As Sterman (2000) emphasizes, model testing should be a process of controlled experimentation and for this reason simulations ideally should be initialized in equilibrium, meaning that that all stocks in the system are unchanging and equal to their desired values.

Initializing the model in balanced equilibrium facilitates the process of model testing because the system remains in equilibrium until disturbed by test inputs. If the model begins out of equilibrium its behavior will confound the response to any test input with the transient behavior induced by the initial disequilibrium. It should be noted, however, that not all models possess a unique balanced equilibrium or any balanced equilibrium at all.

Table 4.3 Initial Conditions.

Stock	Initial Value
$PDI(t)$: Protective decoupling inventory	$PDI(t_0) = CPDI(t_0) + BPDI(t_0) + SPDI(t_0)$
$DFCPP(t)$: Demand to be fulfilled in current PP	$DFCPP(t_0) = 0$
$DICPP(t)$: Demand incurred in current PP	$DICPP(t_0) = ADPEP \cdot ATPP$
$DO(t)$: Demand overflow	$DO(t_0) = 0$
$DBWO(t)$: demand being worked on	$DBWO(t_0) = 0$
$IUDP(t)$: items in use at downstream process	$IUDP(t_0) = 0$
$IUUP(t)$: items in use at upstream process	$IUUP(t_0) = 0$
$HB(t)$: heijunka box	$HB(t_0) = 0$
$APDIK(t)$: assigned PDI kanbans	$APDIK(t_0) = 0$
$UCUP(t)$: units completed by upstream process	$UCUP(t_0) = pitch$
$QOC(t)$: quantity on order from external customer	$QOC(t_0) = ADPEP \cdot ATPP$

Note that, in this hypothetical case, the initial conditions are specified in terms of other variables and parameters, not as numerical values. This way the model will be initialized in dynamic equilibrium for any set of parameters and inputs.



4.4. Model Testing

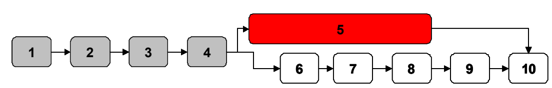
Model testing is a process of building confidence in a model. Forrester (1961) defends that any objective model validation procedure is eventually based on some level of judgment or faith that either the procedure or its goals are acceptable without objective truth. This view is shared by many other authors including Oliva (2001)and Sterman (2000). Below is an excerpt from Sterman's *Business Dynamics* regarding this subject:

"The word validation should be struck from the vocabulary of modelers. All models are wrong, so no models are valid or verifiable in the sense of establishing their truth. The question facing clients and modelers is never whether a model is true

but whether it is useful. The choice is never whether to use a model. The only choice is which model to use. Selecting the most appropriate model is always a value of judgment to be made by reference to the purpose. Without a clear understanding of the purpose for which the model is to be used, it is impossible to determine whether you should use it as a basis for action.

Models rarely fail because the modelers used the wrong regression technique or because the model didn't fit the historical data well enough. Models fail because more basic questions about the suitability of the model to the purpose aren't asked, because the model violates basic physical laws such as conservation of matter, because narrow boundary cut critical feedbacks, because modelers kept the assumptions hidden from the clients, or because the modelers failed to include important stakeholders in the modeling process (Sterman, 2000 p.890)."

The model tests were conducted according to the testing guidelines suggested by Sterman (2000) and adapted to a hypothetical case. For a detailed description of the adopted testing procedures and results, please refer to **Appendix C**.



4.5 System Evaluation and Redesign

In order to specify potential alternative system structure(s), the problem as presented so far is rewritten below as a description of an opportunity for improvement:

Enhancing the system's on-time-delivery robustness in face of demand represents an improvement.

As a general guideline, the characteristics of an ideal solution are defined as simplicity and ease of implementation and use. The key findings obtained from the literature review effort in terms of how to increase robustness of a system under the circumstances of interest in this research are converted into redesign insights in **Table 4.4**.

Table 4.4 Redesign Insights.

Insights
Reduction of time delays can increase the robustness of the system (Towill, 1996a; Sterman, 2000)
A control process based on system backlog is generally more robust than the alternatives in the sense that adequate performance is achieved over a broader range of parameters (Powell et al., 2001)

4.5.1 Policy Design

Three policies were created as potential changes to the traditional value stream structure:

- Policy 1: decision rule for changing the number of kanbans circulating in PDI;
- Policy 2: decision rule for when and how to use protective inventory to absorb demand surges; and
- Policy 3: decision rule on how to determine the system's protective capacity.

These policies are the result of a creative design effort that combined trial and error with insights from the literature. Among the various alternative redesign options explored, these three were selected according to the author's intuitive expectation that they could indeed promote value stream performance improvement. Each one of these policies is related to a major feedback loop in the value stream description (see **Figure 4.19)**. These loops are believed to be the main loops determining the dynamics of value streams. The loops related to Policy 1 and Policy 2 are endogenous to the model developed in this research. Given the assumption of fixed production capacity, a production capacity adjustment loop is not captured in the model. Accordingly, the production capacity level set by Policy 3 remains unchanged during simulations. A description of each proposed policy follows.

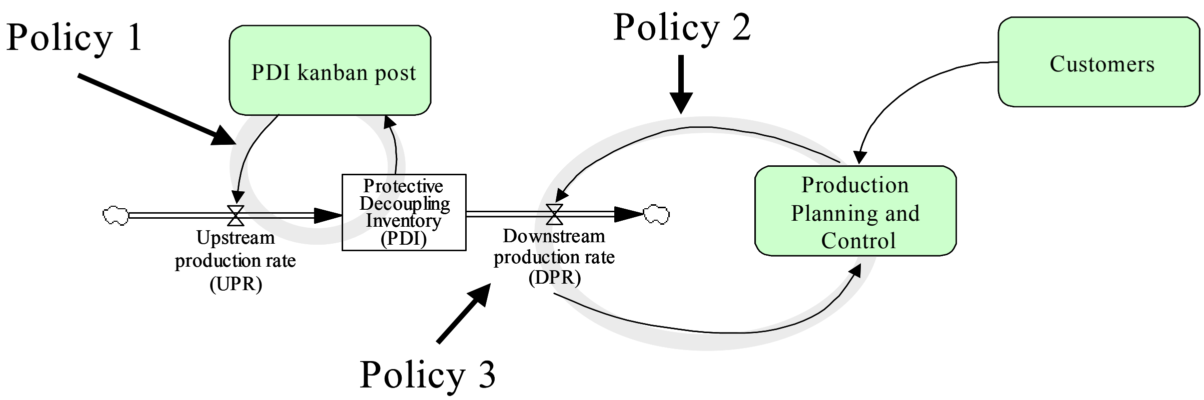


Figure 4.19 *Proposed Policies and Associated Feedback Loops.*

4.5.1.1 Policy 1: Decision Rule for Changing the Number of Kanbans Circulating in PDI

In traditional value streams, the number of kanbans circulating in supermarkets remains fixed during production cycles. Policy 1 reflects a decision rule to alter the number of kanbans in circulation at the PDI supermarket, at the beginning of each PP based on the *Demand Overflow (DO)* level.

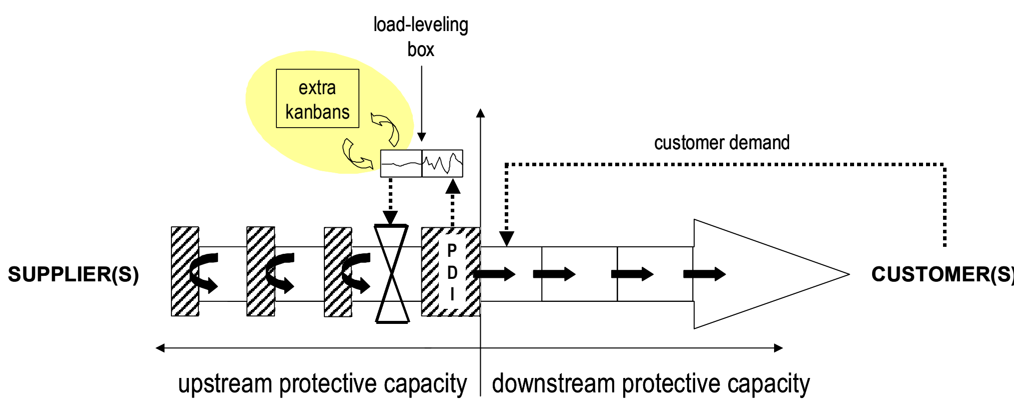


Figure 4.20 *Schematic representation of Policy 1.*

The direct impact of this policy is the alteration of maximum limit for WIP at the PDI. These alterations in WIP ranges could case the PDI to inflate/deflate according to demand conditions. See **Figure 4.20**. Ideally, the number of kanbans would be manipulated in such a way that PDI would inflate prior to demand shocks in order to absorb it. However, no advance notice on the demand rate is being considered in this study. However, manipulating the number of kanbans even after the demand shock was perceived will impact upstream production. The potential benefit of Policy 1 under this condition is not obvious. **Table 4.5** describes Policy 1 rationale.

Table 4.5 *Description of Policy 1.*

Policy 1 rationale	Parameters
At the beginning of each planning period: IF DO > 0 THEN IF NKC < MAXNKC THEN NKA = MAXNKC-NKC ELSE NKA = 0 ELSE IF DO = 0 THEN IF NKC > MINNKC THEN NKR = NKC-MINNKC ELSE NKR = 0	MINNKC = the base number of kanbans at the beginning of the simulation; set according to initial PDI level divided by kanban size. MAXNKC = 2MINNKC

NKC: number of kanbans in circulation
NKA: number of kanbans to add
NKR: number of kanbans to remove
MAXNKC: maximum number of kanbans in circulation
MINNKC: minimum number of kanbans in circulation

4.5.1.2 Policy 2: Decision Rule for When to Use Protective Inventory to Absorb Demand Surges

Policy 2 attempts to ensure adequate use of the PDI as a buffer. It acts as a first filter of demand instability, even prior to the load-leveling box. It was created based on the need identified by

Smalley (2004) for the formalization of a decision rule to systematically allocate demand quantities to the value stream in unstable situations. See **Figure 4.21**.

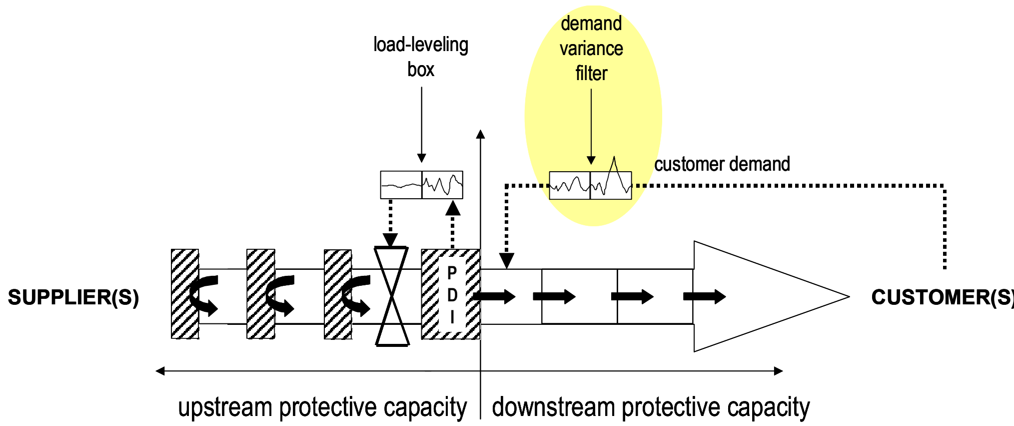


Figure 4.21 *Schematic representation of Policy 2.*

Under normal demand circumstances it attempts to leave the safety stock component of PDI untouched and only use the buffer stock component. Under spike conditions (i.e., when demand incurred in a PP goes beyond the historical average plus 3 standard deviations), the entirely available inventory can be used. A spike is interpreted as a very unusual event when it is worth the risk of using the entirely available PDI. **Table 4.6** describes Policy 2 in detail.

Table 4.6 *Description of Policy 2.*

Policy II rationale	**Parameters**
At the beginning of each planning period: IF DIPP(t)+DO(t)+DFPP(t) < ADPEP+1SDD THEN allocate entire DIPP(t)+DO(t)+DFPP(t) to shop floor ELSE IF DIPP(t)+DO(t)+DFPP(t) >= ADPEP+1SDD AND DIPP(t)+DO(t)+DFPP(t) <= ADPEP+3DD THEN allocate demand to shop floor up to what can be produced in a PP on average plus BS ELSE allocate demand to shop floor up to what can be produced in a PP on average plus the entirely available PDI level	None

DIPP: demand incurred in previous planning period
DO: current quantity in demand overflow
DFPP: any remaining demand supposed to have been fulfilled in the previous planning period
ADPEP: historical average demand incurred in a planning period
SDD: historical standard deviation of demand in a planning period
PP: planning period
BS: calculated buffer stock
PDI: entire protective decoupling inventory available

4.5.1.3 Policy 3: Decision Rule on How to Determine the System's Protective Capacity

The guidelines for designing traditional value streams suggest that production capacity should be balanced all across the system, i.e., all processes should be able to operate at or slightly below the desired cycle time (i.e., takt time). Policy 3 suggests that there might be benefits in setting upstream and downstream capacity at different levels. In particular it attempts to increase value stream flexibility by adding extra capacity downstream (see **Figure 4.22**).

Insight for the design of policy 3 was obtained from literature discussion on the integration of lean and agile manufacturing. The literature emphasizes that the use of protective capacity, and not only protective inventory, is central to the agile supply paradigm (Naylor, Naim & Berry, 1999; Hoek, 2000; Mason-Jones et al., 2000; Stratton & Warburton, 2003).

Policy 3 is also aligned with the concepts advocated by the theory of constraints (TOC). TOC suggests that work should be organized for efficiency at the constraint workstation (the upstream process in this case) while reducing flow time to the customer in non-constraints (the downstream process in this case) (Blackstone & Cox, 2002).

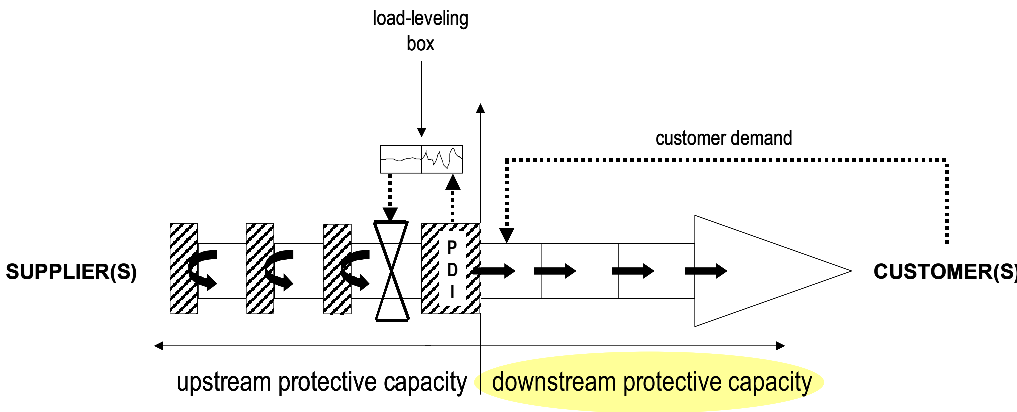


Figure 4.22 *Schematic representation of Policy 3.*

Downstream production capacity (DPC) is defined as the summation of a base capacity (corresponding to the historical average demand rate) plus a protective factor (which can be measured as a certain number of historical standard deviations of demand rate). The assumption adopted here is that traditional value streams allocate just about the right amount of capacity needed to fulfill average historical demand, i.e., the protective capacity is defined as:

$$PUPCF = \alpha \cdot SDD = 1 \cdot SDD \tag{4.34}$$

$$PDPCF = \beta \cdot SDD = 1 \cdot SDD \tag{4.35}$$

Policy 3 suggests increasing the downstream protective capacity factor to 3 standard deviations of demand rate. The upstream protective capacity remains the same and therefore the system's overall capacity remains the same; except that the combination of extra downstream capacity and the protective decoupling inventory size will make it possible for temporary accelerations of the system's production rate. **Table 4.7** describes policy 3.

Table 4.7 Description of Policy 3.

Policy 3 rationale	Parameters
At the beginning of each estimation period: Set $PUPCF = \alpha \cdot SDD$ Set $PDPCF = \beta \cdot SDD$	$\alpha = 1$ $\beta = 3$

Note that policy 3 provides a suggestion for the values of α and β. The specific α and β possible in a real case scenario will depend on the value stream's "production capacity playbook," i.e., the specific values possible for α and β will be determined by the physical specificities of the process in question. **Table 4.8** describes the implications of the possible $\frac{\beta}{\alpha}$ ratios. This ratio is here referred to as flexibility factor (F).

Table 4.8 Description of the flexibility factor.

Flexibility Factor F ($F = \frac{\beta}{\alpha}$)	Description
$F = 1$	No flexibility effect. Balanced system. Time to use PDI inventory is the same as time to build inventory. Protective capacity, if any, is kept all across the system.
$F > 1$	Active flexibility effect. System can temporarily speed up downstream production to absorb demand surges. Time to use inventory is smaller than time to build inventory. Time to build inventory determines how frequent system can absorb demand spikes. Protective capacity is concentrated on downstream processes, enabling better resource utilization upstream.
$F < 1$	Inactive flexibility effect. Undesirable. No ability to speed up downstream production rate to absorb demand surges.

Considering the case when F>1, the demand rate that can normally be absorbed by the value stream is *upstream production capacity (UPC)*. If necessary, while *Protective decoupling inventory (PDI)* exists, the downstream rate can be accelerated up to *downstream production* *capacity (DPC)*. The overall value stream rate will return to UPC once PDI is consumed. The size of demand spike that can be absorbed depends on available PDI at the time of spike occurrence. In case there is enough PDI, the time to absorb the spike will be a function of DPC.

Complementarily, the time to replenish the PDI will be a function of UPC in face of the incurred demand profile. During and after a spike, the upstream processes will be requested to refill the PDI in a leveled manner (via the load-leveling box). The precise time to restitute the buffer stock depends on the net difference between UPR and DPR, which again will depend on the demand pattern incurred after the spike.

4.5.2 "What If…" Analysis

Each one of these proposed policies and related structures constitute value stream redesign alternatives. They can be incorporated either individually or in combination. Evaluation of all possible combinations was performed via simulation in a hypothetical numerical scenario. The baseline and all possible redesign combinations were subjected to the same demand spike. The results are portrayed in **Table 4.9**. Findings suggest synergy between proposed policies. The combination of policies 1, 2, and 3 (i.e., P123) presented the best results.

The baseline value streams, as well as the alternative value stream structures were subjected to the same pattern of demand rate. Simulation starts with demand rate in stasis at 1 piece per minute. At time 2500[min] a spike is initiated in the form of a RAMP function with slope +0.2 and proceeds until time 2540[min]. From time 2540[min] to time 2580[min] the RAMP takes a slope of –0.2 causing the demand rate to return to the original stasis level. The total amount of extra demand incurred equals 320 [units], which is equivalent to approximately 6 historical standard deviations of demand incurred in a planning period.

Table 4.9 *Ranking of Redesign Alternatives.*

Ranking	Combination	Total QOC (TQOC) (Difference relative to baseline)	Recovery Time (RT) (Difference relative to baseline)
1st	P123	-6.3%	-23.6%
2nd	P23	-6.0%	-22.5%
3rd	P12	-4.6%	-21.4%
3rd	P2	-4.6%	-21.4%
4th	P13	-2.7%	-5.4%
5th	P3	-1.5%	-2.1%
6th	P1	0.0%	0.0%

Figure 4.23 focuses on the *Quantity on order from external customer (QOC)* variable. The baseline *QOC* graph is included, as well as a *QOC* graph corresponding to each one of the possible policy configurations (i.e., P1, P2, P3, P12, P13, P23, and P123). Note that different configurations provide different recovery patterns. Given the assumption that all demand is due immediately, *QOC* can be interpreted as the backlog.

As shown in **Table 4.9** the combination P123 presented the best result in terms of total backlog quantity, followed closely by P23. This ranking was calculated in the following manner. First the *Total QOC (TQOC)* quantity incurred in the baseline scenario was calculated, i.e., the area under the baseline *QOC* curve. Next the same was done for each of the redesign alternatives. If less *TQOC* as compared to the baseline is obtained, then it indicates improvement. The relative results against the baseline are provided in the third column of **Table 4.9**.

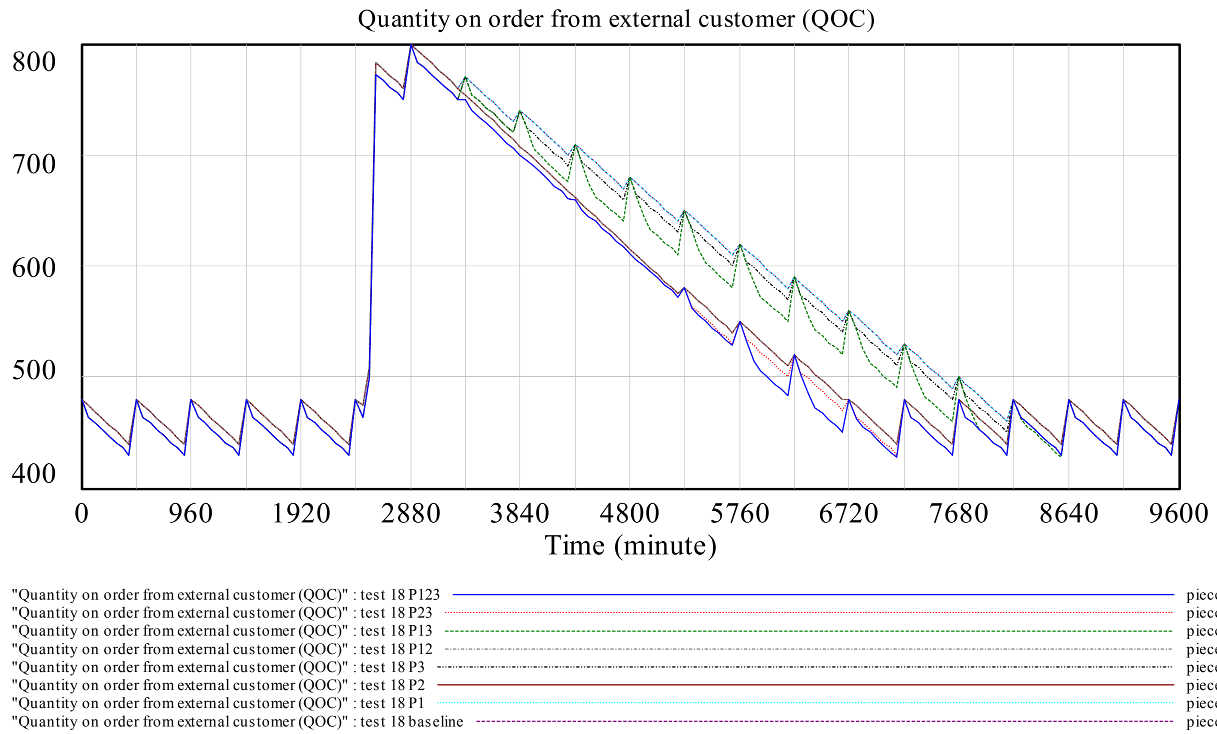


Figure 4.23 *Backlog Behavior Comparison Among Potential Scenarios.*

Complementarily, the *recovery time (RT)* was calculated in a similar manner. *RT* indicates the time from the spike occurrence until the system returns to equilibrium conditions (as indicated by *QOC*). The relative improvement in relation to the baseline is provided in the fourth column of **Table 4.9**. Furthermore, an additional perspective of the recovery performance is illustrated in **Figure 4.24**. It shows the absolute difference in *QOC* in relation to the baseline over time.

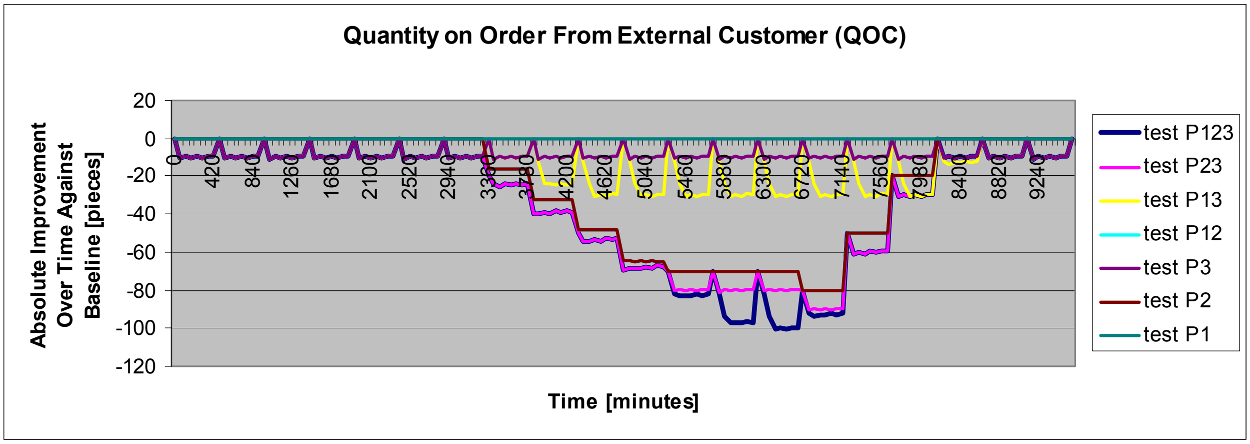


Figure 4.24 *Absolute difference in QOC in Relation to Baseline.*

4.5.3 Sensitivity Analysis

In this hypothetical example, the combination P123 shows the best results. Based on this finding. **Figure 4.25** shows the results of a sensitivity analysis of *QOC* given the incorporation of noise in the demand and production rates. Noise was inserted in these variables by assuming a normal distribution for demand rate and uniform distribution for processing rates with the following parameters:

- *Demand rate*: minimum value = 0.5 [pieces/min], maximum value = 1.5 [pieces/min], mean = 1 [piece/minute], and standard deviation = 0.1 [piece/minute];
- *Upstream production capacity noise*: minimum value = -0.1 [piece/min], and maximum value = 0.1 [piece/min];
- *Downstream production capacity noise*: minimum value = -0.1 [piece/min], and maximum value = 0.1 [piece/min].

Two hundred replications of the simulation experiment were conducted, each one with different random number seeds. The various regions in represent different ranges of confidence in the results obtained. The solid line represents the central tendency in the P123 scenario, and the dashed line represents the central tendency in the baseline scenario.

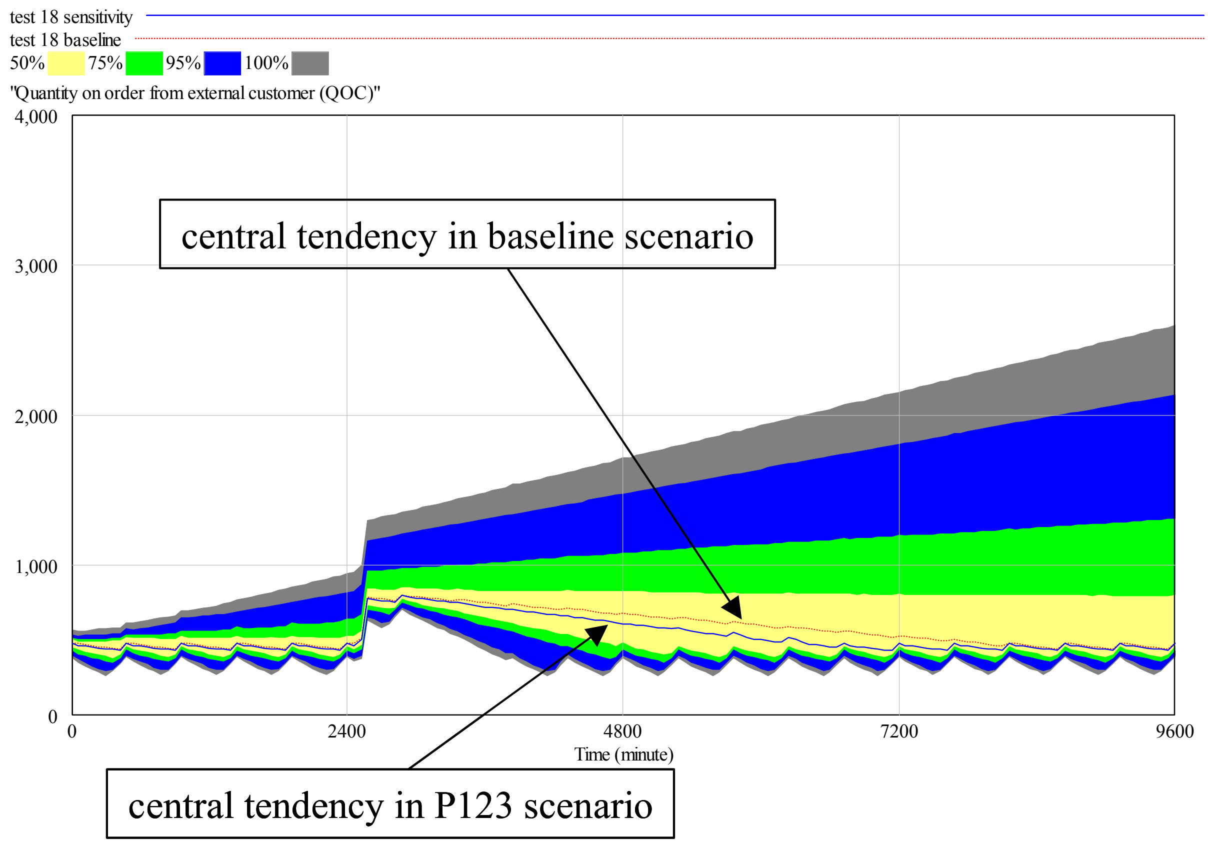


Figure 4.25 *Sensitivity Analysis of P123 under Demand Spike.*

Numerical Sensitivity: Do the numerical values change significantly?

As expected, the further in the future the simulation runs the wider the confidence interval gets. There is small chance that the system will not recover from the demand spike. On the other hand, note that there is a minimum limit for *QOC* values; this limit corresponds to the best possible performance of this system given its structure and production capacity configuration.

4.5.4 Discussion of Results

These results point to the conclusion that scenarios involving the combination of Policies 2 and 3 are capable of providing significant improvement in value stream performance. Furthermore, Policy 1 may slightly enhance the performance of the other scenarios by reducing blocking of the downstream process. A detailed discussion of these results follows.

4.5.4.1 P1

The adoption of Policy 1 alone has provided no performance improvement benefits. One purpose of Policy 1 is to increase the *PDI* level so that no starvation of the downstream process takes place during unstable demand conditions. As illustrated in **Figure 4.26**, P1 does promote increase in the *PDI* level. However, under the balanced upstream and downstream production capacity conditions in question, this increment is unnecessary to ensure uninterrupted flow across the value stream.

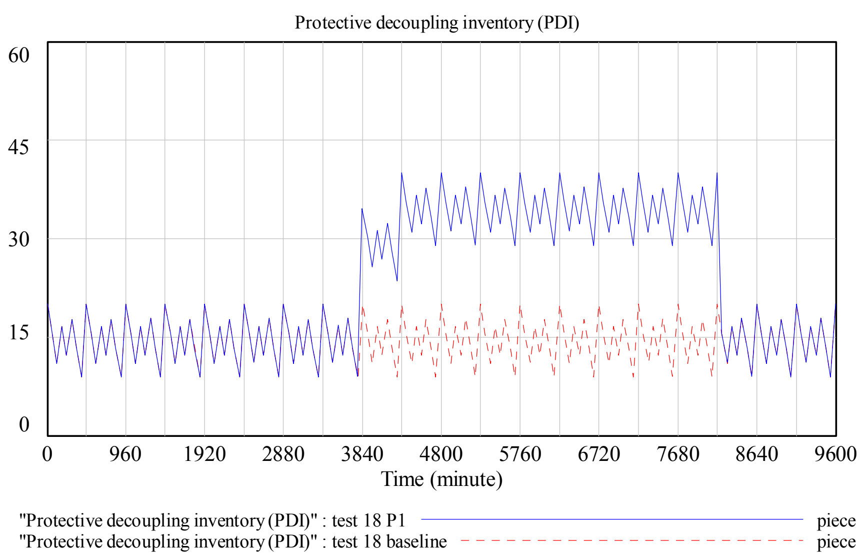


Figure 4.26 *PDI Level: Baseline vs. P1.*

4.5.4.2 P3

Policy 3 is able to slightly increase the delivery performance because even though the same amount of demand is allocated to value stream, the downstream process is able to process it faster and therefore the *QOC* curve does not grow as much as in P1.

4.5.4.3 P13

Policy 1 enhances the performance of Policy 3 by reducing the number of times the downstream process is blocked because of lack of *PDI* inventory. Consequently, under P13 the downstream process can fulfill demand quantity allocated to the floor even faster than under P3 alone.

4.5.4.4 P2

Policy 2 was designed to provide support for the decision of how much demand to allocate to the floor at the beginning of each planning period (PP). It allows normal variation to go through and be absorbed by the buffer stock. Under spike conditions it also allows the use of all safety stock available.

Even though under P2 there is no change in processing capacity, the additional demand allocated to the floor promotes better capacity utilization. P2 may actually increase the number of kanbans processed within a PP. This is so because the extra demand quantity provided may round the equivalent number of kanbans to the next integer value. Therefore, by completing the "incomplete" kanbans, not only an additional kanban can be processed in the current PP but also a higher demand quantity can be allocated to the next PP.

The kanban size interferes in the effect of P2. In this particular hypothetical scenario, the kanban size is 10 [pieces]. As the kanban sizes diminish and approach the "one-piece-flow" condition, the effect of Policy 2 is expected to be neutralized.

4.5.4.5 P12

The addition of Policy 1 to Policy 2 provides no extra benefits. P2 is already promoting the utilization of virtually the entire available value stream capacity. Consequently, as illustrated in **Figure 4.27**, P12 is capable of increasing the *PDI* size only after significant time beyond the spike occurrence has passed – when this additional inventory is no longer necessary.

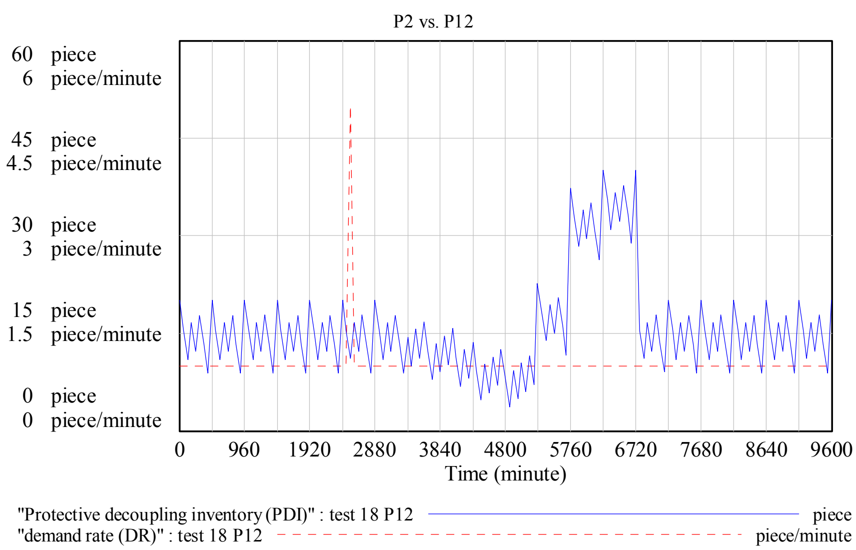


Figure 4.27 *PDI Level vs. Demand Rate under P12.*

4.5.4.6 P23

Adding Policy 3 to Policy 2 provides significant benefits. In addition to the impact already made by P2 as the number of incomplete kanbans is reduced, the extra downstream processing capacity promoted by P23 can now enable the processing of a larger portion of the demand sent to the floor by Policy 2 (see **Figure 4.28**). The higher the additional capacity enabled by Policy 3, the greater the benefits; however only up to a certain point. Beyond that any additional benefits will also require increasing the capacity of the upstream process.

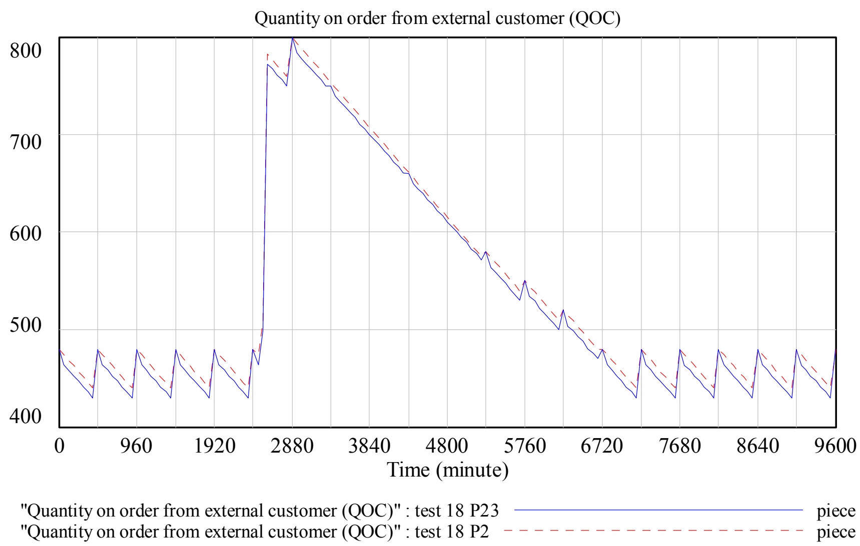


Figure 4.28 *QOC Level: P2 vs. P23.*

4.5.4.7 P123

The results suggest there is a minor benefit of adding Policy 1 to P23. By doing so, less blocking of the downstream process occurs when the system has almost returned to dynamic equilibrium (see **Figure 4.29**). Consequently, the value stream can process a little more and recover slightly faster.

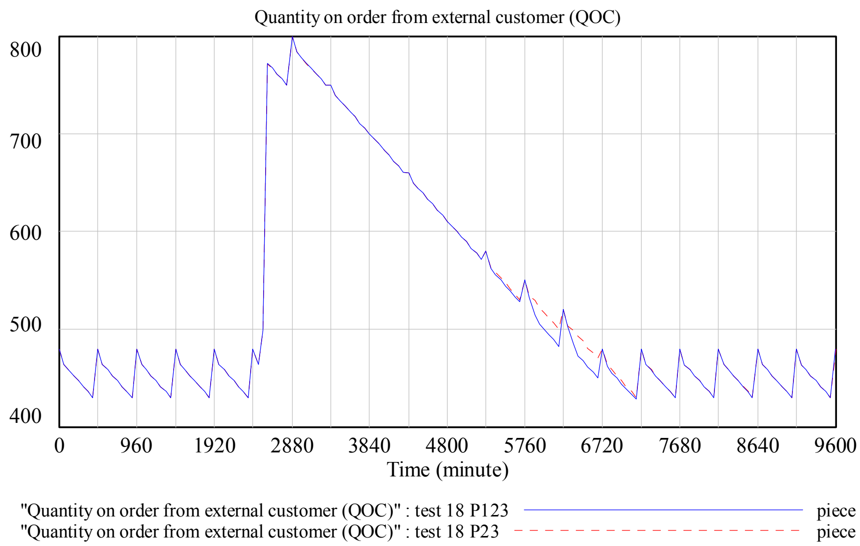


Figure 4.29 *QOC Level: P23 vs. P123.*

Chapter 5
Real Case Application and Results

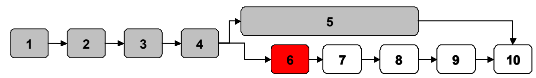
This chapter presents a real case study performed at a manufacturing enterprise. It contextualizes the main generic modeling elements developed in **Chapter 4** in a real world situation, thus serving as an illustrative example. The overall effort is also guided by the system dynamics modeling methodology presented previously, and therefore follows the same sequence of steps as the ones developed for the hypothetical case. These steps are indicated in the research process described in **Figure 3.1**:

6. Problem articulation
7 Formulation of a dynamic hypothesis
8. Formulation of a simulation model
9. Model testing
10. System redesign and evaluation

The real case has been conducted in a manufacturing enterprise specialized in motion control solutions. The product in question consists of a family of servo drives. Servo drives provide electrical drive output to servo motors in closed-loop motion control systems, where position feedback and corrective signals optimize position and speed accuracy (GlobalSpec, 2004).

These drives are required in all types of motion control applications. Intelligent motion control systems applied to computer numerical control (CNC), robotics, and general motion control systems all require high performance servo drives to provide the interface to servomotors (ARC, 2003).

5.1 Problem Articulation



The target enterprise's business philosophy is grounded on lean production principles and emphasizes four core customer-oriented priorities: quality, delivery, cost, and innovation. Significant results have been achieved over the years in many areas. Nevertheless, despite valuable lessons learned, there is still difficulty in maintaining high delivery performance for this particular family of servo drives.

5.1.1 Theme Selection

Demand for these servo drives has historically been unstable with occasional peaks. Production takes place according to lean production guidelines in a manner that incorporates many of the key value stream elements. In special, production takes place only as a result of actual customer requests and not to a forecast. The overall problematic theme is the same as the one presented in **Chapter 4**.

5.1.2 Key Variables and Concepts

Key variables:

- Demand rate
- Production rate
- Backlog
- Production capacity

Key concepts:

- Production pull
- Kanban

5.1.3 Time Horizon

The time horizon relevant for this case is in the order of three months or so. This is the magnitude of time within which the system has been presenting problematic cycles of significant backlog growth followed by recovery.

5.1.4 Dynamic Problem Definition

Figure 5.1 describes the problem by means of graphs of behavior over time using historical performance data. Three key variables are used to define the problem: *demand rate*, *production rate,* and *backlog*. *Demand rate* is unstable, presenting a few significant peaks in the time frame captured. Note how *backlog* increases abruptly after the occurrence of demand peaks, as well as how significant the recovery time is.

The *production rate* data (available only since February of 2004) suggests the system remains relatively insensitive to the demand peaks. Three cycles of backlog growth and recovery are portrayed in this timeframe. The first two appear to be mostly driven by the demand pattern, i.e., by the customer behavior. The third one, however, is strongly determined by the supplier's behavior, i.e., a temporary interruption in the supply of a critical part.

Looking back at this historical performance, some of the demand peaks might be classified as spikes according to the definition developed in **Chapter 4** (i.e., a daily demand quantity that surpasses average daily demand plus three standard deviations of daily demand). For illustrative purposes, using this historical period itself as a reference for average demand and standard deviation, the spike definition line would be approximately 89 pieces per day.

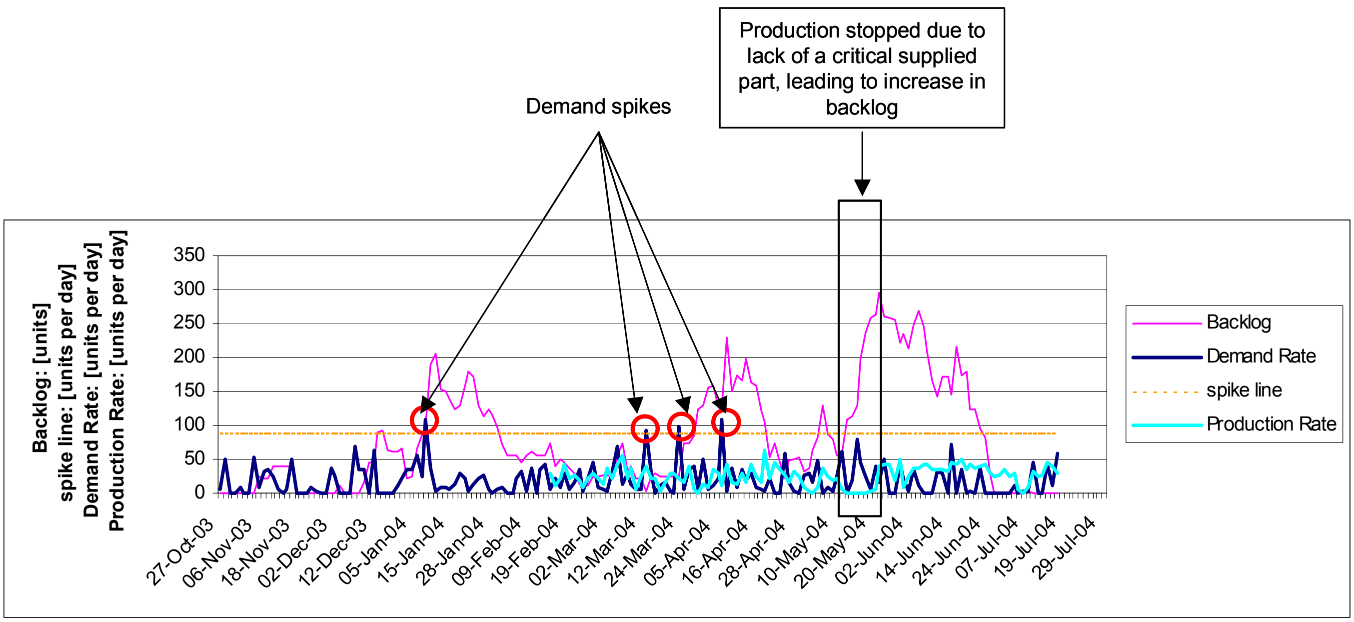
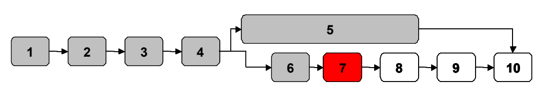


Figure 5.1 *Dynamic Problem Definition.*

5.2 Formulation of a Dynamic Hypothesis



The dynamic hypothesis discussion presented in **Chapter 4** remains valid for this real case.

5.2.1 Mapping

This section presents the effort of formalizing the representation of causality characteristics in the real case. The hypothetical map developed in **Chapter 4** is used as a reference and most of it remains valid.

5.2.1.1 Scope and Level of Aggregation

The scope and level of aggregation are the same as for the hypothetical case, i.e., the focus relies on a door-to-door value stream dedicated to the production of a single family of products (see **Figure 5.2**). Just like in the hypothetical case, product variants are not distinguished here either.

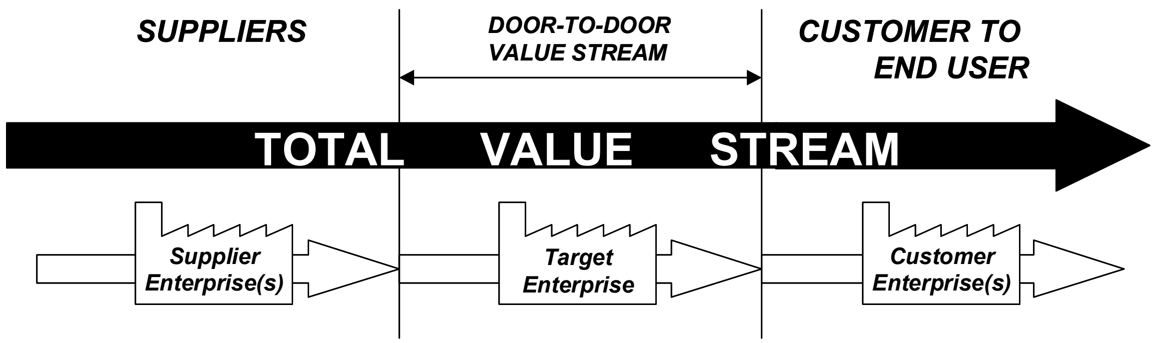


Figure 5.2 *System Scope.*

5.2.1.2 Structure of Target Door-to-Door Value Stream

Figure 5.3 presents a schematic value stream map of the target system. Some key processing steps in this value stream are: surface mount, card assembly, and final assembly. There are supermarkets immediately before final assembly as well as before the combination of surface mount plus card assembly.

The master scheduling function collects orders from customers. In some particular cases it may attempt to postpone requested due dates in an effort to smooth out significant demand peaks. This effort has very limited impact, as is evidenced by the unstable demand rate curve in **Figure 5.1**.

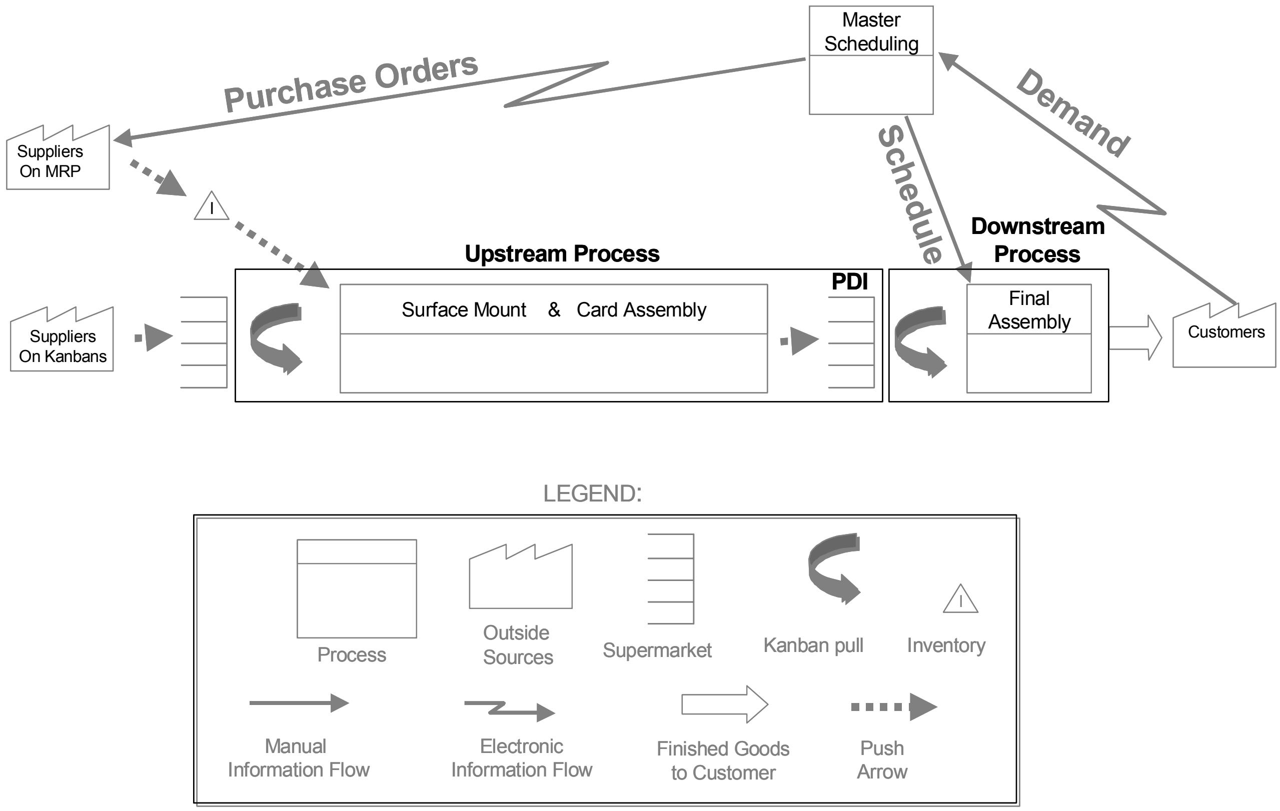


Figure 5.3 *Value Stream Map of Target System.*

From that point the demand signal is sent directly to final assembly without any leveling treatment. This strategy reflects higher management directives to let the system be as much as possible driven by the "voice of the customer". As such, schedules are generated daily and consist simply of the number of parts due on that day as requested by the customers, independent of the system's ability to actually fulfill it. This demand signal received at the final assembly then propagates upstream via the kanban systems, all the way up to external suppliers. Only a few non-critical lower level items are purchased via an MRP mechanism and arrive directly into a raw parts inventory.

5.2.1.3 Policy Diagram

In translating this value stream into system dynamics representation, there are two main balancing feedback loops of interest. The first one is the loop involving the PPC subsystem and the downstream process: as *backlog* increases, the required *demand rate* provided to the downstream process in a planning cycle increases, the *downstream production rate* (up to a certain point) increases, the *backlog* decreases, leading to a decrease in the required *demand rate* in the next planning cycle.

The second loop is the one involving the kanban subsystem and the upstream process: as the *PDI* level decreases, the *PDI* kanban post quantity increases, the upstream production rate increases (up to a certain point), leading to an increase in the *PDI* level. **Figure 5.4** displays the macro structure of the system via a policy diagram.

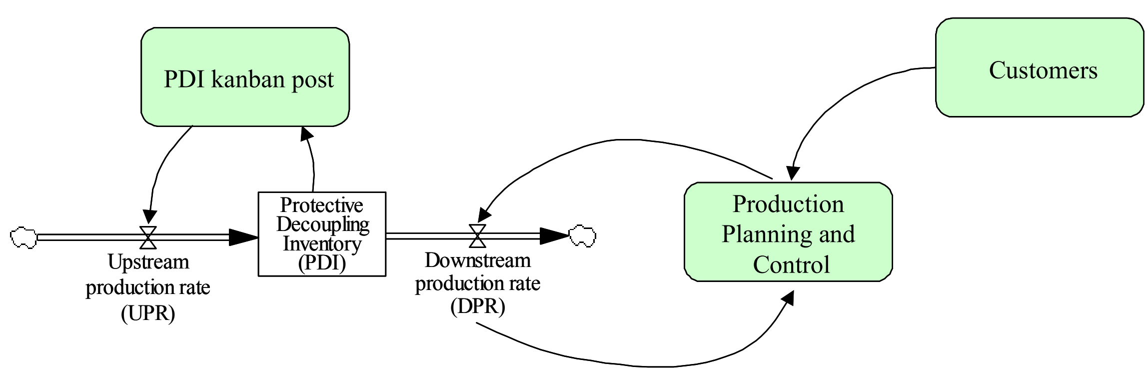


Figure 5.4 *Policy Diagram.*

5.2.1.4 Causal Loop Diagram with Stocks and Flows

Using this policy diagram as a macro reference, the causal loop diagram with stocks and flows was generated. In fact, it constitutes a customization of the generic hypothetical case model described in **Chapter 4**. Some key points in the translation of the value stream map into the system dynamics notation are addressed in **Table 5.1**.

The hypothetical model is mostly applicable to represent the real case. However, some modifications were made to reflect the target system's current structure. The most significant one relates to the fact that, in the real case, the daily production schedules are equivalent to the actual demand incurred, i.e., the master scheduler does not allocate demand to the shop floor based on an estimate of system capacity.

Table 5.1 *Main Value Stream Structural Elements.*

Structural Element	Identification	Explanation
Protective decoupling inventory (PDI)	The supermarket before final assembly	This is the last supermarket location closest to the external customer; therefore it acts as decoupling inventory.
Downstream process	Final assembly	The process downstream from the PDI (shipping is considered part of this process).
Upstream process	Surface mount and Card assembly	The critical processing steps in the internal supply chain upstream from the PDI.
Scheduling mechanism	Master scheduling	The PP&C subsystem provides final assembly with a daily production schedule.
Load leveling mechanism	None	There is no load leveling mechanism to smooth out the demand signal from the downstream to the upstream process.
Pacemaker	Final assembly	Given that the schedule is sent to final assembly, it is this process the one that attempts to set the pace of the entire value stream.

5.2.1.5 Assumptions

Additionally most of the hypothetical case's modeling assumptions have been maintained in the real case, except for assumptions 7 and 8 (see **Table 5.2** and **Table 5.3**).

Table 5.2 *Maintained Assumptions.*

Assumption	Description
Assumption 1	*The base capacity is set according to the average historical customer demand observed during an estimation period.*
Assumption 2	*The supplier value stream production capacity equals to upstream production capacity (UPC). Supply availability is unlimited but maximum supply rate equals to UPC.*
Assumption 3	*Value stream production capacity is adjusted only at the beginning of each estimation period.*
Assumption 4	*The performance of the system does not significantly affect the demand rate, i.e., for the purpose of this study, it is appropriate to model demand rate as an exogenous variable.*
Assumption 5	*The delivery lead-time has already been negotiated with the customer; as such, demand quantities accounted in the model are due immediately.*
Assumption 6	*Work does not extend to fulfill all available time, i.e., if half day worth of work is sent to the floor at the beginning of a PP, production takes place at full pace during the first half of the day and then stops, instead of at half pace during the entire PP.*
Assumption 9	*There is no advance notice on demand; all demand incurred is due immediately.*

Table 5.3 *Rejected Assumptions.*

Assumption	Description	Explanation
Assumption 7	*The amount of protective capacity in both upstream and downstream processes equals one standard deviation of demand, as measured in the previous estimation period.*	*Protective capacity in this real case has been estimated based on historical data. It does not necessarily correspond to one standard deviation of demand.*
Assumption 8	*The size of the Protective Decoupling Inventory is set according to the guidelines suggested by Smalley (2004).*	*The target enterprise has its own guidelines for determining the PDI size. The PDI size in use has been identified.*

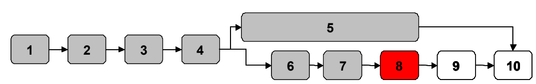
5.2.2 Endogenous Focus

Table 5.4 provides a boundary chart for the real case model.

Table 5.4 Model Boundary Chart.

Endogenous	Exogenous	Excluded
IUUP: Items in use at upstream process	SPP: size of planning period	Demand forecast
UPR: upstream production rate	ADRPEP: average demand rate in previous estimation period	Known demand that is not due immediately
RWPDI: rate of withdrawal from PDI	DR: demand rate	
IUDP: Items in use at downstream process	PUPCF: protective upstream production capacity factor	
PDI: Protective decoupling inventory	PDPCF: protective downstream production capacity factor	
DPR: downstream production rate	one kanban	
RDS: rate of delivery from supplier	SDDPEP: standard deviation of demand rate in previous estimation period	
AKHB: add PDI kanban to heijunka box	time	
HB: Heijunka box	dt: time step	
AKSF: assign kanban to shop floor	NKpitch: Number of kanbans in a pitch	
APDIK: Assigned PDI kanbans	KS: kanban size	
SOF: supplier order fulfillment	DPPD: Downstream process percent downtime	
UCUP: Units completed by upstream process	UPPD: Upstream process percent downtime	
RUCU: Rate of UCUP clean up		
DDPR: desired DPR		
UPC: upstream production capacity		
BUPC: base upstream production capacity		
DPC: downstream production capacity		
BDPC: base downstream production capacity		
EPCS: estimated production capacity of the system		
ADQPP: average demand quantity for a planning period		
ATPP: Available production time in current planning period		

TT: takt time
clock time
DFCPP: Demand to be fulfilled in current planning period
FRCPP: fulfillment rate in current planning period
DO: Demand overflow
RDO: rate of demand overflow
RDOCPP: rate of demand overflow allocation to current planning period
RDAPP: rate of demand allocation to next planning period
DICPP: Demand incurred in current planning period
RDASF: rate of demand allocation to shop floor
DBWO: Demand being worked on
QOC: Quantity on order from external customer



5.3 Formulation of a Simulation Model

Key characteristics of the real case simulation model are described in this section.

5.3.1 Specification of Structure and Decision Rules

As in the hypothetical case, the agents in this value stream include: (1) the value stream manager, and (2) the value stream executors. The role of the value stream manager is simpler as compared to the hypothetical case because no active scheduling or load leveling takes place. At first sight, however, the role of the executors appear slightly more complex from a dynamic point of view because it may go beyond the triggering of production in their respective workstation as a reaction to demand.

Interestingly, it has been part of the company's culture that executors should be given freedom to interfere with the flow of information in the kanban system, under the belief that this practice improves system's performance. Such practice may take place in the following manner: as information (i.e., kanbans) is transported manually between adjacent processes, the number of kanbans provided to the immediate upstream station might be altered. The transporter is allowed to increase the number of kanbans according to his or her perception of an incoming demand spike. This perception seems to be based on informal and qualitative information and the decision is made in an ad-hoc way. Consequently, given its uncertain nature, this practice can be interpreted as the incorporation of noise in the flow of information.

Nonetheless, despite the fact that this practice is appears present in some value streams within the same company, further investigation in the form of interviews revealed that it is seldom utilized in the value stream considered in this study. Therefore, a decision was made to maintain the representation of information flows as a deterministic phenomenon, just like in the hypothetical case. Consequently, in the real case model no alteration in the number of kanbans in transit is allowed.

5.3.2 Model Equations

Similarly to the hypothetical case, a description of the core equation system is provided next. Most equations remain unaltered from the hypothetical case. Equation **5.23** was adapted to reflect the target system's condition of always allocating all demand incurred in a planning period to the shop floor.

$$PDI(t) = \left[\int_{t_0}^{t} (UPR(t) - RWPDI(t))dt \right] + PDI(t_0) \qquad (5.1)$$

$UPR(t):$
IF $IUUP(t) > 0$
THEN $UPR(t) = UPC(t)$
ELSE $UPR(t) = 0$ (5.2)

$$IUUP(t) = \left[\int_{t_0}^{t} (RDS(t) - UPR(t))dt \right] + IUUP(t_0) \tag{5.3}$$

$RDS(t):$
IF $QOS(t) > 0$
THEN $RDS(t) = UPC$
ELSE $RDS(t) = 0$ (5.4)

$$QOS(t) = APDIK(t) \cdot KS \tag{5.5}$$

$$APDIK(t) = \left[\int_{t_0}^{t} (AKSF(t) - SOF(t))dt \right] + APDIK(t_0) \tag{5.6}$$

$$SOF(t) = \frac{RDS(t)}{KS} \tag{5.7}$$

$$UPC(t) = BUPC(t) + PUPCF \cdot SDDPEP \tag{5.8}$$

$$BUPC(t) = ADPEP \tag{5.9}$$

$RWPDI(t):$
IF $DFCPP(t) \geq KS \cdot onekanban$
AND $PDI(t) \geq KS \cdot onekanban$
THEN $RWPDI(t) = (KS \cdot onekanban)/dt$
ELSE $RWPDI(t) = 0$ (5.10)

$$HB(t) = \left[\int_{t_0}^{t} (AKHB(t) - AKSF(t))dt \right] + HB(t_0) \tag{5.11}$$

$AKHB(t):$
IF $RWPDI(t) > 0$
THEN $AKHB(t) = \dfrac{onekanban}{dt}$
ELSE $AKHB(t) = 0$ (5.12)

$AKSF(t)$:
IF ($UCUP(t) \geq pitch$ **AND** $HB(t) \geq 1$ **)**
THEN $AKSF(t) = \dfrac{onekanban}{dt}$
ELSE $AKSF(t) = 0$ (5.13)

$$UCUP(t) = \left[\int_{t_0}^{t} (UPR(t) - RUCU(t))dt \right] + UCUP(t_0) \tag{5.14}$$

$RUCU(t)$:
IF $AKSF(t) <> 0$
THEN $RUCU(t) = \dfrac{pitch}{dt}$
ELSE $RUCU(t) = 0$ (5.15)

$$IUDP(t) = \left[\int_{t_0}^{t} (RWPDI(t) - DPR(t))dt \right] + IUDP(t_0) \tag{5.16}$$

$DPR(t)$:
IF $IUDP(t) > 0$
THEN IF $DPC(t) \geq desiredDPR(t)$
  THEN $DPR(t) = desiredDPR(t)$
  ELSE $DPR(t) = DPC(t)$
ELSE $DPR(t) = 0$ (5.17)

$$DPC(t) = BDPC(t) + PDPCF \cdot SDDPEP \tag{5.18}$$

$$BDPC(t) = ADPEP \tag{5.19}$$

$$desiredDPR(t) = DPC(t) \tag{5.20}$$

$$DICPP(t) = \left[\int_{t_0}^{t} (DR - RDAPP(t) - RDO(t))dt \right] + DICPP(t_0) \tag{5.21}$$

$RDAPP(t) =$
IF MODULO(*clock time, SPP***)**=0
THEN IF $DICCP(t) > 0$
  THEN IF $DFCCP(t) + DO(t) \geq DQPP(t)$
    THEN $RDAPP(t) = 0$

ELSE IF $DICCP(t) \leq DQPP(t) - DFCPP(t) - DO(t)$

THEN $RDAPP(t) = DICPP(t)/dt$

ELSE IF $DICPP(t) > DQPP(t) - DFCPP(t) - DO(t)$

THEN $RDAPP(t) = \left[DQPP(t) - DFCPP(t) - DO(t)\right]/dt$

ELSE $RDAPP(t) = 0$

ELSE $RDAPP(t) = 0$

ELSE $RDAPP(t) = 0$ (5.22)

$$ADQPP(t) = \frac{KS(t) \cdot onekanban}{ESCTKQ(t)} \cdot ATPP(t) \cdot 100 \quad (5.23)$$

$$EPCS(t) = \mathbf{MIN(}UPC(t), DPC(T)\mathbf{)} \quad (5.24)$$

$$ATPP(t) = SPP \quad (5.25)$$

$$DFCPP(t) = \left[\int_{t_0}^{t} (RDAPP(t) + RDOCPP(t) - RDASF(t))dt\right] + DFCPP(t_0) \quad (5.26)$$

$RDOCPP(t) =$

IF MODULO(*clock time, SPP***)**=0

THEN IF $DO(t) > 0$

THEN IF $DO(t) \leq DQPP(t) - DFCPP(t)$

THEN $RDOCPP(t) = DO(t)/dt$

ELSE IF $DO(t) > DQPP(t) - DFCPP(t)$

THEN $RDOCPP(t) = \left[DQPP(t) - DFCPP(t)\right]/dt$

ELSE $RDOCPP(t) = 0$

ELSE $RDOCPP(t) = 0$

ELSE $RDOCPP(t) = 0$ (5.27)

$$RDASF(t) = RWPDI(t) \quad (5.28)$$

$$DO(t) = \left[\int_{t_0}^{t} (RDO(t) - RDOCPP(t))dt\right] + DO(t_0) \quad (5.29)$$

$RDO(t)$:

IF MODULO(*clock time, SPP***)**=0

THEN IF $DO(t) + DFCPP(t) \geq DQPP(t)$

THEN $RDO(t) = DICPP(t)/dt$

ELSE IF $DO(t) + DFCPP(t) + DICPP(t) \geq DQPP(t)$

THEN $RDO(t) = \left[DO(t) + DFCPP(t) + DICPP(t) - DQPP(t)\right]/dt$

ELSE $RDO(t) = 0$

ELSE $RDO(t) = 0$ (5.30)

$$DBWO(t) = \left[\int_{t_0}^{t} (RDASF(t) - FRCPP(t))dt \right] + DBWO(t_0) \tag{5.31}$$

$$FRCPP(t) = DPR(t) \tag{5.32}$$

$$QOC(t) = DR(t) - FRCPP(t) \tag{5.33}$$

5.3.3 Estimation of Parameters and Initial Conditions

Table 5.5 provides a summary of parameter estimates. **Table 5.6** displays the adopted initial conditions. Some additional differences to the hypothetical case model structure may be noted. The variables $NSDDBS$ and $EUPS$ are not applicable in the real case due to the utilization of a different PDI sizing policy. For more details please refer to **Appendix E**.

It should be noted that the parameters related to cycle times were estimated according to measures of actual production rates of the upstream and downstream processes under a condition of demand spike. A historical period was selected where the conditions suggested that both the upstream and downstream processes were operating at maximum rates.

Table 5.5 *Parameter Estimates.*

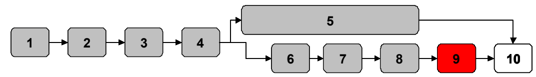
Parameter	Estimated [unit]
$NSDDBS$: number of standard deviations of demand rate used for the calculation of buffer stock	N/A
$EUPS$: expected upstream process shortfall	N/A
$SDDPEP$: standard deviation of demand rate in previous estimation period	$SDDPEP = 0.048$ [piece/min]
$PUPCF$: protective upstream production capacity factor	$PUPCF = 0.40$
$PDPCF$: protective downstream production capacity factor	$PDPCF = 0.21$
$ADRPEP$: average demand rate in previous estimation period	$ADRPEP = 0.041$ [piece/min]

SPP : size of planning period	$SPP = 480$ [min]
DR : demand rate	See historical demand profile [piece/min]
KS : kanban size	$KS = 6.875$ [piece/kanban]
$onekanban$	$onekanban = 1$ [kanban]
$NKpitch$: number of kanbans in a pitch	$NKpitch = 1$ [kanban]
t : time	t : computer clock time [min]
dt : time step	$dt = 0.03125$ [min]
$UPPD$: upstream process percent downtime	$UPPD = 0.2$
$UPCN$: upstream production capacity noise	$UPCN = 0$ [piece/min]
$DPPD$ downstream process percent downtime	$DPPD = 0.2$
$DPCN$: downstream production capacity noise	$DPCN = 0$ [piece/min]
INITIAL TIME: initial simulation time	INITIAL TIME = 0 [min]
FINAL TIME: final simulation time	FINAL TIME = 29280 [min]
SAVEPER: save results every SAVEPER time period	SAVEPER = 120 [min]
$IPDIRC$: initial PDI level in real case	$IPDIRC = 55$ [pieces]

Table 5.6 *Initial Condition.*

Stock	Initial Value
$PDI(t)$: Protective decoupling inventory	$PDI(t_0) = IPDIRC$
$DFCPP(t)$: Demand to be fulfilled in current PP	$DFCPP(t_0) = 0$
$DICPP(t)$: Demand incurred in current PP	$DICPP(t_0) = 0$
$DO(t)$: Demand overflow	$DO(t_0) = 0$
$DBWO(t)$: Demand being worked on	$DBWO(t_0) = 0$
$IUDP(t)$: Items in use at downstream process	$IUDP(t_0) = 0$
$IUUP(t)$: Items in use at upstream process	$IUUP(t_0) = 0$
$HB(t)$: Heijunka box	$HB(t_0) = 0$
$APDIK(t)$: Assigned PDI kanbans	$APDIK(t_0) = 0$
$UCUP(t)$: Units completed by upstream process	$UCUP(t_0) = pitch$
$QOC(t)$: Quantity on order from external customer	$QOC(t_0) = 0$
$UCDP(t)$: Units completed by downstream process	$UCDP(t_0) = pitch$
$NKC(t)$: Total number of kanbans in circulation	$NKC(t_0) = INK$
$NKR(t)$: Number of kanbans to be removed	$NKR(t_0) = 0$

5.4 Model Testing



The historical timeframe selected for model testing is indicated in **Figure 5.5**. It covers the first problematic cycle of inventory growth and recovery identified, corresponding to approximately 60 days. It is appropriate as a reference because no significant supplier shortages occurred during that time. Additionally, the occurrence of a single demand spike facilitates the understanding of the system's dynamics.

Appendix E provides a discussion on tests performed on the real case model. **Figure 5.6** focuses on the results of the particularly illustrative behavior reproduction test. Note how the model not only reproduces quite well the shape of the backlog curve, but it also provides considerable quantitative precision.

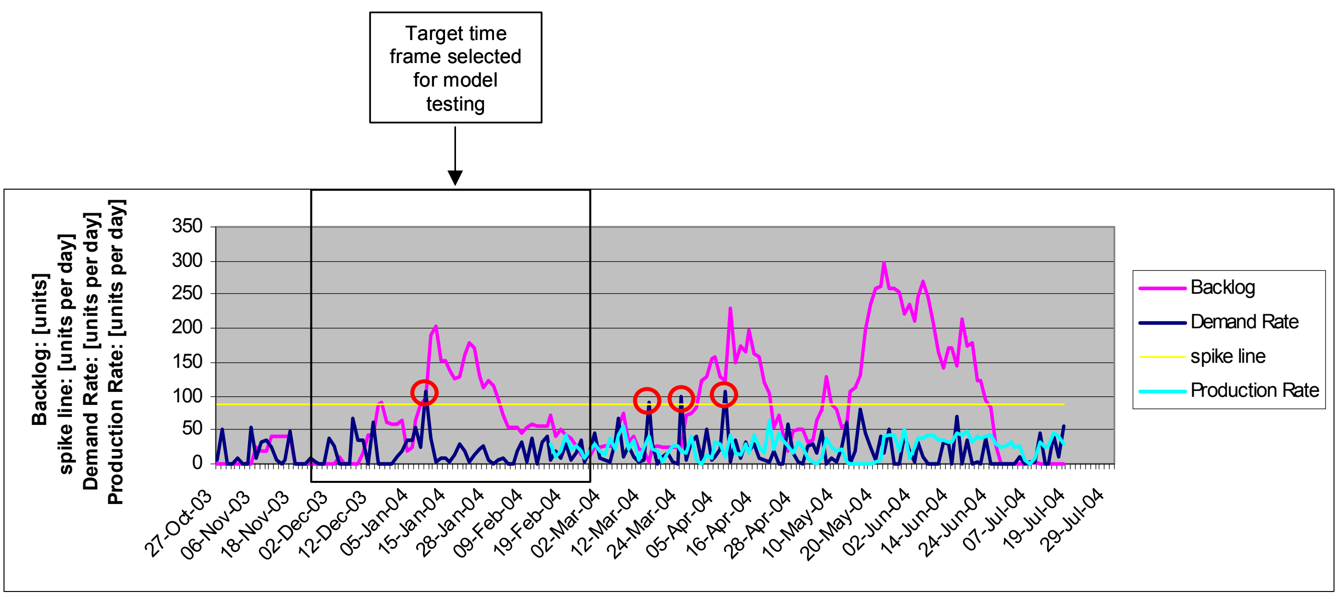


Figure 5.5 *Time Frame Selected for Model Testing.*

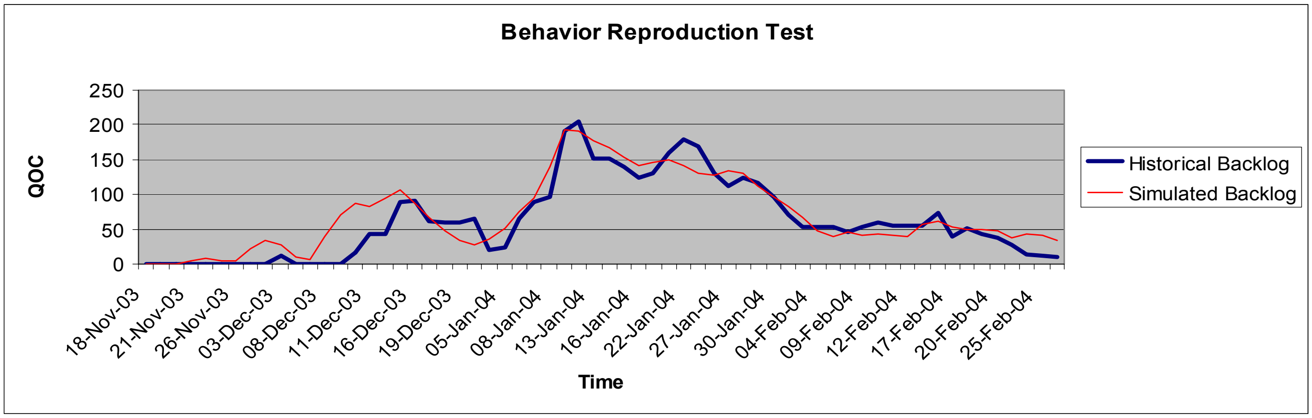
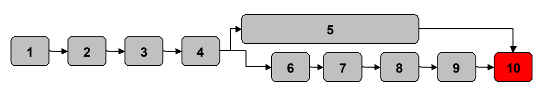


Figure 5.6 *Behavior Reproduction Test.*

As an observation, the model seems to slightly overestimate the amount of backlog in the days prior to Christmas and New Year celebrations. For the purpose of this study, this issue is not significant enough to the point of implying that assumption 9 (regarding no advance notice of demand) should be discarded. As a suggestion, future, more advanced, versions of this model could be used to test the effect of informal communication as well as schedule pressures on the production rate. Such procedure may possibly lead to better understanding of this phenomenon.

5.5 System Redesign and Evaluation



This section describes the evaluation of alternative TO-BE scenarios. These scenarios correspond to the adoption of the policies 1, 2, and 3 described in **Chapter 4**.

5.5.1 Policy Design

This real case study utilizes the same policy rationales developed for the hypothetical case. In fact, policies 1 and 2 are identical to the ones in the hypothetical case, including any embedded parameters. In the case of Policy 3, the specific setting of the protective production capacity parameters used in the analysis was defined according to the real system's actual conditions.

The *protective downstream production capacity factor (PDPCF)* adopted for Policy 3 corresponds to a 20% increase in the downstream production rate. This is the estimated impact of incorporating an additional operator at the final assembly process. **Table 5.7** summarizes Policy 3 characteristics, illustrating the impact on the *Flexibility Factor.*

Table 5.7 *Policy 3 Characteristics in the Real Case.*

Variable	AS-IS Estimate	TO-BE Policy 3 Estimate
Protective downstream production capacity factor (PDPCF)	0.21	0.43
Protective upstream protective capacity factor (PUPCF)	0.40	0.40
Flexibility Factor (F=PDPCF/PUPCF)	0.53 (<1: inactive)	1.08 (>1: active)

5.5.2 "What If…" Analysis

Evaluation of all scenario combinations was performed via simulation. The baseline and each one of the redesign alternatives were subjected to the same historical demand profile. The results are portrayed in **Figure 5.7**.

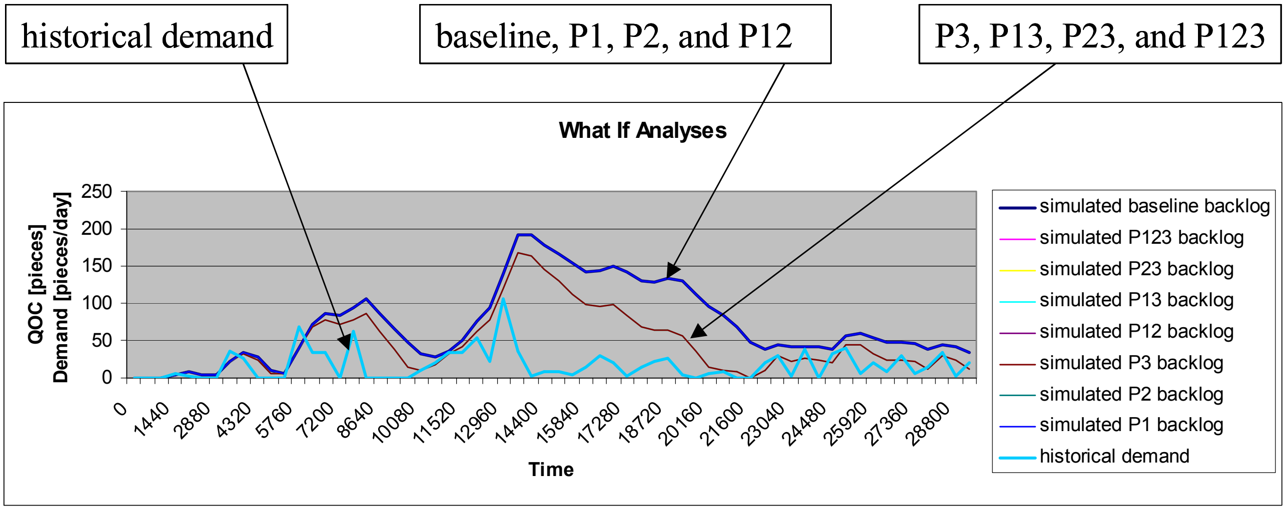


Figure 5.7 *Graphical Results of What If Analyses.*

Table 5.8 provides a ranking of "what if" scenarios according to *Total QOC*, i.e., the area under the *QOC* curve. These findings suggest a significant impact of Policy 3. Policies 1 and 2, in isolation or in combination, have no effect over this system. Scenarios P3, P13, P23, and P123 do present recovery to approximately zero *QOC* around time 21360 [min]; whereas the baseline, as well as scenarios P1, P2, and P12 do not completely recover – the minimum *QOC* level achieved by these scenarios is approximately 50 [pieces].

Table 5.8 *Ranking of Redesign Alternatives.*

Ranking	Combination	Total QOC (TQOC) (Difference relative to baseline)
1st	P3	-36.1%
1st	P13	-36.1%
1st	P23	-36.1%
1st	P123	-36.1%
2nd	P1	0%
2nd	P2	0%
2nd	P12	0%

5.5.3 Sensitivity Analysis

In accordance to these findings, additional analyses were conducted on Policy 3. **Table 5.9** illustrates the impact of Policy 3 on *Total QOC* at various levels of downstream capacity increase. The corresponding *protective downstream production capacity factor (PDPCF)* as well as the *Flexibility Factor (F)* are also presented. **Figure 5.8** illustrates these results graphically.

Table 5.9 *Policy 3 with Varying PDPCF Levels.*

Increase in downstream capacity	PDPCF	Flexibility Factor (F)	Total QOC (TQOC) (Difference relative to baseline)
10%	0.32	0.80	-23.4%
20%	0.43	1.08	-36.1%
30%	0.53	1.33	-45.5%
40%	0.64	1.60	-52.2%
50%	0.74	1.85	-55.1%
60%	0.85	2.13	-57.2%
70%	0.96	2.40	-58.7%
80%	1.06	2.65	-59.6%
90%	1.17	2.93	-60.2%
100%	1.27	3.18	-60.6%

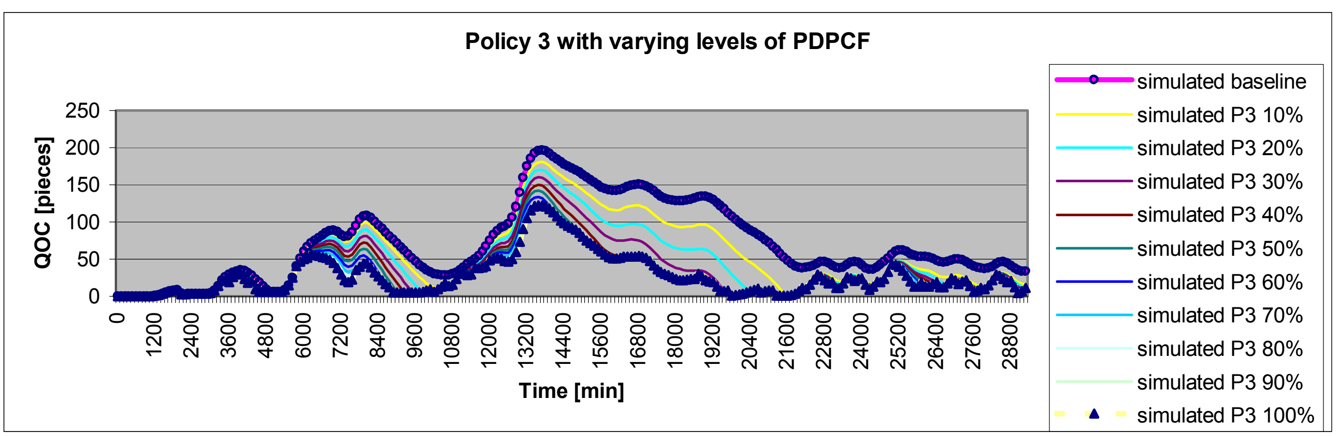


Figure 5.8 *Sensitivity of Policy 3 to the Choice of PDPCF.*

The impact of downstream capacity increase is not linear. For instance, the various *QOC* curves in **Figure 5.8** do not follow a parallel pattern. Note that equal production capacity upstream and downstream is already achieved with a 15% increase in downstream capacity (i.e., at this point the *Flexibility Factor* equals 1). Nevertheless, beneficial impact is observed as downstream capacity is increased beyond that point (i.e., *Flexibility Factor >1*). However, the magnitude of this beneficial flexibility tends to diminish with the increase of capacity (see **Figure 5.9**). The results for this particular case suggest that no significant additional impact can be obtained beyond 60% increase in downstream capacity.

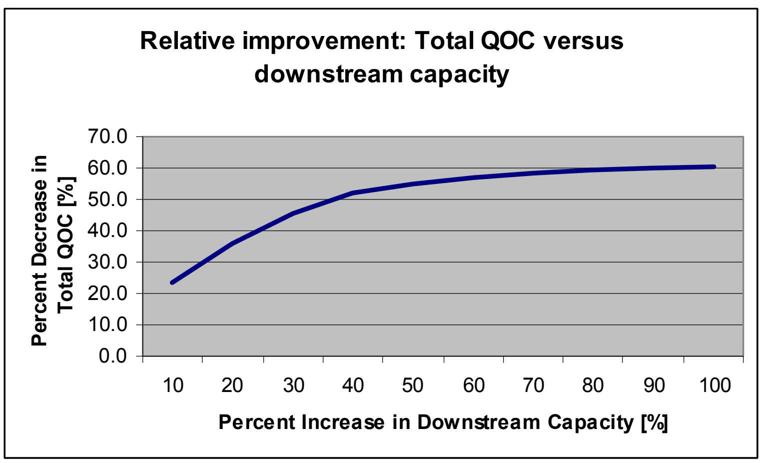


Figure 5.9 *Relation Between Downstream Capacity Increase and Total QOC Improvement.*

5.5.4 Sensitivity of Policy 3 to Noise in Production Rates

Considering a likely scenario of 20% increase in production capacity, **Figure 5.10** portrays the graphical results of the sensitivity of *QOC* to noise in the production rates. Data from 200 independent simulation replications was collected. The central tendency line indicates the average case, whereas the colored areas indicate various degrees of confidence in the results.

The graph suggests that it is possible that *QOC* will grow to alarming levels, should non desirable probabilistic conditions systematically accumulate; however this is very unlikely. On the contrary, there is a significant concentration of results towards the minimum limits. This supports the findings presenting to this point, confirming that actual improvement can be obtained with the adoption of Policy 3.

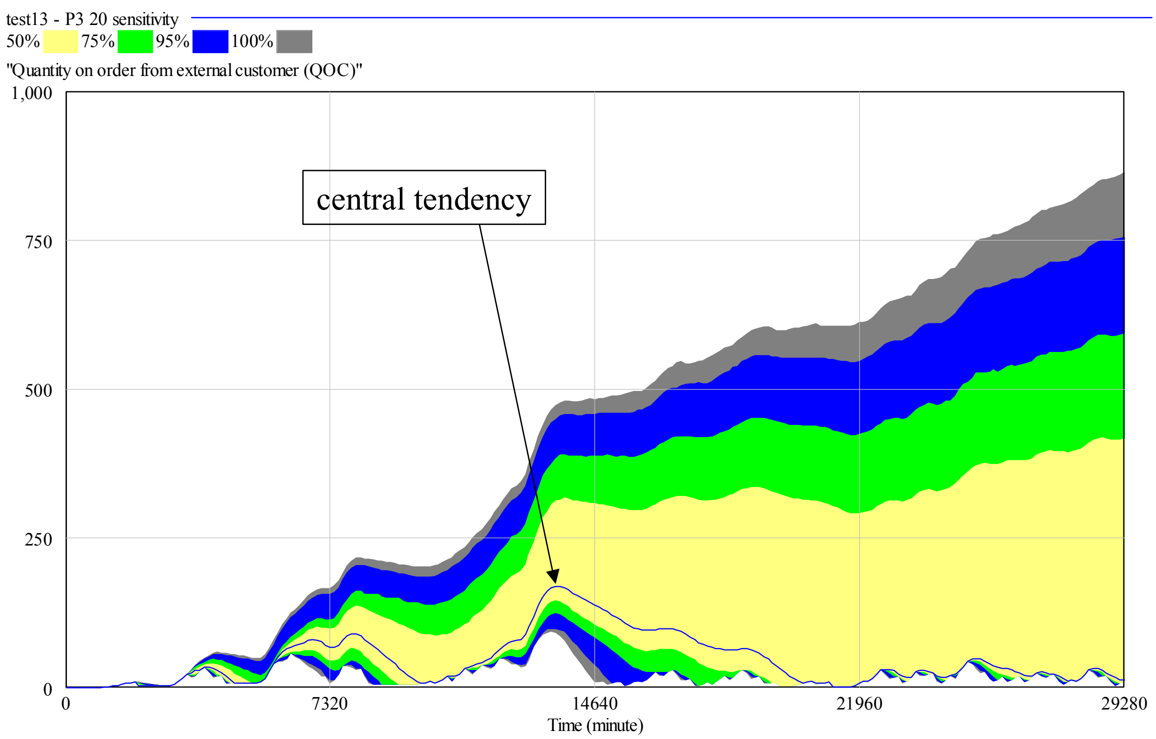


Figure 5.10 *Sensitivity of Policy 3 at 20% Downstream Capacity Increase to Noise in Production Rates.*

5.5.5 Discussion of Results

These results point to the conclusion that the incorporation of Policy 3 is a preferred course of action capable of significantly impacting the performance of the target value stream.

Policy 2 is neutralized by the system's structure, which is built according to the directive that the value stream can or should be able to absorb any amount of customer demand incurred, i.e., to strictly follow the voice of the customer at all times. Under this structure, the *average demand quantity for a planning period (ADQPP)* variable – which is used as a reference for Policy 2 decisions – always equals the entire amount of demand incurred in the previous planning period.

Consequently, the full implementation of Policy 2 as described in **Chapter 4**, is not possible without also incorporating some other structural changes and a fourth policy regarding *PDI* sizing and the identification of its components (cycle inventory, buffer inventory, and safety inventory).

Furthermore, the results also suggest that under the target value stream conditions, Policy 1 cannot add any delivery performance benefit to the other scenarios.

5.5.5.1 P1

This study suggests that manipulating the number of kanbans in circulation at the *PDI* according to Policy 1 rationale provides no benefit for the target value stream in question. In this particular case, given the problematic historic demand behavior, Policy 1 promotes duplication in the number of kanbans early in the simulation. The conditions are such that demand overflow is never reduced to zero, and therefore the number of kanbans does not return to the original value.

Figure 5.11 illustrates the increase in *PDI* size. However, considering that: (1) the original maximum *PDI* size is already large enough to ensure that no interruption in flow occurs, and (2) the downstream process is the one that constrains the flow in the value stream, then no benefit can be obtained from the additional inventory provided by the P1 scenario.

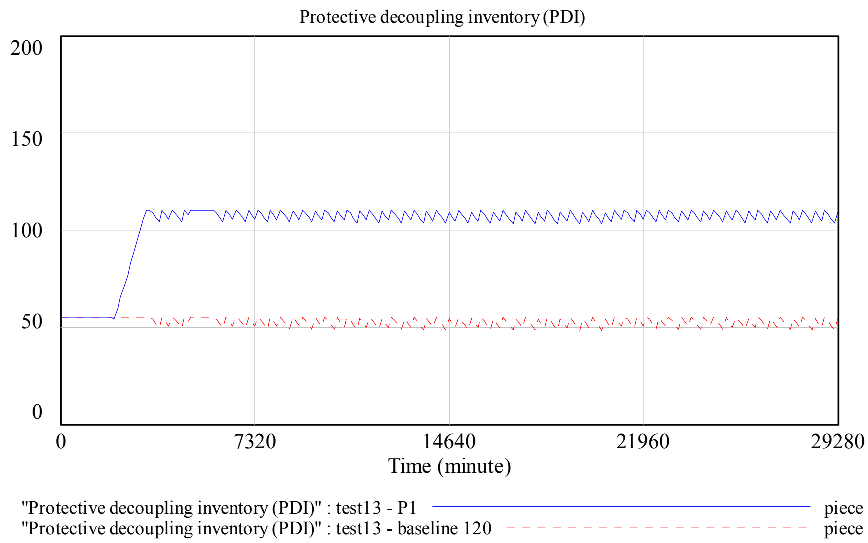


Figure 5.11 *PDI Level: Baseline vs. P1.*

5.5.5.2 P2

The purpose of Policy 2 is to provide support for the decision of what amount of demand to allocate to the shop floor according to an estimate of how much the system can actually absorb. It attempts to secure a certain amount of safety stock for use at an opportune time (such as when an upstream process breakdown occurs, or the event of a demand spike).

As such, under P2, sometimes the entire incurred demand would be allocated and sometimes just a fraction of it would be allocated. However, given that the system forces the perception that the demand quantity to be allocated to the floor equals the entire demand incurred, an infrastructure for the incorporation of P2 is not present. Under these conditions, P2 always provide the same decision, i.e., to allocate the entire demand incurred – therefore remaining neutral.

5.5.5.3 P12

P2 is neutral. This scenario is equivalent to the scenario P1.

5.5.5.4 P3

The adoption of Policy 3 implies an increment in downstream processing capacity. The P3 scenario tested considers a 20% increase. This increment in downstream capacity directly reflects in an increment in the value stream's capacity, impacting the ability of the value stream to increase its flow.

Additionally, this increase is also enough to surpass the upstream capacity and shift the bottleneck to the upstream process. Nevertheless, because the *Flexibility Factor* is now active (i.e., the value stream is able to sustain the maximum downstream production rate while the *PDI* lasts) additional benefits are also achieved, further contributing to diminish the backlog.

5.5.5.5 P13

Given the current *PDI* sizing policy in use at the target system, the adoption of P3 does not incur into complete use of the *PDI* at any time (i.e., the *PDI* does not go to zero at any point during the simulation – see **Figure 5.12**). Consequently, any additional inventory enabled by P1 does not contribute to promote uninterrupted flow through the value stream, therefore not affecting the delivery performance.

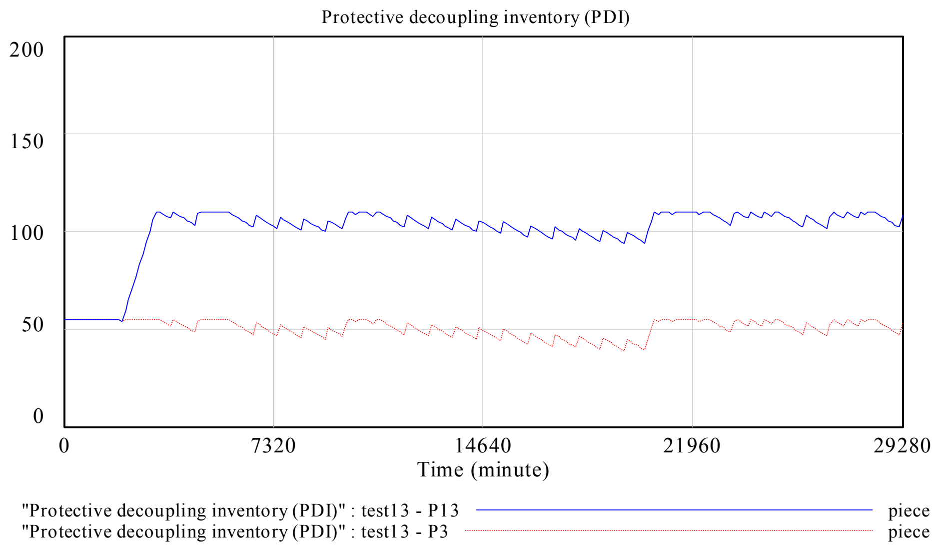


Figure 5.12 *PDI Level under P3 and P13.*

5.5.5.6 P23

P2 is neutral. This scenario is equivalent to the scenario P3.

5.5.5.7 P123

P2 is neutral. This scenario is equivalent to the scenario P13.

5.5.6 Comparison to Hypothetical Case

The results from both the hypothetical case and real case suggest that the adoption of Policy 3, by itself in the real case application and in combination with Policy 2 in the hypothetical case, is capable of promoting significant improvement. **Table 5.10** shows a summary of findings (Note: the percentage results from each case need to be interpreted in light of the fact that in the hypothetical case the simulation was initiated in balanced dynamic equilibrium so that the impact of the demand spike and adopted policies could be isolated. In the real case this was not so. The real case study strategy was to evaluate the impact of proposed polices over a specific historic period of this system.).

In the hypothetical case, the incorporation of Policy 1, either by itself or in combination with other policies, has shown indications of only minor impact. In the real case, given the PDI sizing conditions, the incorporation of this policy does not provide any benefit. In summary, these two sets of findings suggest that no significant benefit should be expected from the incorporation of Policy 1.

Table 5.10 *Comparing the Real and Hypothetical Cases.*

HYPOTHETICAL			**REAL**		
Ranking	**Combination**	**Total QOC (TQOC) Difference relative to baseline**	**Ranking**	**Combination**	**Total QOC (TQOC) Difference relative to baseline**
1st	P123	-6.3%	1st	P23=P3, P123=P13	-36.1%
2nd	P23	-6.0%	2nd	P12=P1, P2=baseline	0%
3rd	P12, P2	-4.6%			
4th	P13	-2.7%			
5th	P3	-1.5%			
6th	P1	0.0%			

With regard to Policy 2, the hypothetical study suggests there could be some benefit in utilizing this policy by itself, especially in situations involving large kanban sizes, given that it may enable the processing of an additional kanban within a planning period. Furthermore, the results also suggest significant benefits from associating Policy 2 with Policy 3; the first promotes the allocation of higher demand quantities to the floor promoting the benefits just described, whereas the second promotes even more improvement by enabling a larger portion the allocated demand to be processed.

In fact, in any value stream, there must be some policy regarding how and how much of the incurred demand is allocated to the shop floor. If upstream breakdowns are a concern, then a policy such as Policy 2 could be utilized. On the other hand, if the upstream process has high reliability then simply allocating the entire incurred demand (as in the real case baseline) is probably the best strategy. In the hypothetical case, the incorporation of Policy 2 implies the incorporation of a less conservative strategy, which promotes higher pay-offs in delivery performance; at the expense of a higher risk of interrupting flow across the value stream due to upstream breakdowns.

In the real case, the full impact of Policy 2 could not be tested. However, under the same logic just described, some conclusions might be drawn. The full incorporation of Policy 2 in the real case would actually imply the incorporation of a more conservative strategy; one with a lower risk of interrupting flow across the value stream but at the cost of less available inventory. Policy 2 would tend to utilize the "safety" portion of *PDI* only in the event of a demand spike. This would mean that generally less *PDI* would be available for the system's operation and possibly the downstream process would be forced to stop due to lack of inventory more often – probably leading to deterioration in delivery performance.

However, caution should be exercised in not generalizing these conclusions around Policy 2 in the real case for every situation. They would be valid if the total *PDI* size remains the same whether or not this policy is utilized. In this real case, however, the incorporation of Policy 2 would force a review of yet another policy (the *PDI* sizing policy) and the clear identification of the *PDI* components (cycle, buffer, and safety inventory). This procedure would require the definition of yet additional parameters. The final result may or may not provide the same total *PDI* size. In case it is reduced, the deterioration in performance could be even more evident. In case it increases, it could mean no deterioration at all.

Despite the fact that the proposed Policy 2 could not be tested in the real case, the results from the hypothetical case and the real case application are very similar. This is so if understood that in the real case there was always a policy similar to Policy 2 in operation. In other words, the real case condition of always sending to the floor the entire demand incurred in fact causes a similar effect of a non-conservative Policy 2. It is as if Policy 2 was set with such a high threshold that no demand variability would be filtered. However, similar results should not be expected in future studies that also incorporate supplier instability in addition to customer demand instability. This is the environment where the proposed Policy 2 is likely to provide its full benefit.

Chapter 6
Conclusions

The purpose of this research was to develop alternative flow control policies and evaluate their impact on value stream delivery robustness under external demand instability. The three research objectives listed in **Table 6.1** support and clarify this purpose.

Table 6.1 *Review of Research Objectives.*

Research Objective	Description
Objective 1	Create a comprehensive explanation of how the behavioral characteristics of traditional value streams emerge as a result of the system structure
Objective 2	Identify and describe modes of demand appropriate for pull-based system operation
Objective 3	Propose and test alternative flow control solutions

This chapter first describes the achievement of each one of these objectives. It then highlights the main research contributions as well as its limitations. Finally, this text provides suggestions for future related research themes.

6.1 Objective 1

The research strategy adopted was one based on system dynamics modeling theory. This approach has promoted insight into the relationship between structure and behavior of value streams that have a serial structure. A novel description of generic value streams was developed by means of causal loop diagrams with stocks and flows. This causality description created the basis of a dynamic hypothesis, which was then converted into a computer simulation model and tested. The results confirmed the ability of the model to explain the main mode of problematic behavior in question, and therefore the hypothesis was accepted.

This effort has served to identify some key value stream structural elements from a dynamic point of view and promoted insight into the role they play in generating the system's dynamics. In particular, the clear distinction between upstream and downstream processes as well as the protective decoupling inventory has proven relevant. Other key components of a value stream include: load-leveling box, additional kanban controlled supermarkets, as well as scheduling and execution agents.

6.2 Objective 2

The problematic impact of demand instability on value streams is the condition that motivated this research. A search for what modes of demand behavior are considered appropriate for high performing pull systems operation was conducted. The answers obtained were mostly qualitative in nature, generically suggesting that the demand should be "stable."

The reasons for this argumentation are basically three-fold: (1) these systems are usually designed in a deterministic manner that assumes average values for variables that in practice contain noise (as such the system tends to behave according to design when these variables operate at tight values close to average, and not so much when significant dispersion of values occur); (2) the typical balanced, integrated, and lean nature of pull supply chains requires change in production capacity at extensive parts of the system and not just at an isolated processes (therefore, there can be significant inertia associated to capacity levels), and (3) sudden changes in capacity are likely to involve different operator configurations and cause quality problems (therefore being avoided in the day-to-day operation, further contributing to the inertia of these systems).

A parallel line of investigation was also conducted with the intent of incorporating more precision into the understanding of these reasons, particularly the second one. It was based on the consideration that demand profile could, in principle, be in the form of any of the main modes of dynamic behavior (i.e., stasis, growth, decay, and other modes combining growth and decay). It also assumed that rates of flow (such as production rates) could be kept under strict control and therefore operate at levels significantly close to average. Under such conditions, and in alignment with additional assumptions adopted in the hypothetical case study, an analysis of impact of each of these modes of behavior was performed.

It became clear that whether or not a certain demand profile was appropriate for pull system's operation would naturally depend on the definition of performance used. In any case, some generic lessons could be obtained. For instance, the ability of a pull system to cope with strictly growing pattern would depend on the synchronization of the rate of demand growth and the rate of production capacity growth. This synchronization, on its turn, will depend on the frequency of capacity changes as well as the policies used to do so. On the other hand coping with strictly decaying patterns would not pose a problem from a customer delivery perspective, but it could be a problem from a resources utilization point of view.

Nevertheless, less intuitive were the impacts of modes of behavior based on stasis but with temporary peaks of growth and decay. In these cases, it was understood that disequilibrium dynamics are not obvious and require mathematical assessment. Being this condition well aligned with the problematic mode of behavior motivating the research, it was chosen as reference for the hypothetical case study.

6.3 Objective 3

Three alternative value stream flow control policies were considered as possible addendums to the state-of-the-art in value stream mapping theory. Two of them (Policy 1 and Policy 2) were treated endogenously in the model, whereas Policy 3 remained exogenous. Policy 3 remained exogenous because the simulation horizon is set to less then an estimation period; and the impact Policy 3, by definition, remains constant within estimation periods. These policies can be contextualized within generic value streams using the policy diagram in **Figure 6.1**.

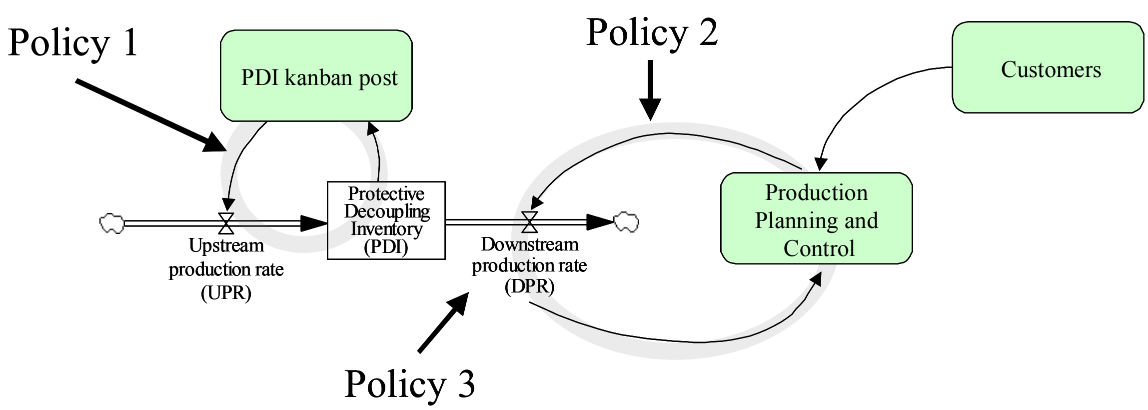


Figure 6.1 *Generic Value Stream Policy Diagram.*

6.3.1 Policy 1

Given the pull characteristic of value streams, information flows upstream in a cascading effect by means of kanbans loops. Policy 1 interferes in the left loop presented in **Figure 6.1** involving the PDI kanban post subsystem and the upstream process. In order to add or remove kanbans circulating in the *PDI* supermarket, Policy 1 requires the creation of a direct flow of information from the PPC subsystem to the PDI kanban post subsystem (not shown in the figure). This "flexible kanban" system inserts an additional flow of information that bypasses the otherwise synchronized cascading flow of information promoted by the kanbans attached to the material parts. As a consequence, Policy 1 ends up inserting noise into the flow of information.

According to the simulation results, both hypothetical and real, only a minor beneficial impact is expected from this practice when there is no advance notice of demand (a condition reflected in the assumption 9 presented in **Chapter 4**). Typically these minor benefits should be expected in situations when supermarkets are calculated with low safety levels. On the other hand, however, Policy 1 has the potential to cause significant destabilizing effects in the entire upstream supply chain, as distorted demand signals are generated. Policy 1 inserts risky instability and unnecessary complexity in value streams, and therefore its use is not recommended.

Policy 1 has been proposed with the intent of providing a very simple and intuitive rule for altering the number of kanbans, one that not necessarily uses an estimate of demand or a precise compilation of demand incurred. Despite the fact that the use of Policy 1 is not recommended, other flexible kanban policies proposed in the literature should not be ignored. However, great care should be exercised in understanding what the "flexible kanban" label actually means as well as which supermarket(s) in the system is (are) target(s) for intervention.

For instance, the line of works developed by Dr. Gupta and his group deserves comment. One line of FKS strategies serves to alter the number of kanbans as a means to provide leveled production schedule to the shop floor when there is advance notice of demand (Gupta & Al-

Turki, 1997; Gupta et al., 1999). In value stream context, it could serve a similar structural purpose as the heijunka box in order to provide a leveled daily demand schedule to the pacemaker. Nevertheless, in alignment with the conclusions here presented, this line of work emphasizes that the timing for addition and removal of kanbans is still very critical in order not to create undesirable instability in the system.

Another line of FKS policies (Gupta & Al-Turki, 1998b) manipulates the number of kanbans in selected supermarkets as a means to break the pull control linkage between succeeding stations. Such breakage allows upstream processes to keep producing in the event of a downstream breakdown. However, this is a practice that does not necessarily promote the total quality orientation of lean production philosophies.

6.3.2 Policy 2

Policy 2 interferes in the loop at the right in **Figure 6.1**, involving the PPC subsystem and the downstream process. This loop is a necessary component of value streams and requires some policy to support the decision of how much demand to allocate to a schedule. Policy 2 in fact constitutes a solution to a problem identified by Smalley (2004), who emphasizes the need for the formalization of such a policy capable of accounting for unstable demand and supply situations in value streams effectively.

The results of the simulation study suggest that under unstable demand (and stable supply) Policy 2 can provide benefit depending on the kanban sizes utilized; as extra quantities provided to the floor may trigger the production of an additional kanban within the same planning period. However, the smaller the kanban sizes, the more neutral Policy 2 will be in this regard.

The hypothetical case showed that Policy 2 promotes better performance under unstable demand conditions when compared to a policy that always allocates the estimated average system production capacity, the basic policy advocated by Rother and Shook (1999). Furthermore, the major benefit of adopting Policy 2, as opposed to a policy that simply always allocates the entire demand incurred (as in the real case described in **Chapter 5**), is likely to be felt in unstable supply conditions (with stable or unstable demand). The reason is that Policy 2 serves the purpose of attempting to keep the safety portion of PDI untouched, so that it can be used in the event of an upstream breakdown. Although a Policy 2 like policy is a need in value stream production planning and control, further study involving upstream supply chain instability is required in order to increase understanding of the effectiveness of the proposed Policy 2.

6.3.3 Policy 3

The incorporation of a management loop based on Policy 3 is not a necessary structural component of value streams. However, simulation results in both the hypothetical and real cases suggest that the use of Policy 3, as opposed to a policy that seeks to maintaining the entire value stream capacity balanced and lean, may provide interesting strategic benefits.

As part of the Policy 3 rationale, a *Flexibility Factor (F)* was defined corresponding to the ratio between the protective downstream capacity and the protective upstream capacity. When F>1, this was referred to as “active” flexibility effect, meaning that given the availability of protective decoupling inventory, the value stream can have its downstream rate accelerated in an attempt to cope with a demand peak.

This unbalanced condition enables the system to absorb certain demand peaks without disturbing the upstream process. The size of demand peaks the system can absorb as well as the timing to do so depend on the available *PDI* at the time of peak occurrence as well as the amount of idle capacity downstream. Ideally the upstream and downstream capacity, as well as PDI size, would be set in such a way that the system can absorb expected peaks in a timely manner and the *PDI* will be refilled between peaks.

The added flexibility characteristic comes with the cost of carrying extra capacity downstream. Various experts agree that this is a necessary cost for the insertion of flexibility to demand quantity into a system (Naylor et al., 1999; Hoek, 2000; Mason-Jones et al., 2000). The results of this research support this argumentation. However, in value streams where the decoupling point is situated close to the external customer the cost benefit tradeoff of the practice promoted by Policy 3 could prove competitive. In such cases, only a small portion of the value stream (i.e., the downstream process) would keep extra capacity; whereas the entire upstream system can follow strict lean guidelines and keep capacity to a minimum.

6.4 Contributions

This research provides some important additions to the theory and practice of value stream design and management. Some key contributions are:

- The constructed dynamic hypothesis regarding how the structures of value streams determine their problematic behavior makes available an initial causality theory. The state-of-the-art in value stream design guidelines has been formally documented within a system dynamics model. This model provides the literature an additional perspective of value stream structures, which improves the conditions for more precise insight into value stream dynamics; including the future development of broader, more complex models.

- The research has also proposed a policy to support the decision of how much demand to allocate in a value stream schedule in a planning period. This proposition serves as an initial solution and design guideline, fulfilling a need identified by Smalley (2004).
- The studies regarding unbalanced value stream capacity have provided results of immediate practical application. In the course of creation of Policy 3, the concept of *Flexibility Factor* was created. It serves as a generic value stream design guideline for the incorporation of flexibility into the system structure.
- The combination of Policies 2 and 3 creates a reference model for the design of key aspects of value stream flow control systems. This solution can be of particular utility for lean firms forced to operate at the decoupling point of their supply chains (please refer to **Figure 6.2**). In these cases, the proposed solution will create better conditions for absorbing eventual demand instabilities generated downstream, while promoting a stabilizing effect upstream. Both characteristics are of interest for such firms, since, on the one hand being better able to absorb demand instability may improve on time delivery. On the other hand, more stable conditions upstream facilitate the development of suppliers.

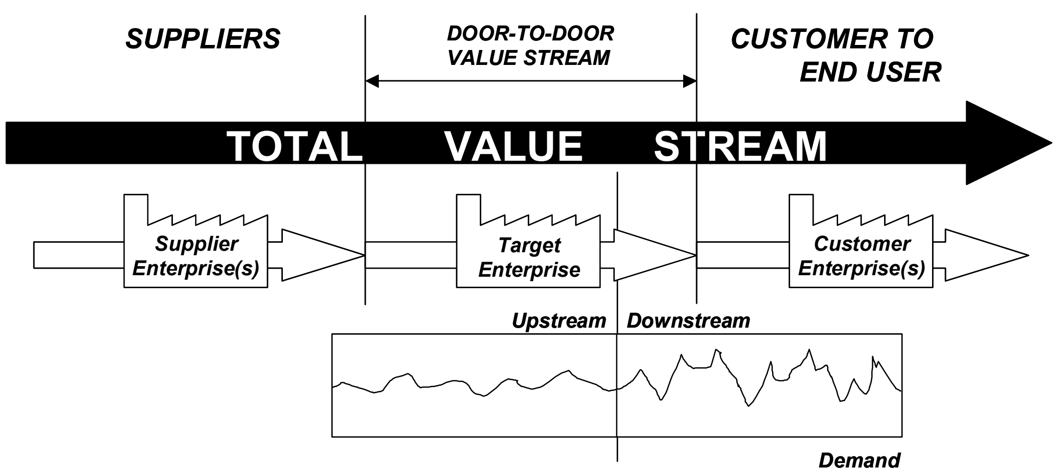


Figure 6.2 *Supply Chain Decoupling Point.*

6.5 Limitations

Given the research strategy based on modeling, this investigation is limited in various ways. By definition, no model can be completely validated in the sense of establishing truth about a phenomenon (Sterman, 2000 p.846). Furthermore, as certain simplifying assumptions are relaxed, the boundaries of the present model are expanded and a more comprehensive network of causality relationships is considered, chances are that the model will become more realistic. Some aspects of this type of limitation will be highlighted. Furthermore, given the current assumption and boundary conditions, limitations that are mostly technical in nature will also be pointed out. This discussion is divided into two parts, one pertaining to the generic hypothetical case, and another one specific to the real case application.

6.5.1 Simplifying Assumptions Related to Hypothetical Case

Limitations related to the adopted assumptions:

- This research has treated demand as an exogenous variable. It assumed that the performance of the system does not feed back to significantly alter the demand behavior. Future work might benefit from endogenous considerations of demand. Please refer to Gonçalves (2003) for discussion on this theme.
- Likewise, the model also does not consider the utilization of demand forecasts or the consideration of demand already known but not due immediately. The incorporation of these features will increase the applicability of the model in real world scenarios.

- The baseline considered in this study does not involve mixed pull situations. In fact, it best represents a replenishment system with no significant differences among product family variants. Expansion of the model in this regard might be necessary.

Limitations regarding technical modeling choices:

- The use of TIME STEP as a way to represent discrete system characteristics limits the choices of numerical integration methods to the Euler method.
- The current model does not allow the incorporation of significant time delays in the transfer of kanbans. It currently assumes that the dynamics of kanban transfers are insignificant in relation to the dynamics of the problem in question.

6.5.2 Simplifying Assumptions Related to Real Case Application

Limitations regarding the adopted assumptions:

- The model does not capture subtle changes in production capacity caused by alterations in schedule pressure, or the eventual use of a few additional extra hours of work. These minor changes can be partially captured by the introduction of noise. However, it does assume that capacity remains constant over time and therefore will be unable to consider situations involving significant capacity change within planning periods.
- The model does not distinguish the few product family variants and would be unable to consider situations involving significant set up times.

Limitations regarding technical modeling choices are the same as in the generic case.

6.6 Areas for Future Research

Generally, it has been observed in this investigation that the closer a value stream structure gets to the ideal value stream structure advocated in the literature, the simpler the system structure gets from a dynamic point of view. Nevertheless this simplicity is relative. Even the best well crafted lean discrete parts manufacturing systems can be tremendously complex. Additionally, no matter how simple they are, there will always be a need to account for certain critical structural elements. Furthermore, in the path towards lean structures, there is likely to be a multitude of potential scenarios requiring direct attention of system designers.

In alignment with these considerations and maintaining the focus on the problematic behavior that motivated this present study, a few areas of related future research have been identified. They are meant to increase the realism and trust in the model under a broader range of situations, as well as improve the ability of lean enterprise engineers to implement solutions derived from the model.

Some relevant complementary areas of future research focused on dynamic complexity issues could include:

- The hypothetical model could be used to study the **impact of other problematic demand profiles**. In some real world scenarios there is particular need for better understanding the transient effects of oscillatory or seasonal conditions.
- The effects of **supplier instability** on value stream performance can be significant. Many potential model users could find great utility in a future version of the model that incorporates these considerations.
- Further benefit might also be obtained from a deeper **analytical understanding of the synergy between Policies 2 and 3**; eventually Policy 3 could be set in a manner that

attempts to optimize the *Flexibility Effect* in face of the expected demand profile. This study could also include a more enhanced consideration of the **performance characteristics of information flows** and how they affect the delivery performance of the value stream.

- The simulation model could be expanded to include **endogenous considerations of Policy 3**. This would also involve incorporating endogenous considerations regarding production capacity allocation. This would open a new dimension of analysis in this research front regarding the **flow of production resources**. The model would then become appropriate for the study of alternative capacity allocation strategies and the assessment of related dynamics. Potential **quality issues involved in transitioning among capacity levels** could also be a focus of investigation.
- From a total value stream point of view, the model could serve as "building block" for the **study of higher order problematic scenarios**; such as in studies involving more complex product structures, various critical suppliers, as well as larger supply chain domains extending beyond the limits of single firms. For instance, a study of the ability of lean supply chains to adapt capacity levels in a cascading effect as a response to changes in demand could be of interest to many industries – as well as to related labor organizations.

The exploration of additional issues involving a higher degree of combinatorial complexity might also provide interesting practical results. In these cases, however, the aggregate nature of system dynamics models might not provide an appropriate representation of physical flows. In such situations, the discrete-event modeling paradigm can be more favorable to the consideration of characteristics of individual entities flowing across the system.

Consequently, the isolated or combined use of discrete-event simulation models could be necessary. Some of these areas of investigation could include:

- Situations where delivery performance needs to be measured against the fulfillment of particular orders instead of overall quantity, the **identification of individual customer orders with varying sizes** will provide more precise estimates.
- The **distinction between product family variants** is important in situations where there is significant difference in processing and set ups. Models incorporating this feature make it possible to evaluate alternative scheduling and sequencing issues, and related queuing dynamics.
- Last, given the ability to track individual orders and distinguish among product family variants, studies involving the consideration of **mixed pull** situations could be considered.

Appendix A – Classification Of Literature Findings

Table A.1 presents a classification of literature findings according to the three main areas of literature review: enterprise engineering, production planning and control, and system dynamics.

Table A.1 *Literature Classification.*

Source	Enterprise Engineering	Production Planning and Control	System Dynamics	Other
(ARC, 2003)				♦
(Bernus & Nemes, 1996)	♦			
(Biemans et al., 2001)	♦			
(Bititci & Muir, 1997)	♦			
(Blackstone & Cox, 2002)		♦		
(Blanchard, 1991)	♦			
(Blanchard & Fabrycky, 1998)	♦			
(Bond, 1999)	♦			
(Bruno & Agarwal, 1997)	♦			
(Carley, 1999)	♦			♦
(Carlock & Fenton, 2001)	♦			
(Chang & Yih, 1994)		♦		
(Chapman, Rozenblit & Bahill, 2001)	♦			
(Cooney, 2002)		♦		
(Corrêa & Gianesi, 1996)		♦		
(Disney et al., 1997)		♦	♦	
(Dong & Chen, 2001a)	♦	♦		
(Dong & Chen, 2001b)	♦	♦		
(Fine, 1998)				♦
(Forrester, 1961)			♦	
(Forrester, 1975)	♦		♦	
(Forrester, 1996)			♦	

Source	Enterprise Engineering	Production Planning and Control	System Dynamics	Other
(Forrester, 1998)	♦		♦	
(Fowler, 1998)	♦		♦	
(Fowler, 1999)		♦	♦	
(Fung, 1999)		♦	♦	
(Gianesi, 1998)		♦		
(GlobalSpec, 2004)				♦
(Gonçalves, 2003)		♦	♦	
(Goranson, 1997)	♦			
(Goranson, 1999)	♦			
(Groesbeck, 2001)				♦
(Grosfeld-Nir et al., 2000)		♦		
(Gupta & Al-Turki, 1997)		♦		
(Gupta & Al-Turki, 1998a)		♦		
(Gupta & Al-Turki, 1998b)		♦		
(Gupta et al., 1999)		♦		
(Gupta & Gupta, 1989)		♦		
(Hoek, 2000)		♦		
(Hornby, 1995)		♦		♦
(Huq & Pinney, 1996)				
(INCOSE, 2002)	♦			
(Kamath, Dalal, Kolarik, Chaugule, Sivaraman & Lau, 2001)	♦			
(Keating et al., 1999)			♦	
(Keough & Doman, 1992)			♦	
(Kosanke et al., 1999)	♦			
(Kotter, 1996)	♦			
(Kumar & Vrat, 1989)			♦	
(Kurstedt, 2000)	♦	♦		
(Law & Kelton, 2000)				♦
(Leedy & Ormrod, 2001)				♦
(LEI, 2003)		♦		
(Liles et al., 2002)	♦			
(Liles & Presley, 1996)	♦			
(Little et al., 2001)		♦		
(Mason-Jones et al., 2000)		♦	♦	
(Mason-Jones & Towill, 1998)		♦	♦	
(Matson & McFarlane, 1999)		♦		
(Moeeni et al., 1997)		♦		
(Monden, 1981)		♦		

Source	Enterprise Engineering	Production Planning and Control	System Dynamics	Other
(Moore & Gupta, 1999)		♦		
(Naylor et al., 1999)		♦		
(Nightingale, 2002)		♦		
(O'Callaghan, 1986)		♦	♦	
(Oliva, 2001)			♦	
(Petrie Jr., 1992)	♦			
(Powell et al., 2001)		♦	♦	
(Presley, 1997)	♦			
(Pressman, 1995)				♦
(Razmi et al., 1998)		♦		
(Rees et al., 1987)		♦		
(Rentes, 2000)	♦			
(Rentes, 2002)		♦		
(Rentes et al., 1999)	♦			
(Repenning & Sterman, 2001)	♦		♦	
(Repenning, Gonçalves & Black, 2001)	♦		♦	
(Resende & Sacomano, 1997)		♦		
(Ritchie-Dunham & Anderson, 2000)	♦		♦	
(Ritchie-Dunham & Rabbino, 2001)	♦		♦	
(Rolstadas, 1998)	♦			
(Rother & Shook, 1999)	♦	♦		
(Scheer, 1991)	♦			
(Scheer, 1992)	♦			
(Scheer, 1993)	♦	♦		
(Scheer, 1994)	♦	♦		
(Scheer, 1999a)	♦			
(Scheer, 1999b)	♦			
(Senge et al., 1994)			♦	
(Shewchuk, 2003)		♦		
(Sipper & Bulfin Jr., 1997)		♦		
(Smalley, 2004)				
(Sousa, 1999)	♦			
(Sousa et al., 2001)	♦			
(Sousa, Carpinetti, Groesbeck & VanAken, 2003)	♦		♦	
(Spearman et al., 1990)		♦		
(Stephens, 1999)	♦	♦		
(Sterman, 1991)			♦	

Source	Enterprise Engineering	Production Planning and Control	System Dynamics	Other
(Sterman, 2000)			♦	
(Sterman, 2001)			♦	
(Stratton & Warburton, 2003)		♦		
(Takahashi & Nakamura, 1999)		♦		
(Takahashi & Nakamura, 2000a)		♦		
(Takahashi & Nakamura, 2000b)		♦		
(Takahashi & Nakamura, 2002a)		♦		
(Takahashi & Nakamura, 2002b)		♦		
(Tardif & Maaseidvaag, 2001)		♦		
(Taylor, 1999)		♦	♦	
(Towill, 1996a)	♦	♦	♦	
(Towill, 1996b)	♦	♦	♦	
(Vensim, 2002)			♦	
(Ventana, 2003)			♦	
(Vernadat, 1996)	♦			
(Vollmann et al., 1997)		♦		
(Walizer & Wienir, 1978)				♦
(Wang & Xu, 1997)		♦		
(White Jr., 1998)	♦			
(White Jr., 1998)	♦			
(Wiendahl & Breithaupt, 1999)	♦	♦		
(Wilding, 1998)		♦	♦	
(Womak & Jones, 1996)	♦	♦		
(Wortmann et al., 1997)	♦	♦		
(Yavuz & Satir, 1995)		♦		
(Yu et al., 2000)	♦			

Appendix B – Background And Extended Literature Review

This appendix is divided into two parts comprising a detailed background and an extended literature review: (1) Enterprise Engineering, and (2) System Dynamics Modeling and Simulation.

B.1 Enterprise Engineering

Enterprise Engineering emerged as a discipline out of the need to understand and design complex production systems operating under new technological paradigms, especially the ones created by the digital computer revolution (Kosanke et al., 1999). Its roots can be strongly traced to the Industrial Engineering discipline, especially the more recent efforts in Computer Integrated Manufacturing as well as in Systems Engineering.

As a young field of study, the literature in the area is not as plentiful and mature as in other more traditional engineering fields. Initial work has extensively utilized the label *Enterprise Modeling and Integration*. Recently, the expressions *Enterprise Engineering and Integration* and *Enterprise Engineering* have also been used.

B.1.1 Background

By the end of the 1950s and beginning of the 1960s, it became clear that one should be able to engineer a shop floor (the part of the manufacturing enterprise that actually converts raw material into products) the same way any other complex product is engineered. Although product design already had various specialized engineering disciplines (e.g. mechanical engineering, civil engineering, electrical engineering, etc.) that allowed relevant factors to be identified, analyzed, and manipulated, this was not true at the time for factory or shop floor design (Goranson, 1999).

The Air Force Manufacturing Practices group in Dayton, Ohio (USA) took the initiative to create a discipline focused on engineering the factories that produced their products. The central component was the representation or the modeling language that allowed managers to get a clear picture of what was going on in an unambiguous way and then make reasoned decisions. This research was originally called Intelligent Computer Aided Manufacturing (ICAM) program. Its results had a great impact on the discipline of Industrial Engineering and consisted basically of a set of modeling methods and the metrics that resulted from them (Goranson, 1999).

Furthermore, Goranson (1999b) also points out that the results obtained from the ICAM program were a major component of the U.S. military industrial strategy in the early days of Cold War. Such effort made some critical assumptions, all to be proven wrong later on:

- The whole product, in this case the weapons systems, was to be designed and manufactured completely within a single firm or enterprise;
- One could model the requirements for the product, while the world would sit still, and neither the requirements nor the implementing technology would change significantly;
- The hard part of everything was on the engineering side, not the business process side.

However, the following years showed a reality where weapon systems have been assembled by a systems integrator of components and have been designed and manufactured by thousands of suppliers. It may take 15 years or more to design and test a complex weapon, and in that timeframe the world changes considerably. The biggest problems have not been on the engineering side, but on the business side as large firms insist on managing complexity with a traditional top-down management style, even though the problems, systems, and nature of change have become more and more complex. As a result of this poor infrastructure, it is argued that these military manufacturing systems cost more then what they should, and until this day there are several advanced weapons that cannot be built at all, at any cost, until the production infrastructure is fixed (Goranson, 1999).

In recognition of these problems, the U.S. Department of Defense (DoD) responded with several research programs, some of them focused on how to manage dynamism and complexity. At this point, it was understood that not only the products and the facilities that produce them needed to be formally designed in an integrated manner. Evidences suggested that there was a real need for solid engineering approaches capable of putting together a system with even a larger scope, i.e., the total enterprise.

As a result, these DoD programs evolved first from a focus on C*oncurrent Engineering*, which investigated many distributed simultaneous designers. The next step was to coordinate the effort under the rubric of *Enterprise Engineering* and tackle agile manufacturing and integration issues (Goranson, 1999).

A closer consideration of these early Enterprise Engineering efforts reveals that integration was at that time treated as a concept very closely related to the notion of how to obtain high performance. Interestingly, the results obtained have impacted not only military operations but also many other industries across the globe. Consequently, the concept of integration has, indeed, a special significance in this context. Better understanding its various facets provides a valuable complementary perspective into the origin of this discipline.

B.1.2 Integration

As the historical facts just described were taking place, a parallel technological phenomenon was intensively evolving. Radically new technologies were appearing. Among them were the digital computers and fast communication technologies, which created a revolution in the way systems could be engineered as well as how integration itself was understood. *Computer Aided Design* and *Computer Aided Manufacturing* turned into a normal jargon in the Industrial Engineer vocabulary. The end of the 1980's marks the incipient stages of an even more advanced concept: *Computer Aided Enterprise Engineering* (Vernadat, 1996).

From a point of view of historical evolution of significant tools and technology, Kosanke et al. (1999) present a classification in terms of the volume and emphasis in effort dedicated to various engineering fronts, from the computer hardware integration in the 1960s, through software integration in the 1970s, business process integration in the 1980s and 90s, to the integration of enterprise in the 2000s (see **Figure B.1**). It is worth noting that, according to this classification, integration has always been, one way or another, a question of coordinating networks of some sort.

Physical systems integration has been a strong focus of attention since the 1960s. Topics of interest here are: (1) physical systems interconnection, (2) data exchange rules and (3) conventions, and inter system communication, network configuration and management.

Application integration focus on: (1) common (shared) data resources, (2) common services and execution environments, (3) portable applications distributed processing (Vernadat, 1996).

Business integration relies strongly upon the assumption that enterprises are a network of business processes. Main topics of interest are: (1) production and process simulation, (2) automated business process monitoring, and (3) knowledge-based decision support business control. In many cases, attention has been focused on intra enterprise business processes. One of the main outputs of these efforts has been the creation of enterprise-wide information systems, the so-called *Enterprise Resource Planning* systems or ERP's (Vernadat, 1996; Scheer, 1999a).

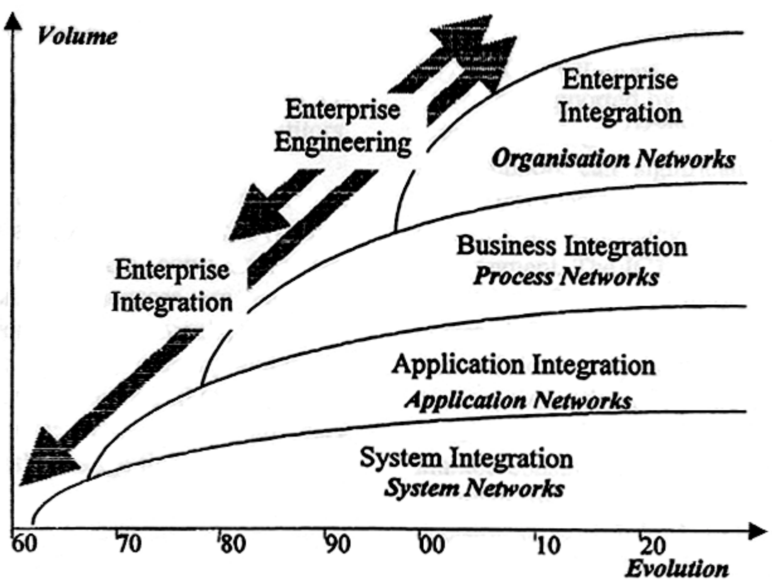


Figure B.1 *A Technological Perspective in the Origin of the Enterprise Engineering Discipline. Source: taken from Kosanke et al.* (1999)

Enterprise integration as defined here is a more recent concern. Some topics of interest in this area are: (1) change (i.e. configuration) management at the enterprise level, (2) flexibility and agility in enterprise operations, and (3) inter-enterprise coordination in the form of supply chains or virtual enterprises. A novel difference from previous efforts seems to be the concern with organizational elements in a holistic and integrated manner, i.e. the systemic considerations about the characteristics, role, place and involvement of people as intelligent agents acting in the system (Liles et al., 2002).

The key aspects promoted by *Enterprise Integration* efforts emphasize needs for changes in paradigms and promotes discussions towards the move from traditionally deterministic and monolithic considerations to highly non-deterministic and heterogeneous concerns (Goranson, 1999; Kosanke et al., 1999; Sterman, 2001). From a modeling point of view, Vernadat (1996) argues that Enterprise Engineering falls under the broad scope of Industrial Engineering, filling a gap between Operations Research and Management Science issues; and built upon several disciplines in science, engineering, and humanities. Liles et al. (2002) identify the following complementary supporting disciplines: (1) Systems Engineering and Systems Theory, (2) Information Systems, (3) Information Technology, (4) Business Process Reengineering, and (5) Organizational Design and Human Systems.

Many references indicate that Systems Engineering has perhaps been the most significant basis for the work developed under the label *Enterprise Engineering* (Liles & Presley, 1996; Bruno & Agarwal, 1997; Goranson, 1999; Biemans et al., 2001; Carlock & Fenton, 2001). INCOSE, the International Council on Systems Engineering, currently has an active working group named IEWG (Intelligent Enterprises Working Group) exploring applications of the systems engineering process in the case when the system in question is the enterprise (INCOSE, 2002).

In the U.S., at Virginia Polytechnic Institute & State University (VPI&SU), the discipline of Management Systems Engineering (MSE) has been under development for almost two decades. In this case the term *management system* has been used as a synonym to *enterprise*. The focus of MSE has been on investigating how the behavior of managers (i.e. decision-makers) affect the behavior of the enterprise, and on engineering the management processes used by them (i.e. the regulatory mechanisms that promote alignment, direction, and control to the total enterprise operations) (Kurstedt, 2000).

In Germany, at a few organizations such as the *Institut für Wirtschaftinformatik* (IWi) in the University of Saarlandes, the discipline of Business Process Engineering has also been under development for approximately two decades. In this case the systems engineering roots are also evident and the concern is also with the total enterprise system. However the emphasis is on specifying business process structures for the purpose of building integrated information systems for computer integrated manufacturing applications. Impressive products have appeared as a result of these efforts, chief among those are the computer aided enterprise engineering tool known as ARIS Toolset, and the ERP system known as SAP R/3 (Scheer, 1992; 1993; 1994; 1999b; 1999a).

The role of modeling and design has been a central focus of Enterprise Engineering efforts and a few other centers have developed very significant work specifically devoted to this purpose, including: Purdue University (USA), University of Toronto (Canada), and University of São Paulo (Brazil).

A somewhat similar purpose has been sought by the scientific discipline of System Dynamics. This discipline originated out of the pioneering work of Jay Forrester starting in the late 1950s at the Massachusetts Institute of Technology, Sloan School of Management (USA), in an attempt to apply engineering approaches to systems containing social components; in fact, the initial efforts started with a very specific and explicit focus on enterprises (Forrester, 1961; 1975; 1998; Sterman, 2000).

Therefore, research and practice under the label *enterprise design* has been part of the System Dynamics research agenda since the beginning of the field (Forrester, 1961). Efforts in this area, however, seem to have been primarily focused on the high level (and usually conceptual) assessment of dynamic behavior derived from enterprise structures. Considerable attention has been given to the dynamics of systems with large scope such as supply chains. The concept of feedback control is central, and mature simulation modeling techniques have been developed (Sterman, 1991; Fowler, 1999; Sterman, 2000; Repenning et al., 2001; Repenning & Sterman, 2001; Ritchie-Dunham & Rabbino, 2001).

B.1.3 The Enterprise as a System

All Enterprise Engineering related approaches mentioned so far strongly adopt one common assumption: an enterprise is a system (i.e. a collection of processes) that, despite its enormous complexity, can be engineered to achieve specific organizational objectives.

The term *enterprise* is used to designate the target system scope under consideration, i.e. the boundaries describing what part(s) of a corporation(s), firm(s), or production system(s) are targeted for engineering or analysis. Consequently, the term enterprise may possibly refer to the scope that is defined legally by the concept of firm, company; corporation, etc. but this isn't necessarily so. As utilized under the engineering approach, the term enterprise denotes a certain organizational domain of interest, which may actually refer to parts or wholes of several firms as in the case of the extended enterprise (Vernadat, 1996).

Treating an enterprise as a system implies that it consists of a set of interrelated components working together towards some common objective or purpose (Blanchard & Fabrycky, 1998). Traditionally, the enterprise has been viewed as a sequential arrangement of functions such as design, manufacture, R&D, marketing, and finance. According to a systems view of the world, the recent trend has been to view the enterprise system as a synergistic network of value-delivering business processes (Dong & Chen, 2001b; 2001a).

The concept of business process (BP) is tremendously emphasized in the Enterprise Engineering literature. The enterprise is viewed as a large and complex network of concurrent BPs executed by a set of functional entities (i.e. agents or resources) in order to attend costumer needs by generating value and therefore achieve business objectives (Goranson, 1999; Scheer, 1999b). A BP is formally defined a sequence (or partially ordered set) of enterprise activities, execution of which is triggered by some event and will result in some observable and quantifiable end result. They represent the flow of control of things happening in an enterprise. Consequently, they materialize in flows of materials, flows of documents, generation of services, conversion of perceived system state into feedback control actions, etc. (Vernadat, 1996).

The Enterprise Engineering Research Laboratory at Virginia Tech defines an enterprise as a human-made endeavor involving significant scope, complication and risk, which is carried out with the purpose of fulfilling customer needs for products or services. Structurally, the enterprise is considered a collection of two major subsystems: (1) the core system, and (2) the management system – a legacy of the prevailing Management Systems Engineering paradigm (see **Figure B.2**).

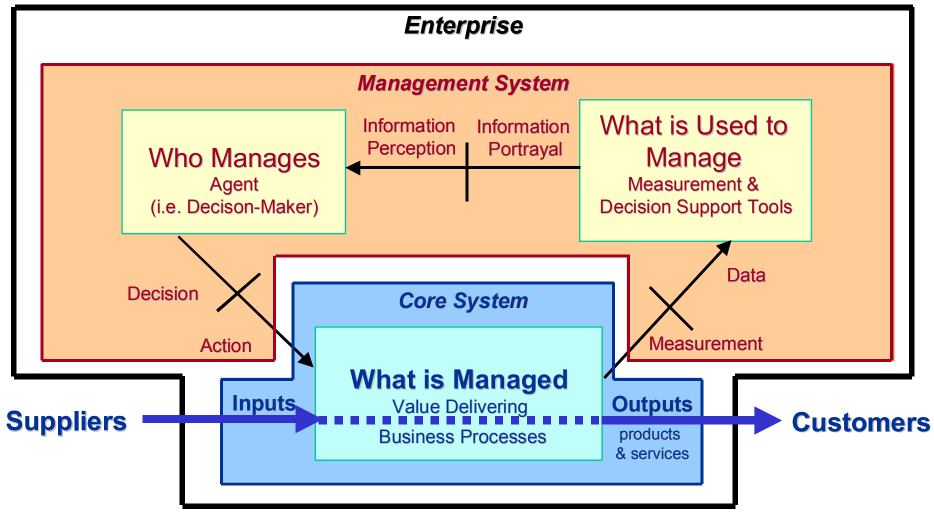


Figure B.2 *EERL's Definition of Enterprise.*
Source: adapted from Kurstedt (2000)

The core system transforms inputs (material, information, energy, knowledge, etc.) provided by suppliers into outputs (products and/or services) to customers by means of value delivering (i.e. core) business processes. These core BPs are overseen by a management system composed of management business processes. Management BPs convert measurements taken from core BPs into data, data into information, and information into decisions. These decisions are then fed back onto the core BPs in the form of regulatory and/or improvement actions (Kurstedt, 2000).

Liles and Presley (1996) present an alternative yet very similar definition. They advocate that, from a total enterprise perspective, BPs fall into three categories (see **Figure B.3**):

- *Category 1* - those processes which translate external constraints into internal constraints to set direction for the overall system;
- *Category 2* - those processes which acquire resources and manage assets; and
- *Category 3* - those processes that make use of such directions and resources in order to actually generate products or services.

According to this definition, an effective enterprise structure is one that promotes the harmonic interplay of BP components across all three categories; enabling the generation of value through the production of products and/or services so that overall enterprise goals can be achieved.

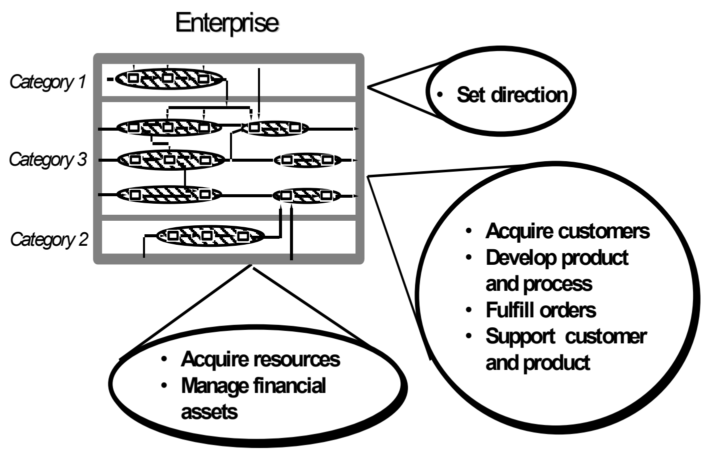


Figure B.3 *Liles and Presley's Definition of Enterprise.*
Source: taken from Liles and Presley (1996)

When considering enterprise domains that span beyond the boundaries of a single firm, the perspective presented by the Supply Chain Council may also be very useful. By means of a Supply Chain Operations Reference Model (SCOR), the connection between various firm's domains – as value is provided to the ultimate customer – becomes evident (Stephens, 1999). **Figure B.4** portrays a graphical simplification of SCOR.

Other reference enterprise architectures include: Purdue Enterprise Reference Architecture - PERA, Architecture for Integrated Information Systems - ARIS, Computer Integrated Manufacturing Opens Systems Architecture – CIMOSA (Petrie Jr., 1992; Bernus & Nemes, 1996; Vernadat, 1996).

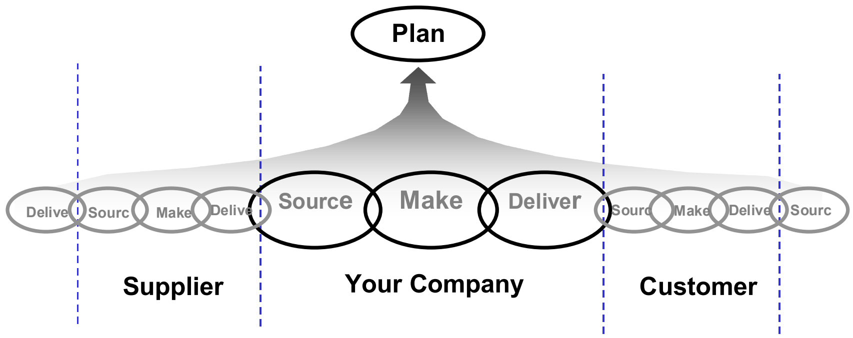


Figure B.4 *Symbolic Overview of SCOR - Supply Chain Operations Reference Model. Source: taken from Stephens (1999)*

B.1.4 Total System versus System Components

The fundamental assumption that an enterprise is a system that can be engineered has basic implications under the purview of the engineering process. To engineer an enterprise means that a need for the generation of products or services (or an opportunity for improvement in doing so) is transformed into a coherent collection of elements capable of actually physically fulfilling such need (or capable of fulfilling such need in an improved way).

Blanchard & Fabrycky (1998) emphasize that when attempting to engineer a system, the objective or purpose of the system must be explicitly defined and understood so that the system components can be selected to provide the desired output for each given set of inputs. The objective or purpose, therefore, makes it possible to establish a measure indicating how well the system performs given a certain arrangement of its components.

This distinction between the overall system of interest and its components is relevant for some important practical reasons. In any engineering effort it is natural to make use of existing subsystems as components of the system being created. For example, one could simply purchase a lathe when attempting to engineer a manufacturing cell (the enterprise system in question in this example). The performance of this lathe would naturally impact the performance of the cell, perhaps in undesirable ways due to its current limitations. An alternative would be to engineer (i.e. design and construct) the required lathe strictly according to this particular manufacturing cell's requirements, which could be interpreted as a mechanical engineering effort of its own derived from the original enterprise engineering effort.

This example illustrates that the decision of utilizing pre-existing components or engineering them from scratch is a natural and important part of any engineering effort and involves the consideration important trade-offs. When dealing with high performing enterprise systems, this issue is of particular criticality due to the scope and complexity of the considerations usually involved. Consequently, many pre-conceived components are likely to be used in these cases for reasons of feasibility.

This fact brings up an important issue widely discussed in the Systems Engineering literature. Blanchard & Fabrycky (1998), for instance, raise awareness for weaknesses in traditional bottom-up engineering design methods where the effort starts with a set of pre-existing elements and a system is largely created by means of synthesizing a combination of these system elements. The basic idea against the over reliance on this practice is that the desired ultimate functional need is very unlikely to be met on the first engineering attempt unless the system is simple – and changing the system structure later on could be a very arduous, if not impossible, task.

A successful enterprise development strategy will probably be one where top-down deployment of functional needs, by means of enterprise models (with details added gradually and consistently), is followed by bottom-up validations of content to ensure physical realizability (Vernadat, 1996; Bititci & Muir, 1997; Blanchard & Fabrycky, 1998). Therefore, when attempting to design and implement high performing enterprises, significant cooperation is expected to occur between enterprise engineers, other engineering specialties, as well as other professional specialties. For instance, starting from a conceptual level, the enterprise engineering effort could identify requirements for specific technological components such as the ones related to computer systems (Computer Engineering), facilities (Architecture and Civil Engineering), or machine tools (Mechanical Engineering).

B.1.5 The Enterprise Engineering Process

A generic engineering process usually consists of five main phases (Blanchard & Fabrycky, 1998):

- Identify customer need or opportunity for improvement;
- Design system (i.e. solution) to fulfill the need;
- Implement or build the system according to design;
- Use or operate system; and
- Dispose system – after eventually engaging in one or more cycles of transformation.

This approach is considered generic enough to be applied over any type of system. As such, these phases are aligned with the ones presented in the enterprise life cycle concept (illustrated in Figure 1.A), forming the basis of the Enterprise Engineering process – just as with other engineering specialties (see **Figure B.5**).

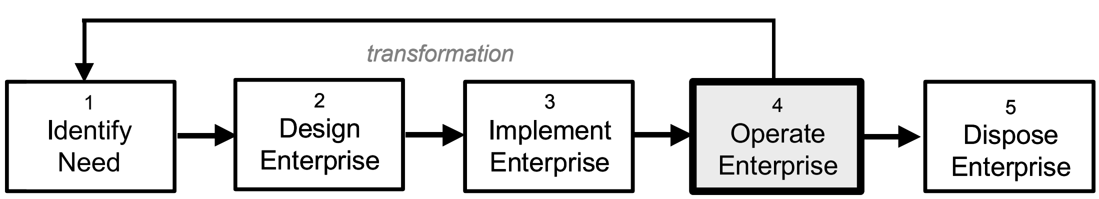


Figure B.5 *The Enterprise Engineering Process.*
Source: compiled from Blanchard & Fabrycky (1998), Vernadat (1996), and Scheer (1999b)

The enterprise engineering process begins with the **identification of a need** or an opportunity for improvement in the structure of a given enterprise, which will somehow impact the ability of delivering products or services to a final customer. If the enterprise does not yet exist, this effort usually takes the form of entrepreneurial activities (e.g. a high-tech start up, or a new plant dedicated to the production of an innovative product just launched by a large corporation). However if the enterprise is already in place and under operation, basically requiring some sort of adjustment in the current structure, the nature of the activities following this first phase is different and is often referred to as enterprise change or transformation.

In either case, once the need is identified, the next step is to **design** the new or improved enterprise structure. As mentioned before, this effort involves generating design choices and selecting among them according to some decision-making criteria that make use of performance indicators. This task is accomplished in a number of steps that varies according to the complexity of the case and the design paradigm adopted. The design of a complex system usually starts from a conceptual description capable of representing the system in its entirety, identifying major functions to be performed, as well as assessing logical and mathematical relationships that define the expected behavior. These conceptual considerations are then gradually followed by physical descriptions where the supporting technology is allocated to the identified functions, creating a description that can actually be used for construction or implementation purposes.

Naturally, the design phase is therefore followed by the **implementation** phase where the system is physically constructed according to the plans, guidelines or blueprints provided by the design phase. At the end of the implementation phase the system is ready to be used or operated, therefore marking the beginning of the **operation** phase. Usually, once the enterprise is brought into being for the first time, the operation phase is constantly taking place, and new cycles of need identification, redesign, and implementation may take place in parallel. This phenomenon reflects the fact the enterprises structures are usually “born” immature or incomplete, therefore requiring constant adaptation as it grows, and just like a living system, extremely rarely stop functioning until its “death” – even when changes or repairs need to be put in place.

It is a fact that for certain types of human-made systems the original structural configuration created does not change during the entire operational life. However in the case of an enterprise, as just mentioned this is rarely true. Examples of events requiring adaptation of enterprise structures include: launching of a new product, recognition of the need for better service quality and lower costs, a significant cultural change, a new constitutional law, a natural disaster, etc. In either case, these events promote parallel cycles of redesign and implementation as described in Figure 1.2. These cycles are, in fact, the mechanisms behind the ability of enterprises to adapt themselves to changing conditions in their environment and survive (Goranson, 1997; Rentes et al., 1999).

Cycles of transformation may repeat for many years, as has been observed in traditional firms. Certain industries, however, have shorter life as an indication of more dynamic or perhaps higher clockspeed conditions (Fine, 1998). In any case, the life of an enterprise is not expected to be infinite and events like a merger, or a purchase by a bigger enterprise, or even plain bankruptcy mark the **disposal** phase.

B.1.6 A Deeper Look at the Enterprise Design Phase

A structured enterprise design task begins with a clear identification of a need (or improvement opportunity), hereafter referred to as “organizational need”. Once this is accomplished, the design effort as advocated in the systems engineering literature gradually and systematically promotes the translation of the organizational need into the description of an enterprise implementation description (see **Figure B.6**).

This translation of need into a solution in the context of enterprise systems is a complex activity involving creativity and decision-making. In complex systems, designers can rarely ensure an optimal solution. The excerpts below further illustrate these ideas.

> *Design is the creative process by which our understanding of logic and science is joined with our understanding of human needs and wants to conceive and refine artifacts that serve specific human purpose. Design is central to the practice of systems engineering and systems engineers understand that design is a creative, iterative, decision-making process (White Jr., 1998 p.285).”*

> *“The system design process translates the customer’s needs into a buildable system design. It requires selecting subsystems from an allowable set and matching the interfaces between them. Designs that meet the top-level input and output requirements are tested to see how well they meet the system’s performance and cost goals. This paper proves that the System Design Problem is NP-complete by reduction from the Knapsack Problem, which is known to be NP-complete. The implication of this proof is that designing optimal systems with deterministic, polynomial time procedures is not possible. This is the primary reason why engineers do not try to produce optimal systems: They merely produce designs that are good enough (Chapman et al., 2001 p.222).”*

Conducting initial analysis is frequently the first step in the design enterprise phase. This involves defining the functional boundaries of the system, identifying customers and what is valued by them, outputs, inputs, and suppliers. This effort also requires an understanding of the very basic mission or purpose of the system as well as main strategies and market conditions. Basically, at this point, sufficiently detailed understanding of the current structure (if one exists) is sought.

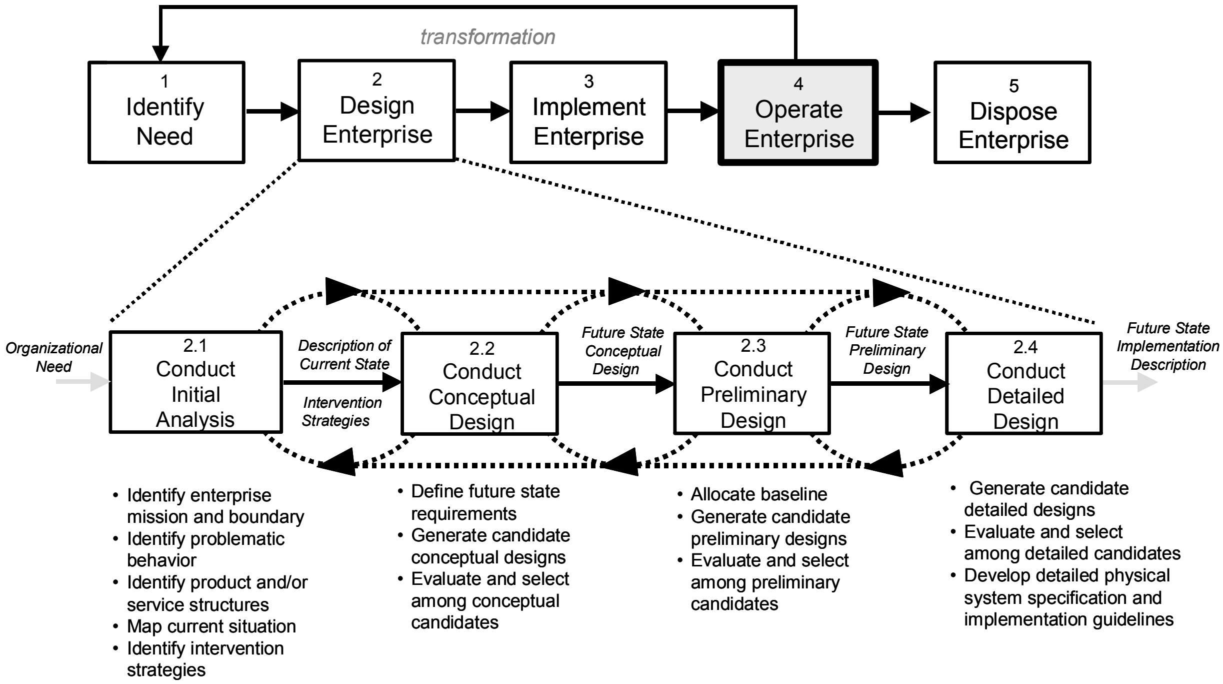


Figure B.6 *Generic Enterprise Design Sequence From a Systems Engineering Perspective. Source: compiled from Blanchard & Fabrycky (1998), Pressman (1995), Rentes (2000), Scheer (Scheer, 1999b; 1999a), Sousa et al. (2003), Sterman (2000), and Vernadat (1996)*

Because most enterprises do not maintain appropriate or updated documentation regarding its operations, this step may frequently require the generation of current state descriptions or models. The degree of detail utilized in the generation of these models may vary depending on the situation and purpose of the intervention. Current state models support the analysis of the current system and help promote insight into causes of problems. As such, they serve as an aid in the creative process of defining possible intervention actions. These desired interventions are then converted into future state system requirements, which serve as a basis for the generation of alternative candidate designs.

As such, this initial analysis effort is followed by the specification and selection of the most appropriate **conceptual design** alternative according to decision rules and specified performance criteria prevalent in the design paradigm adopted. This may often be accomplished using simulations due to the complex relationships among system components. The selected alternative is then further refined, generating even more specific design alternatives. This procedure is conducted until system functions have been allocated and a satisfactory preliminary solution (i.e. a **preliminary design** of the enterprise system) has been developed.

At this point, specific characteristics of physical infrastructure need to be added to the specification, i.e. to the **detailed design**. For instance, consistent technological descriptions of the infrastructure that enables material and information flows need to be generated. Consequently, this is the step where physical layouts, resources, database structures, software interfaces, sensors, actuators, etc. are specified in detail. Quite possibly, challenges faced at this and previous steps will require the revision of earlier steps, creating a cycle that promotes bottom-up validations in order to guarantee physical realizability. At this point in the design enterprise phase, enterprise models will likely be of considerable complexity and involve numerous combined modeling methods and dimensions.

The importance of a good design approach is high. The impact of design decisions on the life of the system is direct. For instance, the evaluation of past experiences regarding the development of complex systems shows that most of the problems identified have been a direct result of not applying a disciplined top-down engineering approach in meeting the desired objectives. In many cases, the overall requirements for the system were not defined well from the beginning. In such situations the impact can be very significant because it is at this early stage in the life/transformation cycle when decisions are made that can have a large impact on the overall effectiveness and cost of the system (Blanchard, 1991; Pressman, 1995; Blanchard & Fabrycky, 1998). **Figure B.7** illustrates this discussion.

The figure shows that, according to expert opinion, there can be a large commitment in terms of technology applications, the establishment of a system configuration and its performance characteristics, the obligation of resources, and potential life cycle cost at the very early stages of a development program (Blanchard & Fabrycky, 1998). It is at this very point, when system specific knowledge is limited, that major decisions are made pertaining to the selection of technologies, materials, suppliers, manufacturing processes, the maintenance approach etc. It is estimated that 50% to 75% of the projected life-cycle cost for a given system can be committed based on design decisions made during the early stages of conceptual and preliminary design. It is at this early stage that the implementation of Systems Engineering concepts is critical (Blanchard & Fabrycky, 1998). It is at this stage that Enterprise Engineering approaches could greatly improve through the adoption of more quantitative methods and tools (Fowler, 1999; Yu et al., 2000). It is for these reasons that this present work focuses on the utilization of adequate modeling languages and strategies at the conceptual design phase of Enterprise Engineering undertakings.

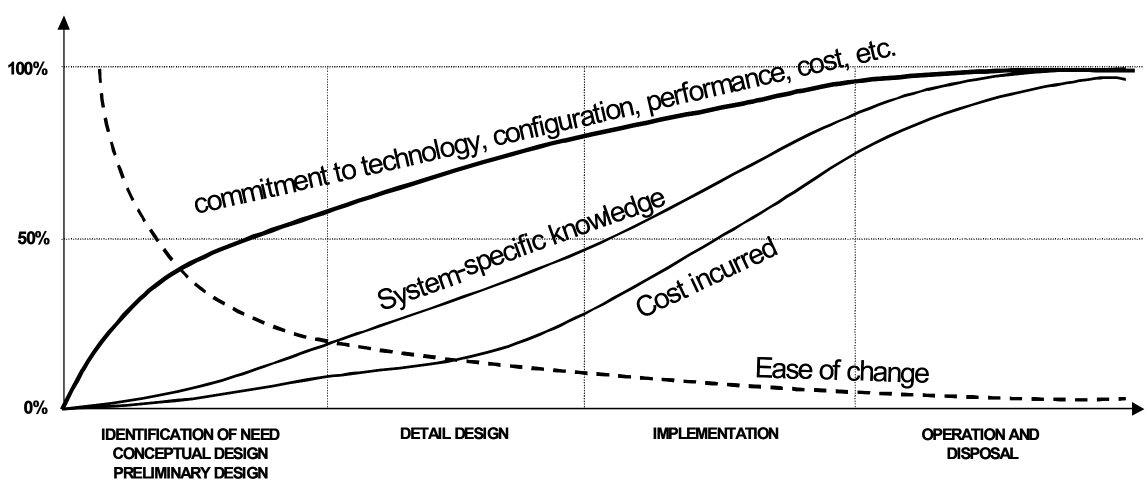


Figure B.7 *Impact of Early Design Decisions.*
Source: taken from Blanchard and Fabrycky (1998)

B.1.7 The Role of Modeling in Enterprise Design

Throughout the design process many complementary formalisms or methods of system representation (i.e. enterprise models) can be used. Models are used to support the representation and manipulation of the elements that compose the enterprise architecture and serve a variety of purposes. These include learning, communication of requirements, analysis of intricate phenomena, behavior assessment, decision-making, policy design, structure specification, etc.

Models are simplified abstractions of reality, making it feasible to deal with complex matters. There is no all-inclusive model of an enterprise. No matter how sophisticated a model is other models can always complement it. In other words a model, by definition, is just a representation of reality and therefore it can never be reality itself. The challenge is to choose the right modeling methodology for the purpose at hand.

Enterprise modeling, in general, concerns the manipulation of certain key aspects:

- WHAT – operations and objects being processed, leading to the creation of products and services and the ultimate delivery of value to the customer(s);
- HOW MANY/MUCH – quantity being processed as well as processing capacity;
- WHO – agents or resources making decisions and executing the operations;
- HOW – manner through which things are done, the method, the logical arrangement of operations that enable coordinated flows to take place;
- WHERE – logistics aspects; and
- WHEN – time.

From the rich and informal natural human language to formal and precise mathematics, an array of modeling options exists to handle all these facets or views, including symbolic, graphical, and semi-formal languages. As a result, enterprise modeling efforts may easily become overwhelming even considering the resources of large corporations if modeling purpose, scope, and strategy are not defined properly (Vernadat, 1996).

B.1.7.1 Master versus Unified versus Federated Modeling Paradigms

Three main modeling strategies are available to the enterprise designer: (1) master modeling, (2) unified modeling, and (3) federated modeling. The master modeling approach utilizes one type of modeling language as reference (or master model) and from there all other models are derived and instantiated in a straightforward manner. In a sense, the models created represent a customization to specific needs. This is probably the most utilized approach. The main drawback is that when changes need to be applied on the master model, the alterations need to be propagated to all instantiations of the model so that the consistency is maintained (Petrie Jr., 1992; Bernus & Nemes, 1996; Vernadat, 1996).

In the unified modeling approach, there can be as many local models as necessary and the semantic unification is guaranteed through the use of meta-models. Each model created do not necessarily need to be complete once, differently from the master modeling case, it is necessary to consider only those elements pertinent to the case at hand. Therefore, theoretically, it is always possible to implement a viable solution in a system in operation since changes are expected to propagate consistently across the models. However, the practical use of this modeling paradigm depends heavily on the adoptions of standards (Petrie Jr., 1992; Bernus & Nemes, 1996; Vernadat, 1996).

The federated approach is a more sophisticated paradigm that allows richer descriptions. Here as many local models as desired can be created. However once created, these models are maintained in their original configurations and new models are added. The nature of these models can vary significantly and only a weak connection among them is guaranteed. The interaction among them is possible by means of a federated mechanism, which works in the following way: when system A requests information from system B, system B needs to recognize the request based upon system's A own knowledge domain. In other words, system B needs to map the concepts utilized by system A according to the definitions used by A. Ontologies and knowledge sharing languages have been under development for this purpose. However, the feasibility of its use in industrial applications is still not clear (Petrie Jr., 1992; Bernus & Nemes, 1996; Vernadat, 1996).

Despite these conceptual classifications, comprehensive enterprise models for specific present-day practical purposes are usually based on multi-dimensional descriptions and require computational support. Common modeling dimensions include: functions, data/information, resources, organization, and control (Vernadat, 1996; Presley, 1997; Bond, 1999). To a large extent, the ability to deal separately with each dimension is crucial. Otherwise, the dimension categorization would have little value in reducing perceived complexity. On the other hand, one should also be able to synthesize these individual dimensions into a single modeling architecture so that integrated system analysis, synthesis, and evaluations can be performed (Presley, 1997; Whitman et al., 1998; Yu et al., 2000; Kamath et al., 2001).

B.1.7.2 Function Dimension

Functional modeling concerns the identification of the steps or activities that are carried on in an enterprise. A common approach adopted is the top-down functional decomposition where activities are “exploded” into increasing levels of detail creating a hierarchical functional description (Liles & Presley, 1996; Blanchard & Fabrycky, 1998). This dimension does not necessarily provide a good representation of sequence or logical order. This is frequently left to the control (or process) dimension (Vernadat, 1996; Presley, 1997; Scheer, 1999b).

B.1.7.3 Data/Information Dimension

The purpose of data/information modeling is to provide a representation of the information subsystem of an enterprise. Information systems are made of all pieces of data and information used/stored/processed for the needs of users and applications in an enterprise (Scheer, 1991; Petrie Jr., 1992; Vernadat, 1996; Kosanke et al., 1999; Sousa, 1999; Kurstedt, 2000).

B.1.7.4 Resources Dimension

Generally speaking, a resource is whatever is required to carry on an activity. Resources normally considered in modeling efforts include “human resources” as well as technological resources. Material and service related information are considered in the output dimension. The considerations of financial aspects as well as energy may also apply but the Enterprise Engineering literature does not seem to be as mature. Some argue that time can also be considered a resource and it is usually treated as an independent variable (Vernadat, 1996; Kamath et al., 2001).

B.1.7.5 Outputs Dimension

The output dimension includes a description of products and services generated by the activities and flowing along processes. The purpose of this dimension is to capture, respectively, the details of product structures in terms of material, as well as the details of service structures in terms of information elements (Vernadat, 1996; Scheer, 1999b; 1999a).

B.1.7.6 Organization Dimension

The organizational dimension is basically concerned with organizational structures, i.e. the distribution of roles and jobs in terms of responsibility and authority over the various individuals or groups. This dimension has a strong relationship with the management processes. Perhaps the most common output of an organizational modeling effort is the organizational chart, portraying assignment of responsibility and accountability over various domains (Vernadat, 1996; Whitman et al., 1998; Kurstedt, 2000).

B.1.7.7 Control (or Process) Dimension

The control (or process) dimension is definitely the most critical one for the purpose of performance evaluation. It involves a combination of elements from other dimensions creating a description capable of representing expected behavior over time. Activities (or functions) captured through the functional dimension are here arranged the form of networks of some sort.

By definition, every activity to be carried out needs a resource. Activities are also managed according to organizational agents. They transform input material and information into output material and information, generating products and services as it happens. Strictly speaking activities consume time and represent the conversion of energy into work through the utilization of resources. Models used in this dimension include, among other types, optimization and other decision support models from Operations Research and Control Theory, such as continuous as well as discrete-event simulation (Walizer & Wienir, 1978; Scheer, 1992; 1993; 1994; Vernadat, 1996; Wortmann et al., 1997; Blanchard & Fabrycky, 1998; Kosanke et al., 1999; Scheer, 1999b; 1999a).

In Enterprise Engineering, models are used in all phases of the life cycle. Among the many possible types of models an enterprise engineer could use, computer models are of special significance in the design phase as considerations regarding the relationship between architecture (structure) and performance (behavior) are made.

Overall, such computer models can be conveniently classified into two main groups: (1) optimization models, and (2) simulation models. Optimization models are **prescriptive** whereas simulation models are **descriptive**, i.e. optimization models provide a solution to a problem, whereas simulation models mimic the behavior of a system. Furthermore, simulation models may be classified into the following categories: role-playing games, stochastic modeling, discrete simulation, and system dynamics (Sterman, 1991).

Given that in this research there is specific interest in the relationships between structure and behavior of enterprises over time and at a strategic/conceptual level, system dynamics simulation modeling has been selected as the approach of choice. This approach is supported by Sterman (2000), Kumar and Vrat (1989), as well as Forrester (1961).

B.2 System Dynamics Modeling and Simulation

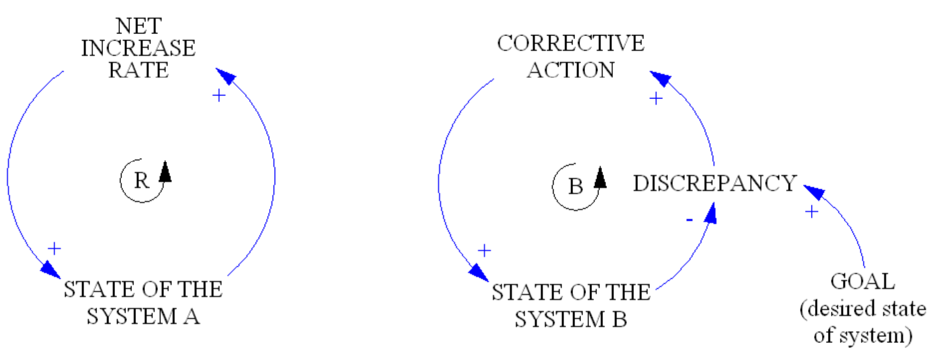
System dynamics modeling involves identifying and representing the feedback processes and other elements of complexity that determines the dynamics of a system. It is argued that all dynamics arise from the interaction of just two types of feedback loops: positive (or self-reinforcing) and negative (or self-correcting) loops.

B.2.1 Tools of System Dynamics

In order to represent loops, and determine their polarity (i.e. positive or negative) one may use the causal loop diagram.

B.2.1.1 Causal Loop Diagram (CLD)

Positive (i.e. + or **R**) loops tend to amplify whatever is happening in the system, whereas negative (i.e. – or **B**) loops oppose and balance change. See **Figure B.8** for a generic description of each type. Note that every link in a CLD is meant to represent what is believed to be a causal relationship between two variables and not correlations between these variables.



Notation	Description
X → + Y	All else equal, if X increases (decreases), then Y increases (decreases) above (below) what it would have been.
X → - Y	All else equal, if X increases (decreases), then Y decreases (increases) below (above) what it would have been.

Figure B.8 *Generic Structure of Reinforcing and Balancing Loops. Source: adapted from Sterman (2000)*

Consider the reinforcing loop in **Figure B.8**. The state of system A impacts its net increase rate, which impacts the state of the system A back. Basically, the larger the quantity representing the state of system A, the larger its net increase rate, the larger the quantity representing the state of system A, and so on. This is the structure capable of generating exponential growth. Imagine a savings account: the larger the balance, the larger the interest payment, the larger the balance, and so on. If let alone, the savings account balance can only increase, right?

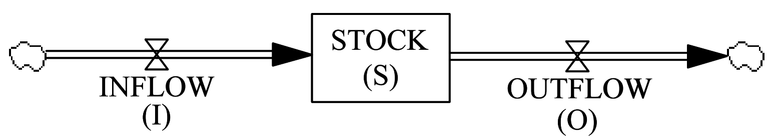
Now consider the balancing loop in **Figure B.8** By comparing the state of system B against the goal we have a quantification of discrepancy. The larger the discrepancy, the larger the corrective action, the closer system B gets to the goal. As system B gets closer to the goal the discrepancy diminishes and so does the corrective action until the system reaches the goal. Imagine a room temperature regulator. The further the room temperature is from the desired temperature the larger the discrepancy. The larger the discrepancy the larger the corrective action (in the form of heating or cooling depending on the case). As the temperature gets closer to the goal, the discrepancy diminishes and so does the corrective action. When the thermostat in a well insulated room is set for a comfortable temperature, if given enough time the room temperature will achieve the goal, right?

Sterman (2000) argues that although there are only two types of feedback loops and the dynamics of isolated loops can be easily interpreted, complex systems may contain thousands of both types establishing complex interactions with one another, altogether with time delays, nonlinearities as well as accumulations.

In situations when multiple loops interact it has been shown generally impossible to infer the dynamics simply by intuition, and when intuition cannot be used, computer simulation becomes a very powerful aid (Forrester, 1961; Senge et al., 1994; Sterman, 2001). However, in order to develop a simulation model it is also important to identify the main stocks and flows that take place in the system.

B.2.1.2 Stocks and Flows

CLDs are very useful in the beginning of a modeling project as an important aid in capturing mental models. One key limitation, however, is their inability to capture stocks and flows. Stocks and flows altogether with feedback loops are two fundamental concepts of system dynamics theory. Therefore, in addition to the causal linkages capture in CLDs, some other important elements of graphical representation are also used (see **Figure B.9**).



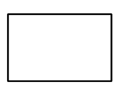
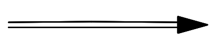
Notation	Description
	Stock
	Flow
	Valve (i.e. flow regulator)
	Source or sink (i.e. stocks outside model boundary)

Figure B.9 *Stock and Flow Structure.*

Stocks represent accumulations in the system and are important for dynamic considerations for a few reasons (Sterman, 2000 p.195).

- Stocks characterize the state of the system and provide the basis for actions.
- Stocks provide the system with inertia and memory.
- Stocks are the source of delays by accumulating the difference between the inflow to a process and its outflow.
- Stocks decouple rates of flow and create disequilibrium dynamics.

Consequently, stocks are measured in terms of *units.* Flows, on the other hand, promote change in stock levels and are measured in terms of *units over time*. For instance, the stock of finished inventory in a production facility is increased by the rate of production and decreased by the rate of consumption. As such, flows represent the actions or processes taking place in the system.

The distinction between stocks and flows can be difficult at times. However, if one could "freeze the scene" at a certain point in time, the stocks would be those things that can be counted – since, by definition, flows only exist in under a time reference. This procedure is known as the snapshot test (Sterman, 2000).

B.2.1.3 Model Equations

From a mathematical point of view, stocks are also known as *state variables* or *integrals; a*nd flows as *rates* or *derivatives.* From any stock and flow map we can construct a system integral and differential equations and vice-versa (Sterman, 2000).

Consider **Figure B.9**. A mathematical description of this stock and flow structure can be derived as follows:

$$I = I(t).dt \tag{B.1}$$

$$O = O(t).dt \tag{B.2}$$

$$S(t) = \int_{t_0}^{t} [I(t) - O(t)]dt + S(t_0) \text{ or} \tag{B.3}$$

$$\frac{dS(t)}{dt} = I(t) - O(t) \tag{B.4}$$

Each one of the functions describing flows can be further expanded to incorporate additional variables by means of pertinent relationships suggested by CLDs. From this perspective, system dynamics models are systems of nonlinear ordinary differential equations (Sterman, 2000 p.903).

B.2.1.4 Simulation

Given the usual complexity of this equation system, analytic solutions can generally not be found and the system behavior expressed in the model must be computed numerically by means of a simulation procedure. Such procedure requires the appropriate definition of a "delta time" quantity (i.e., dt) during which the rates are considered to remain constant. A commonly used numerical integration technique is the one proposed by the mathematician Leonhard Euler in the 18th century, known as *Euler integration.*

Given a set of initial conditions for the stocks and other exogenous variables, it is possible to calculate the initial values of the flows. Assuming the rates remain constant during the time step dt, the stock values for the beginning of the next time step can be calculated according to **equation B.5**. These new values of stock enable the calculation of rates for the next period and so on until the entire simulation period is considered.

$$S_{t+dt} = S_t + dt.(I_t - O_t) \quad \text{(B.5)}$$

The choice of time step influences the precision of the results. In Euler's method, as the time step is reduced, the accuracy of the assumption of constant rates between time steps increases – and therefore the integration error reduces. In the limit, when dt becomes an infinitesimal fraction of time, **equation B.5** is reduced to the continuous-time differential equation governing the dynamics of the system:

$$\lim_{dt \to 0} \frac{S_{t+dt} - S}{dt} = \frac{dS}{dt} = (I_t - O_t) \quad \text{(B.6)}$$

However, the smaller the value of dt the higher the number of calculations and the longer it will take. Additionally, when using computer support, there is a technological limitation worth mentioning: the smaller the time step the greater the round-off and truncation error, which arise because computers function with finite precision arithmetic. This means that simulation accuracy cannot be increased arbitrarily by reducing the time step. At some point, smaller time steps can cause round-off errors to actually offset the reduction in integration error. Therefore, selecting the time step for a simulation is a critical task involving a trade off between integration error against simulation cost and round-off error (Sterman, 2000).

B.2.1.5 Optimization

For the purpose construction and testing of dynamic hypothesis by means of system dynamics simulation models, optimization techniques can be used for two main purposes. Both involve manipulation of exogenous variables or constants: (1) to calibrate parameters, and (2) to optimize policies (Vensim, 2002).

Parameter calibration means refining the estimate of certain exogenous variables. In other words finding the values of model constants that make the model generate behavior curves that best fit the real world data.

Policy optimization, on the other hand, refers to selecting the best among alternative policy settings. For instance consider an inventory management policy based on targets for maximum and minimum desired inventory. How should these targets be set such that over time the payoff function (e.g. profit) is maximized?

Observation: In certain cases, optimization models may have to be embedded in the simulation model as part of the system structure represented. However, simulating such models might require additional computational capability than what commercially available tools can offer.

For instance, imagine the situation where there is the need to represent and simulate in Vensim® (the system dynamics simulation tool from Ventana Systems, Inc.) a scenario where production capacity is adjusted weekly by means of a linear program based policy. Also suppose that the simulation time horizon considered is 50 weeks, meaning that the linear program will need to be calculated 50 times, and each time according to the state of the system at the beginning of each week.

In this case, an external function (i.e. a software application) can be constructed to calculate this linear program using C or C++ for example and then compiled into a dynamic link library (DLL), which is loaded by Vensim® during simulation (Ventana, 2003). Consequently, by means of the combined use of Vensim® and the external function it becomes possible to evaluate the dynamic impact of this policy on the behavior of this system over time.

B.2.2 Structure and Behavior

An essential principle of system dynamics is that the structure of a system gives rise to its behavior (Forrester, 1961). As highlighted previously in this text, in the context of enterprise systems, a goal of system dynamics modeling is to improve understanding of the ways in which enterprise performance is related to its internal structure, including operating policies, and other factors within management control or influence. This understanding can then be used to implement changes in the current enterprise system's structure in such a way as to enable the generation of the desired behavior, measured according to a set of performance metrics.

In a modeling effort, defining the system boundary and the degree of aggregation are two of the most difficult steps (Sterman, 2000). The understanding of the typical structure-behavior relationships can be particularly useful in these situations. **Table B.1** presents some common modes of behavior and the corresponding dominant feedback structures that can be generating them.

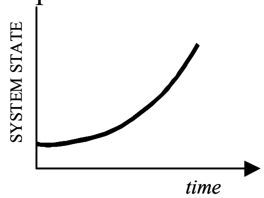
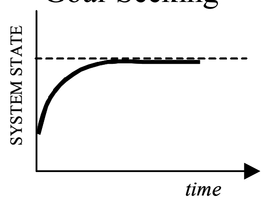
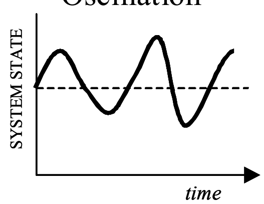
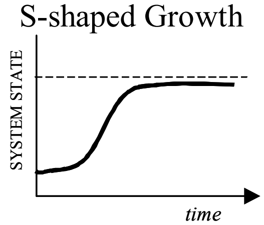
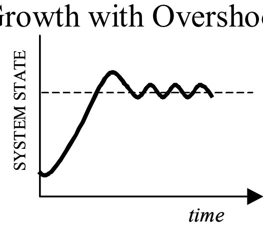
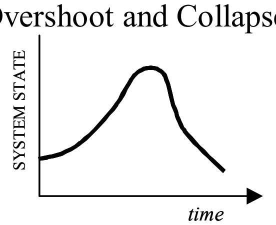
Other modes of behavior include stasis or equilibrium, randomness, and chaos. **Stasis** arises in situations when the dynamics affecting the state of the system are so slow that change is imperceptible, or because there are powerful negative feedback processes keeping the state of the system nearly constant even in the face of environmental disturbances. The classification of a behavior as **random**, however, is frequently an indicator of the inability to explain it. Although randomness might have physical meaning at a microscopic level in the arena of quantum mechanics, it is believed that it has insignificant impact in the behavior of macroscopic systems such as an enterprise. Therefore, simply referring to something as random at the enterprise level is likely to mean that the reasons for such variations are not known or understood. **Chaos** is a form of unstable oscillation and an important conclusion arises from the study of this phenomenon: while equilibrium may be locally unstable, any real system must be globally stable (Sterman, 2000).

B.2.3 Principles for Successful Use of System Dynamics

The successful use of system dynamics tools in the context of significantly complex undertakings requires the ability to: (1) plan in advance, (2) document well the work developed, and (3) to integrate past relevant experience. Especially in regards to the third item, Sterman (2000 p.79) provides some useful guidelines for the adequate utilization of system dynamics in general, but more specially for the context of business related projects.

- *"Develop a model to solve a particular problem, not to model the system;*
- *Modeling should be integrated into a project from the beginning;*
- *Be skeptical about the value of modeling and force the "why do we need it" discussion at the start of the project;*
- *System dynamics does not stand alone; use other tools and methods as appropriate;*
- *Focus on implementation from the start of the project;*
- *Modeling works best as an iterative process of joint inquiry between client and consultant;*
- *Avoid black box modeling;*
- *Validation is a continuous process of testing and building confidence in the model;*
- *Get a preliminary model working as soon as possible; add detail only as necessary;*
- *A broad model boundary is more important than a great deal of detail;*
- *Use expert modelers, not novices; and*
- *Implementation does not end with a single project."*

Table B.1 Common Behaviors and Corresponding Feedback Structures.

Common Mode of Behavior	Feedback Structure
Exponential Growth 	Positive feedback
Goal-Seeking 	Negative feedback
Oscillation 	Negative feedbacks with time delays
S-shaped Growth 	Nonlinear interactions of the fundamental feedback structures
Growth with Overshoot 	Nonlinear interactions of the fundamental feedback structures
Overshoot and Collapse 	Nonlinear interactions of the fundamental feedback structures

B.2.4 Modeling the Dynamics of Businesses

Over the past four decades, system dynamics has been used for a variety of purposes in various fields of study, including physics, chemistry, biology, ecology, sociology, economy and engineering. Interestingly, its origins in the late 1950s are intimately related to business applications as described by Forrester's *Industrial Dynamics* (Forrester, 1961; 1996). In fact, from a management point of view, the historical evolution of this practice is aligned with a wider movement towards adoption of a systems-oriented view of the world by the business community.

Nowadays a commonly accepted paradigm is that all types of enterprise systems convert inputs into outputs (i.e. generate service and/or goods) by means of a set of processes. These processes, also referred to as business processes or supply chain business processes, may involve a large number of activities, inventory locations, as well as decision making points.

The observation of behavior over time of a given set of enterprise processes encompassing various decision-making points usually highlights complex dynamic relationships. The excerpt below extracted from Forrester (1961 p.22) helps to illustrate this idea:

> *"The central core of many industrial companies is the process of production and distribution. A recurring problem is to match the production rate to the rate of final customer sales. It is well known that factory production rate often fluctuates more widely than does the actual consumer purchase rate. It has often been observed that a distribution system of cascaded inventories and ordering procedures seem to amplify small disturbances that occur at the retail level (...)"*

In this excerpt, Forrester refers to the need of controlling the production rate so that it matches the customer sales rate; a fundamental managerial need in any enterprise system. Furthermore, he also makes a critical observation related to the instability observed along typical supply chain business processes; in this case he refers to the demand amplification phenomenon nowadays referred to as the Forrester's effect or the bullwhip effect.

Wilding (1998) makes the point that competition in the future will not take place among individual organizations but among groups of firms, at the supply chain level. He advocates that instability arises as a result of he calls *the supply chain complexity triangle*, i.e. three interacting yet independent effects: deterministic chaos, parallel interactions, and demand amplification. According to this theory, serial and parallel interactions can occur between each echelon in a supply chain; and therefore stable equilibrium cannot be reached. Small perturbations will always prevent equilibrium from being achieved.

Towill (1996b) brings this discussion to the context of JIT. He points out that, in practice, as JIT is gradually implemented from the customer end of the process, it actually pushes the effects of unpredictability upstream in the supply chain.

Wilding (1998) argues that many lean production approaches to manufacturing do not rely on complex feedback systems. In fact, he suggests that focusing on the uninterrupted flow of material that matches the pull from the customer (a fundamental goal of lean production) can actually be interpreted as an effort to eliminate feedback and consequently the conditions to produce chaos. A big challenge, however, for organizations attempting to implement JIT related practices is to ensure that their systems are flexible and responsive enough to cope with the increased uncertainty that is likely to be experienced in the marketplace.

Fowler (1999) confirms some of Wilding's assertions and compares instability effects of JIT versus MRP. He advocates that JIT or pull production is a form of feedback control, while MRP or push production is a form of feedforward control. Feedforward systems are unstable by nature, and this characteristic can actually be an advantage in dealing with external instability.

Furthermore, it is also argued that feedforward systems – differently from feedback systems – by definition require a model of the system as a basis for regulatory decision-making. However, given the imprecise nature of models as simplifications of reality, pure feedforward systems are unlikely to succeed in all but the very simple cases.

Feedback systems, on the other hand, are simpler and do not necessarily require a model of the system itself as part of its infrastructure. The drawback in this case is that it may be impossible to produce simultaneously responsive yet stable behavior using feedback control (one reason why JIT requires stable demand conditions). Nevertheless, the fact is that the complexity of modern operations is likely to require combinations of feedback and feedforward control in most cases (Fowler, 1999).

Mason-Jones and Towill (1998) approach the supply chain instability problem from a complementary perspective related to lead times. The idea is that lead-time through a supply chain consists of two components: the information pipeline and the material flow. The argument is that although much improvement can be achieved from material flow cycle time reduction, reacting faster to a slow order does not significantly improve control of the supply chain dynamic response to a change in consumer demand. To improve control of the highly undesirable overshot effects, the supply chain has to provide undistorted order information fast. In fact, reducing time delays in information flows is well known among system dynamicists as a preferred route to reduce supply chain instability and improve performance (Towill, 1996b).

Sterman (2000) reinforces these ideas by affirming that not only amplification, but also oscillation and phase lag are pervasive in supply chains. These persistent and expensive undesirable dynamics represent the combined effects resulting from the stock management structure of each decision making point in the process. As enterprises worldwide strive for a competitive edge, a deeper comprehension of how its role and place affect the behavior of the entire supply chain is of particular importance. Therefore, before further analyzing value delivering business processes that span beyond the boundaries of single firms, it may be insightful to focus on the managerial structure that regulates material flow at each individual decision making point along the way.

B.2.5 The Stock Management Problem

Typically at each decision making point along the path of a manufacturing business process, management mechanisms attempt to balance production with orders by considering both the inventory and the resources at hand. Such structure always involves negative feedbacks and according to the system dynamics paradigm, this problem can be divided into two parts: (1) the stock and flow structure of the system – representing the core physical process, and (2) the decision rule used by managers – which is embedded in the management process that regulates the core process. See the example in **Figure B.10**.

In this example, suppose that a manager has control over the inflow rate to a stock as he or she tries to compensate for changes in the outflow. Assume there is no delay in acquiring new units. The *stock* (S) at time t can be represented in the following way:

$$S = \int_{t_0}^{t} (AR - LR) + S_{t_0} \qquad \text{(B.7)}$$

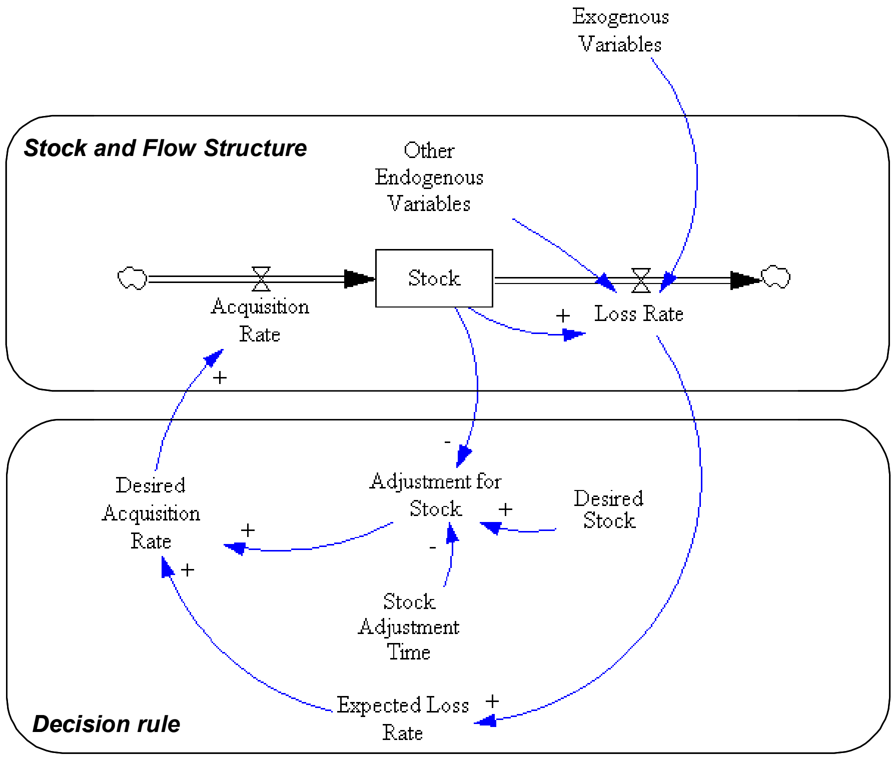


Figure B.10 *Generic Structure of a Stock Management System. Source: adapted from Sterman (2000)*

Losses from *stock* may take place in the form of usage, depreciation, etc. Therefore, it depends on the value of the stock itself, i.e. as the value of the stock approaches zero losses must also approach zero (which is indicated by the positive causal link between *stock* (S) and *loss rate* (LR)). Losses could also be influenced by other endogenous (E) as well as exogenous (X) variables. Therefore, *loss rate* is considered a function of S, E, and U:

$$LR = f(S, E, U) \tag{B.8}$$

Assuming there is plenty of capacity and that consequently there are no significant delays in the acquisition of new units, the *acquisition rate* (AR), can be modeled as the maximum between zero and *desired acquisition rate* (DAR), ensuring that the acquisition rate remains nonnegative:

$$AR = \max(0, DAR) \tag{B.9}$$

Desired acquisition rate (DAR) reflects the hypothetical manager's decision-making process illustrated in this case. DAR can be defined as the sum of *expected loss rate* (ELR) and *adjustment for stock* (AS) as follows:

$$DAR = ELR + AS \tag{B.10}$$

The consideration of *expected loss rate* (ELR) instead of the actual *loss rate* (LR) is recognition of the fact that measurement instruments cannot measure instantaneous rates of change but only average rates over some finite interval. However, if the this measurement delay is short relative to the dynamics of interest in the model it may be assumed that *expected loss rate* (ELR) equals actual *loss rate* (LR):

$$ELR = LR \tag{B.11}$$

Furthermore, assuming that the adjustment for stock is linear according to the discrepancy between *desired stock* (DS) and *stock* (S), it can be defined in the following way, where SAT is the *stock adjustment time:*

$$AS = \frac{DS - S}{SAT} \tag{B.12}$$

Note that a negative loop is created (*adjustment for stock → desired acquisition rate → acquisition rate → stock)*, acting as a goal seeking mechanism where the goal sought is *desired stock.* This completes the description of this particular inventory control mechanism system. Many other variants are possible and are indeed practiced; especially in what concerns the logic used in the manager's decision-making process.

Furthermore, from the perspective of flow taking place across one or more manufacturing facilities, various such structures might interact to generate the (desirable or undesirable) dynamics observed. As more inventory control structures are considered, the complexity increases, the ability to infer the dynamics of the system by intuition decreases, and formal expert knowledge regarding the production planning and control infrastructure becomes essential.

Appendix C – Hypothetical Case Model Tests And Evaluations

Details about hypothetical model tests are presented in this appendix. The first part of this text focuses on the hypothetical AS-IS model. These tests are intended to build trust in the model and create a baseline for the purpose of this research. The second part is directed at analyzing alternative TO-BE scenarios derived from incorporation of the proposed policies.

C.1 Hypothetical AS-IS Model

The results of some key model tests as suggested by Sterman (2000) are presented in this section.

C.1.1 Boundary Adequacy

Are the important concepts for addressing the problem endogenous to the model?

Key concepts for addressing the problematic behavior of a value stream include: demand rate, demand leveling, production rate, production pull, kanbans, production capacity, work-in-process inventory, and backlog. These are all included in the model.

Demand rate is exogenous and reflects a choice of where to set the boundaries for this study. The assumption is that the dynamics of the systems do not feed back to significantly affect the demand rate. Production capacity is set partially endogenously (directly interpreted from the average demand incurred in the previous estimation period) and partially exogenously (as an ad-hoc choice of how lean the system owners want it to be). All other above referred concepts are endogenous to the model.

Does the behavior of the model change significantly when boundary assumptions are relaxed?
The behavior could indeed change significantly if the assumption concerning fixed production capacity during the estimation period was relaxed. This would affect the backlog level. Such a relaxation, in practice, could happen in various ways. E.g., in the form of additional hours of work or even the allocation of more or less production power within one or more planning periods.

A change in this assumption would imply the incorporation of feedback loops that endogenously regulate the level of resources available in the system. However, the current study focus on fixed capacity situations within estimation periods, and therefore the current model is adequate.

C.1.2 Structure Assessment

Is the model structure consistent with relevant descriptive knowledge of the system?
Yes. The structure of the model is compatible with descriptive knowledge of value streams found in the literature and expert opinion. The hypothetical situation resembles the purchase-assembly-shipment structure found in discrete parts manufacturing companies.

In between processes there are inventory locations or "supermarkets". The supermarket levels, the withdrawal of parts, as well as the subsequent processing are regulated according to kanban concepts. Consequently, production takes place in a pull mode as a reaction to present demand only. In other words, resources are not allowed to stay busy for the purpose of maintaining high utilization levels.

Is the level of aggregation appropriate?

Yes. The scope and level of aggregation are appropriate. The selection of a door-to-door scope enables the clear identification of responsibility and accountability. The focus on the flow of generic product family units without distinguishing variants is also appropriate. This enables the identification of units flowing across the system according to value stream theory, without going into unnecessary detail required to exploring the current research questions.

Additionally, the upstream and downstream processes are referred to generically. This is appropriate because the specific physical nature of tasks (as described by flows) need not be specified in this hypothetical case. These production tasks could actually represent purchasing, assembly, or shipment activities of any specific nature; the characteristic that matters for the hypothetical dynamic assessment is the rate at which they can take place. Furthermore, each kanban (card) is considered individually, which enables the emulation of the regulatory mechanisms with good precision.

Does the model conform to basic physical laws such as conservation laws?

Yes. The model conforms to the classical law of conservation of matter. Inventories can only assume nonnegative values and vary only according to clearly specified replenishment and withdrawal processes. The levels of work-in-process inventory always reflect the net difference between inflows and outflows. There is a clear origin and a clear destiny to all units flowing across the system.

Additionally, the model does not assume that information signals can travel at an infinite speed, i.e. with zero delay. Therefore, the physical movement of kanban cards is also considered in itself a process that consumes time.

Do the decision rules capture the behavior of the actions in the system?

The main actors in the system are: (1) the value stream manager, and (2) the value stream executor(s). The manager's decision rules involving demand leveling and scheduling have been incorporated as suggested by value stream mapping theory. The manager's role in terms of fixed capacity allocation at the beginning of each estimation period has also been incorporated.

The executor's actions from a dynamic assessment point of view basically involve the triggering of flows only as dictated by demand occurrences. This behavior has been captured in the decision rules involved in specifying the kanban mechanism.

C.1.3 Dimensional Consistency

Is each equation dimensionally consistent without the use of parameters having no real world meaning?

Yes. Equations are dimensionally consistent and have been verified using Vensim®'s dimensional consistency check. Only parameters with real world meaning have been used.

Observation: the use of the SMOOTH function has been causing a dimensional consistency error in two variables. The affected variable definitions have been checked and are apparently correct. The numerical values of the variables are correct. This error might be related with an internal problem with the software. In any case, the affected variables do not serve as inputs for any other variables in the system, and therefore this condition is not affecting the simulation results.

C.1.4 Parameter Assessment

Are the parameter values consistent with relevant descriptive and numerical knowledge of the system?

Being a hypothetical model one could argue that almost any choice of parameters would be appropriate. Nevertheless parameters were set in such a way as to reflect typical real systems characteristics. For the sake of simplification the demand rate as well as process rates were set in the magnitude of single minutes per parts; and the container size is set to carry a few units.

Do all parameters have real world counterparts?

Yes. All exogenous variables have real world correspondents.

C.1.5 Extreme Conditions

Does each equation make sense even when its inputs take on extreme values?

The only possible problem would be with the rate of kanban movement given that it is calculated according to the *time step*. Therefore, in the limit, if *time step* assumed the value of zero then the number of kanbans moved would be infinite. However, since this is not an option because it goes against the concept of numerical integration, the model is safe. In fact, any positive value of *time step* is utilizable.

Does the model respond plausibly when subjected to extreme policies, shocks, and parameters?

Under the extreme condition of zero demand, no production takes place. If production capacity is set to zero, then no assembly or purchasing takes place. If the number of kanbans in the system is set to zero, no production takes place either. In summary movement of parts only takes place if there is actual demand and production capacity. These are indeed expected characteristics of pull systems.

C.1.6 Integration Error

Are the results sensitive to the choice of time step or numerical integration method?

Since TIME STEP has been used to approximate discrete characteristics of the system, the Euler numerical integration method has to be used given the mathematical requirements involved (Sterman, 2000, pp. 909).

C.1.7 Behavior Reproduction

Does the model reproduce the behavior of interest in the system (qualitatively and quantitatively)?

The model has been satisfactorily reproducing key behaviors of interest, and therefore matches quite well the generic case described in the literature. For instance, the amount of *WIP* in the system has been varying in a range that is proportional to the *number of kanbans* in the system. When *demand* levels are above production capacity, backlog increases. Quantitative statistical assessments cannot be performed due to the fact that this is a hypothetical case and therefore there is no historical data to serve as a reference of previous behavior.

Does it endogenously generate the symptoms of difficulty motivating the study?

Yes.

Does the model generate the various modes of behavior observed in the real system?

Yes. The model successfully reproduces the problematic mode of behavior that originated this investigation. **Figure C.1** represents a simulated baseline: *Quantity on order from external customer (QOC)* increases abruptly immediately after the demand spike and then gradually returns to the equilibrium condition. The simulated time horizon is one month or 9600 minutes, i.e., twenty 480-minute planning periods.

Work in process (WIP) remains almost undisturbed due to the fact that the spike tends to affect only the inventory immediately downstream from the pacemaker (i.e., the *Protective decoupling inventory (PDI)*); upstream from the pacemaker the demand signal is smoothed out via the load-leveling box. Fulfillment rate goes briefly to zero in between planning periods. This happens for three reasons: (1) process works at full capacity until fulfilling the target and then system becomes idle, (2) the amount of demand allocated to the shop floor is based on an estimate and therefore there is likely to be estimation errors, and (3) even in situations where there is backlog, the fulfillment rate goes to zero for just a few moments in between planning periods – the third reason is a feature of the simulation logic and although it causes graphically appealing fact that the curve to goes to zero temporarily, its effect is insignificant in the simulation results.

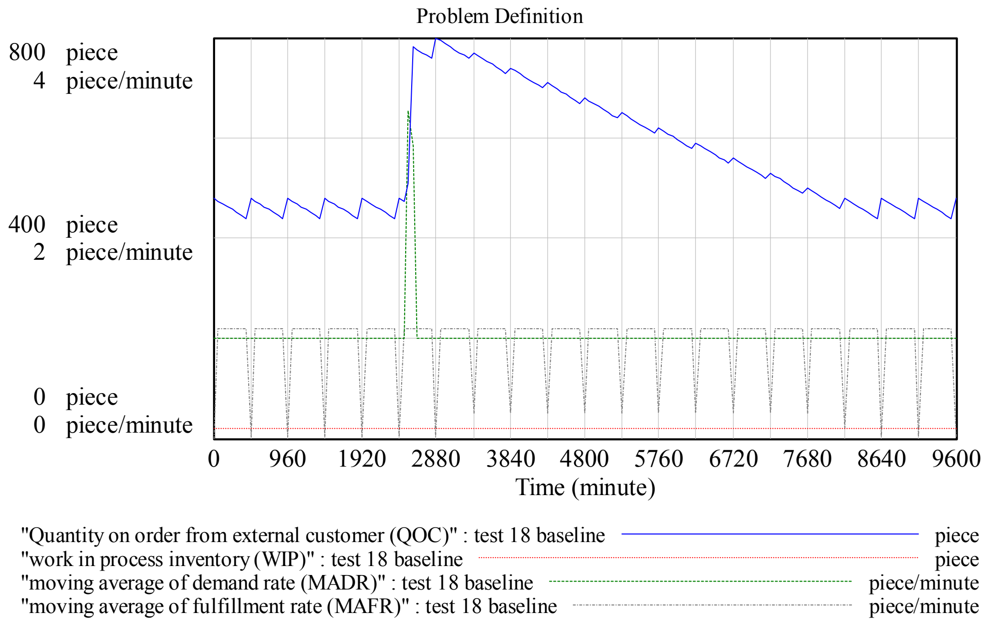


Figure C.1 *Baseline Under Demand Spike.*

C.1.8 Family Member

Can the model generate the behavior observed in other instances of the same system?

Yes. The model as it is can be easily adapted to represent other value streams with similar or more complex process structures. It constitutes a building block in the sense that connecting replications of this model enable the representation of value stream with broader scopes (e.g., total value streams). However, further development will be necessary for the model to represent multi-product families as well as mixed pull situations.

C.2 Hypothetical TO-BE Model

Given the AS-IS baseline established, alternative TO-BE scenarios where generated based on the three proposed flow control policies. The scenarios generated out of policy 1 and policy 2 required the specification of related management structures.

C.2.1 Policy 1 Scenario

This first scenario requires change in the physical structure of the value stream to incorporate an additional management process capable of monitoring the *Demand overflow (DO)* level and adding or removing kanban cards from the PDI kanban system at the beginning of each planning period.

C.2.2 Policy 2 Scenario

This second scenario utilizes the already existing demand allocation infrastructure. However it requires a new specification of the decision rule in mathematical terms. The following variables required a new analytical specification: *RDAPP(t): rate of demand allocation to next planning period*, *RDO(t): rate of demand overflow*, and *RDOCPP(t): rate of demand overflow allocation to current planning period.* Such analytical description is provided next.

$RDAPP(t)$:
IF MODULO (*clock time, SPP***)**=0
THEN IF $DICPP(t) > 0$
---------**THEN IF** $DICPP(t) + DO(t) + DFCPP(t) < (ADRPEP + 1 \cdot SDDPEP) * SPP$
------------------**THEN** $RDAPP(t) = DICPP(t)/dt$
------------------**ELSE IF**
-------------------------- $DICPP(t) + DO(t) + DFCPP(t) \geq (ADRPEP + 1 \cdot SDDPEP) * SPP$
--------------------------**AND**
-------------------------- $DICPP(t) + DO(t) + DFCPP(t) \leq (ADRPEP + 3 \cdot SDDPEP) * SPP$
--------------------------**THEN IF** $DO(t) \geq ADQPP(t) + BS(t) - DFCPP(t)$
-----------------------------------**THEN** $RDAPP(t) = 0$
-----------------------------------**ELSE IF**
-- $DICPP(t) \leq ADQPP(t) + BS(t) - DFCPP(t) - DO(t)$
--**THEN** $RDAPP(t) = DICPP(t)/dt$
--**ELSE**
-------------------------------------- $RDAPP(t) = (ADQPP(t) + BS(t) - DFCPP(t) - DO(t))/dt$
--------------------------**ELSE IF** $DO(t) \geq ADQPP(t) + PDI(t) - DFCPP(t)$

------------------------------------**THEN** $RDAPP(t) = 0$
------------------------------------**ELSE IF**
------------------------------------ $DICPP(t) \leq ADQPP(t) + PDI(t) - DFCPP(t) - DO(t)$
--**THEN** $RDAPP(t) = DICPP(t)/dt$
--**ELSE**
------------------------------------ $RDAPP(t) = (ADQPP(t) + PDI(t) - DFCPP(t) - DO(t))/dt$
---------**ELSE** $RDAPP(t) = 0$
ELSE $RDAPP(t) = 0$ (C.1)

$RDO(t)$:
IF MODULO (*clock time, SPP***)**=0
THEN IF $DICPP(t) > 0$
---------**THEN IF** $DICPP(t) + DO(t) + DFCPP(t) < (ADRPEP + 1 \cdot SDDPEP) * SPP$
-----------------**THEN** $RDO(t) = 0$
-----------------**ELSE IF** $DICPP(t) + DO(t) + DFCPP(t) \geq (ADRPEP + 1 \cdot SDDPEP) * SPP$
-------------------------**AND**
------------------------- $DICPP(t) + DO(t) + DFCPP(t) \leq (ADRPEP + 3 \cdot SDDPEP) * SPP$
-------------------------**THEN IF** $DICPP(t) + DO(t) + DFCPP(t) \leq ADQPP(t) + BS(t)$
----------------------------------**THEN** $RDO(t) = 0$
----------------------------------**ELSE IF** $DO(t) \geq ADQPP(t) + BS(t)$
--**OR** $DFCPP(t) \geq ADQPP(t) + BS(t)$
--**OR** $DO(t) + DFCPP(t) \geq ADQPP(t) + BS(t)$
--**THEN** $RDO(t) = DICPP(t)/dt$
--**ELSE IF** $DICPP(t) \leq ADQPP(t) + BS(t) - DFCPP(t) - DO(t)$
--**THEN** $RDO(t) = 0$
--**ELSE**
------------------------- $RDO(t) = [DICPP(t) - (ADQPP(t) + BS(t) - DFCPP(t) - DO(t))]/dt$
-------------------------**ELSE IF** $DICPP(t) + DO(t) + DFCPP(t) \leq ADQPP(t) + PDI(t)$
----------------------------------**THEN** $RDO(t) = 0$
----------------------------------**ELSE IF** $DO(t) \geq ADQPP(t) + PDI(t)$
--**OR** $DFCPP(t) \geq ADQPP(t) + PDI(t)$
--**OR** $DO(t) + DFCPP(t) \geq ADQPP(t) + PDI(t)$
--**THEN** $RDO(t) = DICPP(t)/dt$
--**ELSE IF**
-- $DICPP(t) \leq ADQPP(t) + PDI(t) - DFCPP(t) - DO(t)$
--**THEN** $RDO(t) = 0$
--**ELSE**
------------------------ $RDO(t) = [DICPP(t) - (ADQPP(t) + PDI(t) - DFCPP(t) - DO(t))]/dt$
--------- **ELSE** $RDO(t) = 0$
ELSE $RDO(t) = 0$ (C.2)

$RDOCPP(t)$:
IF MODULO (*clock time, SPP***)**=0
THEN IF $DO(t) > 0$
---------**THEN IF** $DICPP(t) + DO(t) + DFCPP(t) < (ADRPEP + 1 \cdot SDDPEP) * SPP$
-----------------**THEN** $RDOCPP(t) = DO(t)/dt$
-----------------**ELSE IF**
-------------------------- $DICPP(t) + DO(t) + DFCPP(t) \geq (ADRPEP + 1 \cdot SDDPEP) * SPP$
-------------------------- **AND**
-------------------------- $DICPP(t) + DO(t) + DFCPP(t) \leq (ADRPEP + 3 \cdot SDDPEP) * SPP$
--------------------------**THEN IF** $DICPP(t) + DO(t) + DFCPP(t) < ADQPP(t) + BS(t)$
-----------------------------------**THEN** $RDOCPP(t) = DO(t)/dt$
-----------------------------------**ELSE IF**
-- $DICPP(t) + DO(t) \geq ADQPP(t) + BS(t) - DFCPP(t)$
--**THEN IF** $DO(t) < ADQPP(t) + BS(t) - DFCPP(t)$
---**THEN** $RDOCPP(t) = DO(t)/dt$
---**ELSE**
--- $RDOCPP(t) = (ADQPP(t) + BS(t) - DFCPP(t))/dt$
--**ELSE** $RDOCPP(t) = DO(t)/dt$
--------------------------**ELSE IF** $DICPP(t) + DO(t) + DFCPP(t) < ADQPP(t) + PDI(t)$
-----------------------------------**THEN** $RDOCPP(t) = DO(t)/dt$
-----------------------------------**ELSE IF**
-- $DICPP(t) + DO(t) \geq ADQPP(t) + PDI(t) - DFCPP(t)$
--**THEN IF** $DO(t) < ADQPP(t) + PDI(t) - DFCPP(t)$
---**THEN** $RDOCPP(t) = DO(t)/dt$
---**ELSE**
--- $RDOCPP(t) = (ADQPP(t) + PDI(t) - DFCPP(t))/dt$
--**ELSE** $RDOCPP(t) = DO(t)/dt$
---------**ELSE** $RDOCPP(t) = 0$
ELSE $RDOCPP(t) = 0$ (C.3)

C.2.3 Policy 3 Scenario

The implementation of this third scenario is straightforward since the changes required involve only a new setting for a few exogenous variables.

Appendix D – Evaluation Of Demand Profiles

D.1 Introduction

The goal of this text is to identify potential demand profiles and evaluate their impact on the performance of typical pull systems. Performance is indicated by means of the production rate as well as demand not fulfilled. An important assumption is that demand is exogenous, i.e., demand can be treated as an independent variable since the performance of the pull system does not feedback to affect its behavior.

In regards to the demand profiles, first of all two possibilities are considered: (1) demand is known in advance, and (2) demand is unknown. This study considers that demand is unknown. The idea is that if demand is known in advance, it can either be level loaded in over time to use a fixed amount of production capacity; or, up to a certain level of precision, capacity can be allocated at the right times according to a plan.

On the other hand, if demand is unknown, infinite possible profiles exist and the pull system is forced to react to demand changes as they happen. This reaction involves decision-making. According to the original Toyota Production System's guidelines, fixed (i.e., monthly) reviews of historical demand are recommended, during which occasion, if necessary and feasible, production capacity (and the number of kanbans) should be adjusted accordingly (Monden, 1981). This review period is here referred to as ΔT.

An initial analysis of typical demand profiles (e.g. stasis, linear growth, exponential growth, oscillation, etc.) has been conducted. The findings suggest that, among the profiles considered, stasis is an appropriate profiles to be further investigated in detail in this study. The reason is that, given realistic considerations (regarding characteristics such as noise, signal, period of oscillation, damping and amplification), the impact on the pull system's performance is likely to be less intuitive. The oscillation mode would be another interesting profile for a future study.

D.2 Literature Findings

Table D.1 presents a sample of papers showing how demand and processing times have been considered in previous studies. This initial sample includes the papers presented in detail in the literature review section concerning flexible kanbans systems: Rees et al (1987) and Gupta et al (1997); as well as other papers addressing demand instability issues.

Table D.1 *Sample of Demand and Processing Time Profiles Considered in the Literature.*

Authors	Demand	Processing Times
(Rees et al., 1987)	Constant	Stasis with noise
(Chang & Yih, 1994)	Stasis with uniform noise	Stasis with uniform noise
(Yavuz & Satir, 1995)	Stasis with uniform noise	Stasis with normal noise
(Huq & Pinney, 1996)	Stasis with uniform noise	Constant
(Gupta & Al-Turki, 1997)	Stasis with uniform noise	Stasis with normal noise
(Gupta et al., 1999)	Stasis with uniform noise	Stasis with normal noise
(Moore & Gupta, 1999)	Constant	Stasis with exponential noise
(Takahashi & Nakamura, 1999)	Unstable changes in mean; with normal noise	Stasis with gamma noise
(Takahashi & Nakamura, 2000b)	Unstable changes in mean; with normal noise	Stasis with gamma noise

D.3 Assumptions for Evaluation of Typical Profiles

- Demand is exogenous;
- Processing times are exogenous;
- Production only takes place as a reaction to actual demand;
- Capacity (and the number of kanbans) is adjusted on a periodic (i.e. ΔT) basis;
- At the time of review, capacity can be adjusted to match the average level of demand during the last review period;
- The time to adjust capacity is negligible compared to the dynamics of the system;
- Capacity remains fixed in between reviews;
- Demand not fulfilled on time is lost;
- Initially, demand rate and production rate are equal.

D.4 Summary of Analysis

Apparently, the majority of the work developed to date seems to have considered the *stasis* profile of demand. If the presence of noise in demand and processing times is disregarded, this represents the simplest scenario; one in which production rate matches the demand rate over time without any difficulty (see **Figure D.1**). The system is designed and set up once and does not require alteration afterwards. Demand is fulfilled with a delay equivalent to the process lead-time.

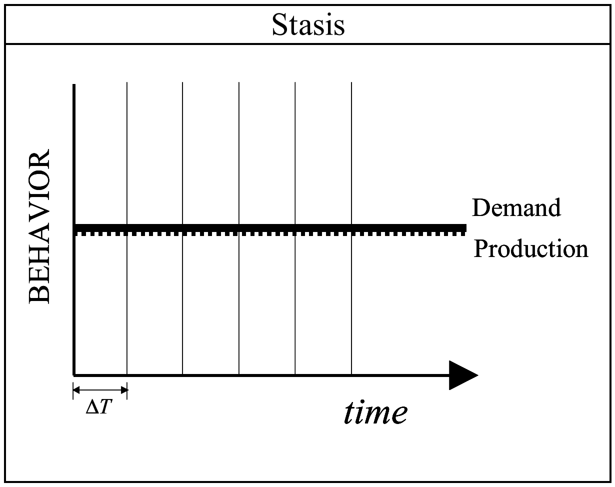


Figure D.1 *Stasis.*

However, even under the *stasis* profile the situation is different once noise is considered. Noise in processing times will cause lack of synchronization of flow and consequent reduction of production capacity – a certain amount of buffer inventory between processing steps is now needed if flow is to take place uninterruptedly. The inclusion of noise in demand will further enhance this problem.

For the purpose of this initial analysis, no noise will be considered. Along these lines, a similar evaluation has been conducted assuming that actual demand assumes a pattern equivalent to each of the typical dynamic profiles. The results suggests the following:

- If demand has a strictly growing pattern (or growing with eventual stabilization), a fraction of demand will always be lost; however, the smaller the value of ΔT the greater the ability of the system to fulfill demand;
- If demand has a strictly decaying pattern (or decaying with eventual stabilization), production will always be able to meet demand independent of ΔT;

- When demand profiles present combined growth and decay tendencies (and orders can be backlogged and fulfilled at a later time if necessary), no immediate conclusion can be drawn; whether or not the system can fulfill demand under these conditions depends on the specific demand pattern, the time allowed to fulfill backlogged orders, as well as ΔT .

Profiles that fit particularly well in this last category include: (1) *stasis* with noise and/or signal(s); and (2) *oscillation* with noise, signal(s), variable period of oscillation, amplification or damping.

An illustration of each particular demand rate profile, altogether with the resulting production rate and lost demand (dashed area) is presented next. These profiles are classified as *growth*, *decay*, or *other modes combining growth and decay* (see **Figures D.2**, **D.3**, and **D.4**)

D.4.1 Growth

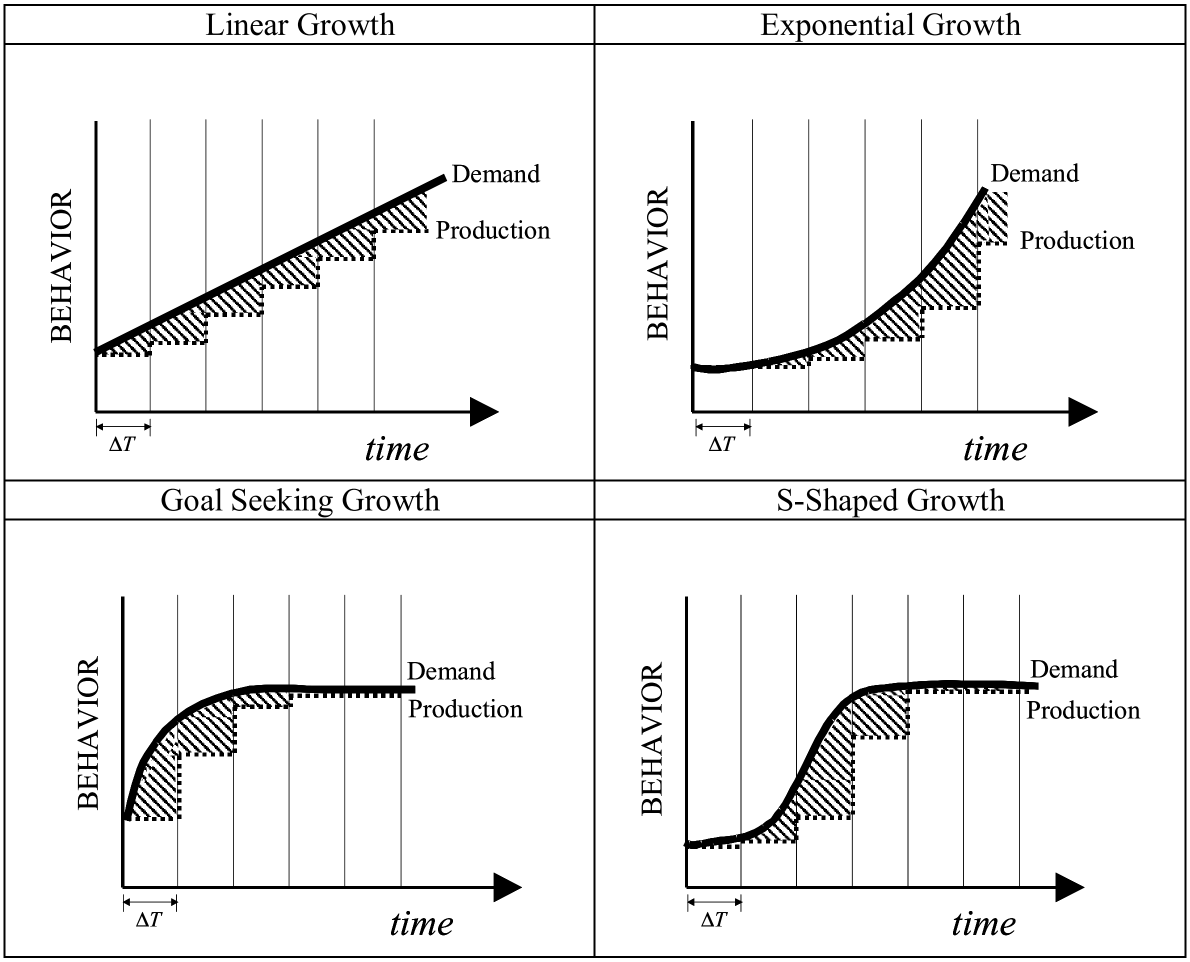


Figure D.2 *Growth.*

D.4.2 Decay

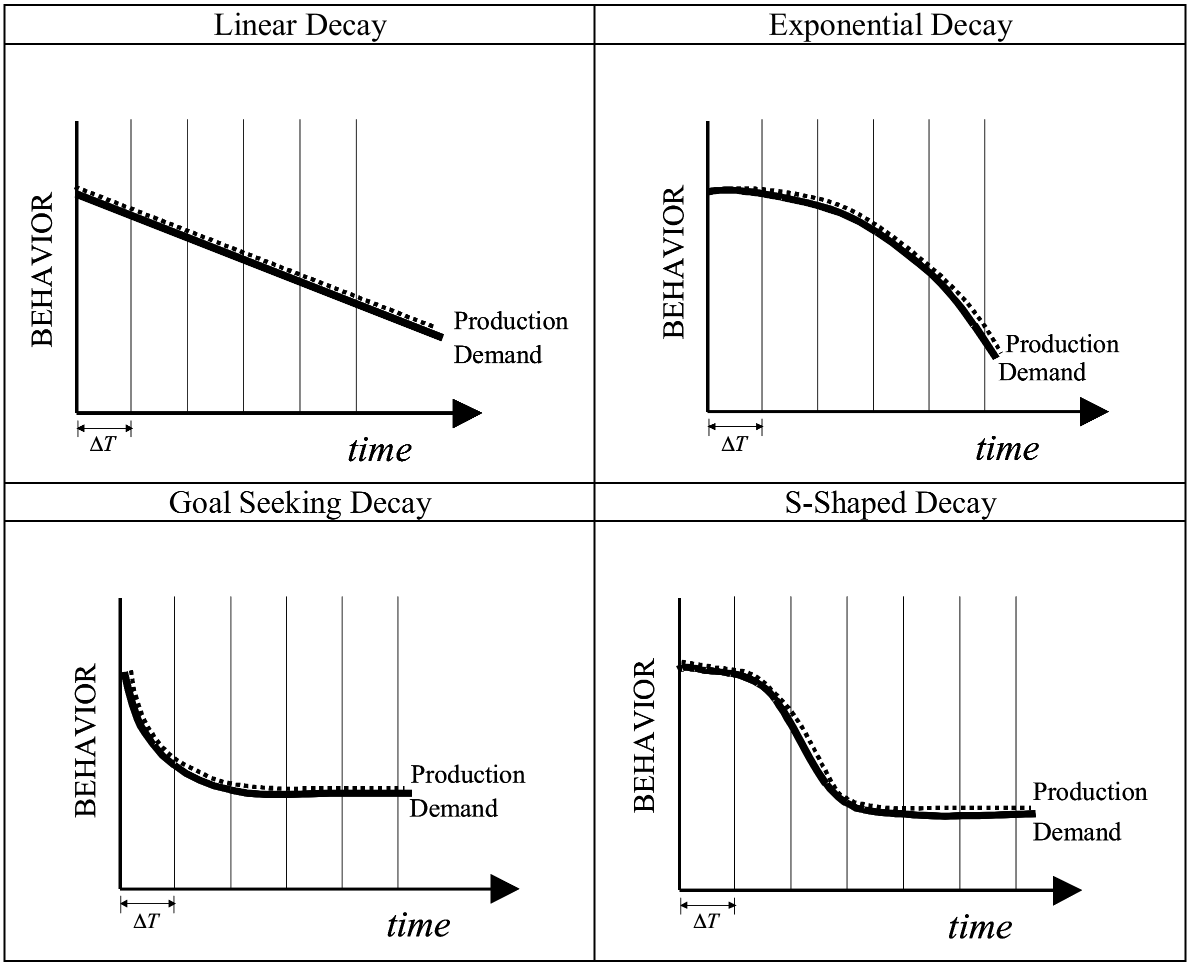


Figure D.3 *Decay.*

D.4.3 Other Modes Combining Growth and Decay

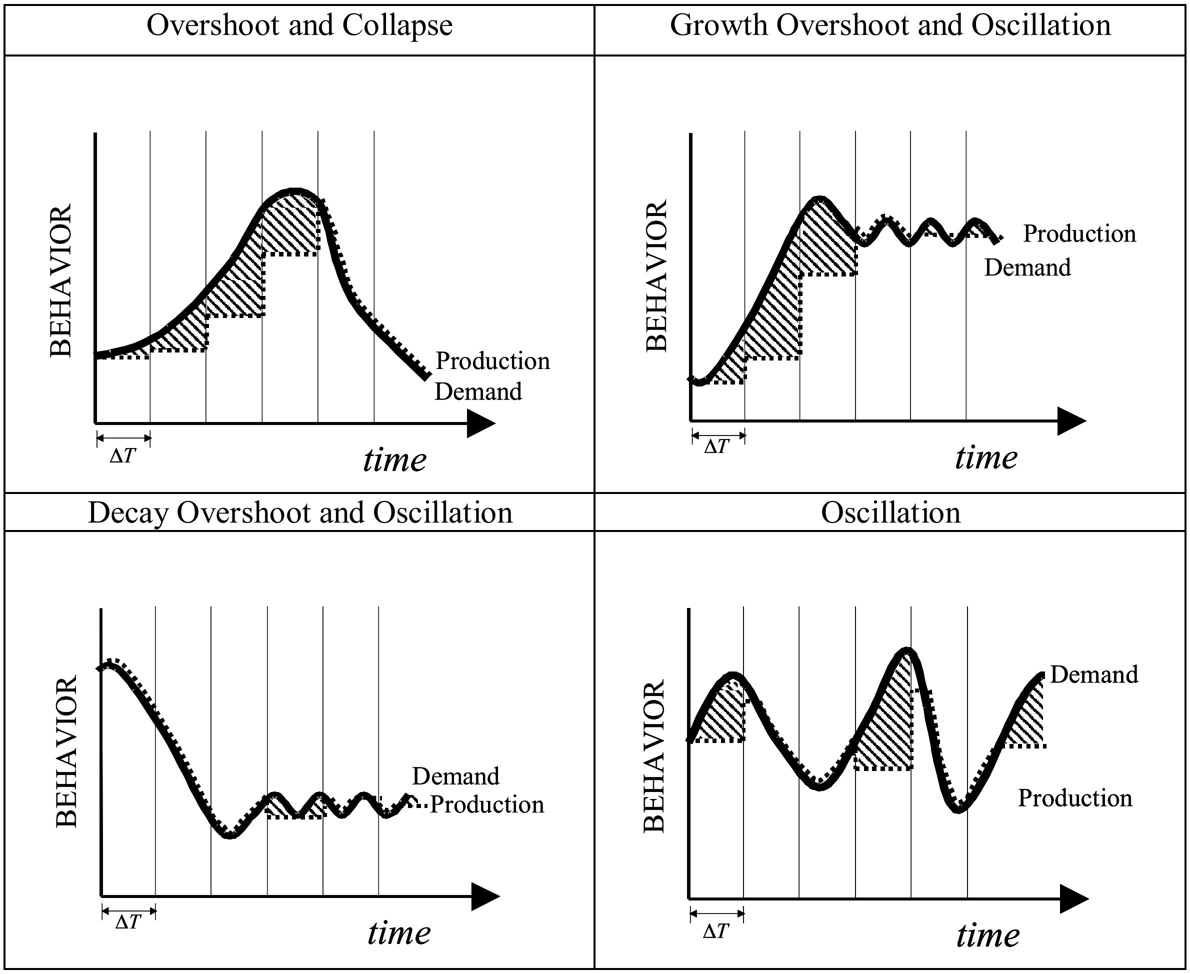


Figure D.4 *Combined Growth and Decay.*

D.5 Conclusions

In alignment with the problem definition, it was understood that this study should focus on the stasis mode of behavior but with special attention not to noise but to signals (i.e., unexpected brief demand occurrences that significantly surpass the expected demand). The stasis mode was chosen because under spike conditions the precise effects upon the value stream are less intuitive than most of the other modes.

Furthermore, considering that a signal is a very brief occurrence, its effects are likely to be perceived only at the end of a planning period. This is so because typically demand is not monitored and converted into production schedules in real time; demand tends to be accumulated over a significant time period (typically a shift, i.e., a “planning period” according to the definitions adopted in this research) and then allocated to the shop floor according to a certain scheduling policy. Such a realization lead to an important conclusion: the precise shape of the signal curve is not relevant for the study of the dynamics of interest in this research; what really matters is the quantity of additional demand incurred over the period of measurement.

Appendix E –
Real Case Model Tests And Evaluations

Details about real model tests are presented in this appendix. This text focuses on the real AS-IS model. These tests were intended to build trust in the model, estimate parameters for which data was unavailable, and create a baseline for the real case.

E.1 AS-IS Simulation Model

In order to capture the real case condition of allocating the entire demand to the floor every planning period, the simulation model was adapted. In this case, the definition of the *average demand quantity for a planning period (ADQPP)* was adapted to always equal the quantity of demand incurred in the previous period. This was done simply by multiplying by 100 fold the value that otherwise would be obtained for *ADQPP* under the hypothetical model conditions.

E.1.1 AS-IS Parameters and Initial Conditions Estimation

A description of the setting of parameters and initial conditions is presented in this section (see **Table E.1** and **E.2**).

Table E.1 *Parameter Estimates.*

Parameter	Estimated [unit]	Description
NSDDBS : number of standard deviations of demand rate used for the	*N/A*	The PDI sizing policy embedded in the hypothetical model was

calculation of buffer stock		substituted the target system's own policy, which does not use this parameter.
EUPS : *expected upstream process shortfall*	*N/A*	Same reason as above.
SDDPEP : *standard deviation of demand rate in previous estimation period*	$SDDPEP = 0.048$ *[piece/min]*	Calculated out of the data gathered using the definition of standard deviation for normally distributed variables.
PUPCF : *protective upstream production capacity factor*	$PUPCF = 0.40$	Calculated according to the definition presented in **Chapter 4**.
PDPCF : *protective downstream production capacity factor*	$PDPCF = 0.21$	Calculated according to the definition presented in **Chapter 4**.
ADRPEP : *average demand rate in previous estimation period*	$ADRPEP = 0.041$ *[piece/min]*	Calculated out of the data gathered using the definition of average.
SPP : *size of planning period*	$SPP = 480$ *[min]*	Same as in the hypothetical case.
DR : *demand rate*	*See historical demand profile [piece/min]*	This variable was inserted into the model using a lookup table, which was set up to reproduce the historical demand rate curve.
KS : *kanban size*	$KS = 6.875$ *[piece/kanban]*	Represents the average kanban size among family variants.
onekanban	$onekanban = 1$ *[kanban]*	A mathematical operator.
NKpitch : *number of kanbans in a pitch*	$NKpitch = 1$ *[kanban]*	Defined according to the assumptions presented in **Chapter 4**.
t : *time*	*t* : *computer clock time [min]*	An independent variable generated by the computer clock.
dt : *time step*	$dt = 0.03125$ *[min]*	Defined by testing as a value that does not significantly interfere in the dynamics of the model.
UPPD : *upstream process percent downtime*	$UPPD = 0.2$	No data available to estimate this parameter.

		Defined according to discussion with the target value stream manager and accepted by means of testing.
UPCN : upstream production capacity noise	*UPCN* = 0 *[piece/min]*	No noise is considered in the "what if…" analysis.
DPPD downstream process percent downtime	*DPPD* = 0.2	No data available to estimate this parameter. Defined according to discussion with the target value stream manager and accepted by means of testing.
DPCN : downstream production capacity noise	*DPCN* = 0 *[piece/min]*	No noise is considered in the "what if…" analysis.
INITIAL TIME: initial simulation time	*INITIAL TIME = 0 [min]*	The simulation starts at time zero.
FINAL TIME: final simulation time	*FINAL TIME = 29280 [min]*	The length of the selected historical period in minutes. Each day is equivalent to a planning period.
SAVEPER: save results every SAVEPER time period	*SAVEPER = 120 [min]*	This parameter defined the number of data points collected from the simulation. Results in this case are saved every two hours. Actual data is available in the format of one data point per day.
IPDIRC : initial PDI level in real case	*IPDIRC* = 55 *[pieces]*	The total PDI size obtained from the summation of all its kanban cards. No data is available to determiner the precise level of this supermarket. However, it is considered to have entered the target historical period full because the system at that time had significant time with zero demand and backlog.

Table E.2 Initial Conditions.

Stock	Initial Value	Description
$PDI(t)$: Protective decoupling inventory	$PDI(t_0) = IPDIRC$	Same as above.
$DFCPP(t)$: Demand to be fulfilled in current PP	$DFCPP(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period.
$DICPP(t)$: Demand incurred in current PP	$DICPP(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period.
$DO(t)$: Demand overflow	$DO(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period.
$DBWO(t)$: Demand being worked on	$DBWO(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period.
$IUDP(t)$: Items in use at downstream process	$IUDP(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period.
$IUUP(t)$: Items in use at upstream process	$IUUP(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period.
$HB(t)$: Heijunka box	$HB(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period. All kanbans were attached to parts at the PDI.
$APDIK(t)$: Assigned PDI kanbans	$APDIK(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period.
$UCUP(t)$: Units completed by upstream process	$UCUP(t_0) = pitch$	It is assumed that the system had completed a number of units equivalent to a pitch, otherwise the pull logic used would block the system. The error inserted in the results is insignificant.
$QOC(t)$: Quantity on order from external customer	$QOC(t_0) = 0$	There was zero demand and zero backlog at the beginning of the simulation period.
$UCDP(t)$: Units completed by downstream process	$UCDP(t_0) = pitch$	It is assumed that the system had completed a number of units equivalent to a pitch, otherwise the pull logic used would block the system. The error inserted in the results is insignificant.

$NKC(t)$: Total number of kanbans in circulation	$NKC(t_0) = INK$	Defined as a function of INK, i.e., by the initial PDI level divided by the kanban size.
$NKR(t)$: Number of kanbans to be removed	$NKR(t_0) = 0$	Zero. This stock is only used in TO-BE scenarios involving the consideration of Policy 1. Therefore, it remains empty in the AS-IS model.

It should be noted that Policy 2 utilizes parameters used to describe a normally distributed function (i.e., average and standard deviation) in order to provide a simple guiding rule for decision-making. However, the actual demand profile might not necessarily be best described by a normal distribution function. The distribution of demand values for the entire historical period available in the real case application is given in the histogram in **Figure E.1**. The figure suggests that the exponential probability distribution might actually provide a better description of demand behavior. Future development of Policy 2 could be focused on incorporating parameters from other distribution functions, such as the exponential distribution in this case, and testing its effect on value stream performance over time.

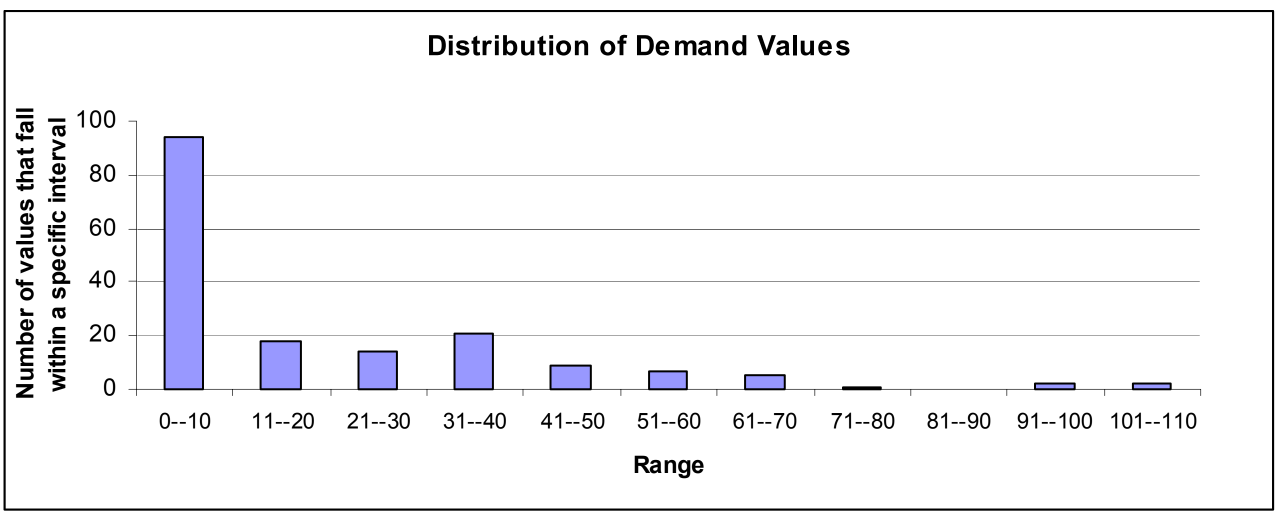


Figure E.1 *Distribution of Demand Values.*

E.1.2 Real AS-IS Model Testing

The results of some key model tests as suggested by Sterman (2000) are presented in this section. The Boundary Adequacy, Structure Assessment, Extreme Conditions, Integration Error, Family Member tests developed in the hypothetical case remain valid here.

E.1.3 Dimensional Consistency

Is each equation dimensionally consistent without the use of parameters having no real world meaning?

Yes. Hypothetical case comments are also valid here.

E.1.4 Parameter Assessment

Are the parameter values consistent with relevant descriptive and numerical knowledge of the system?

Yes. Most of the parameters were estimated from real data. Some of them where estimated using model testing to confirm a guess.

Do all parameters have real world counterparts?

Yes. All exogenous variables have real world correspondents.

E.1.5 Behavior Reproduction

Various cycles of behavior reproduction testing were conducted (please refer to **Figure E.1)**, These iterations served to define parameter setting and also eventually led to identify an error in the model which violated the value stream assumptions presented in **Chapter 4**. This error was then fixed in real case as well as in the hypothetical model. The curves shown in the figure are built using one data point per day or planning period, the same format as the available data.

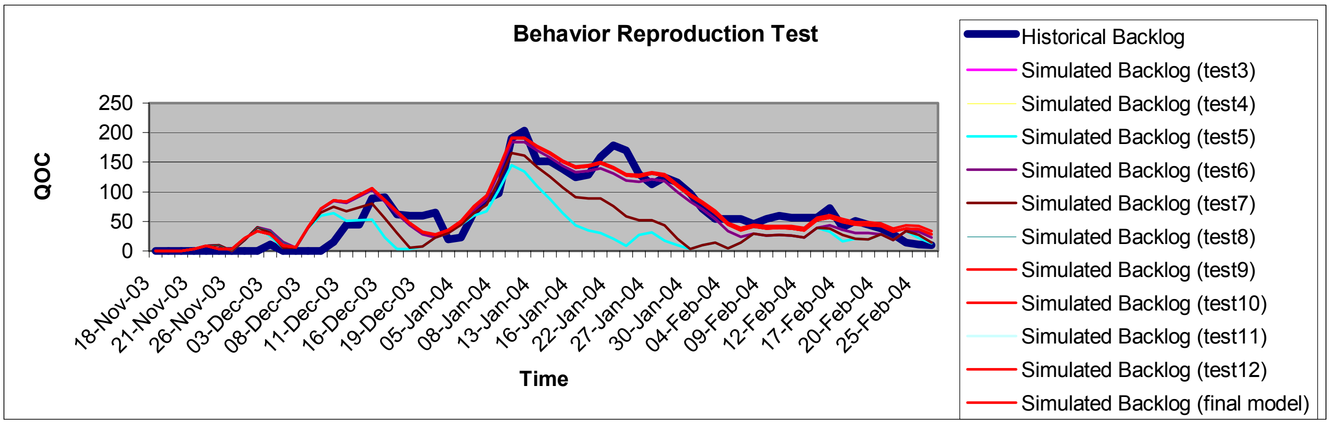


Figure E.2 *Behavior Reproduction Test Results.*

References

ARC (2003). Servo Drives Worldwide Outlook, ARC Advisory Group. **2004**.

Bernus, P. and L. Nemes, Eds. (1996). Modelling and Methodologies for Enterprise Integration. London, UK, Chapman & Hall.

Biemans, F. P. M., M. M. Lankhorst, et al. (2001). "Dealing with the Complexity of Business Systems Architecting." Systems Engineering **4**(2): 118-133.

Bititci, U. S. and D. Muir (1997). "Business process definition: a bottom-up approach." International Journal of Operations & Production Management **17**(4): 365-374.

Blackstone, J. H. and J. F. Cox (2002). "Designing unbalanced lines - understanding protective capacity and protective inventory." Production Planning and Control **13**(4): 416-423.

Blanchard, B. S. (1991). Systems Engineering Management, John Wiley & Sons, Inc.

Blanchard, B. S. and W. J. Fabrycky (1998). Systems Engineering and Analysis. Upper Saddle River, New Jersey, Prentice Hall.

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

Bruno, G. and R. Agarwal (1997). "Modeling the Enterprise Engineering Environment." IEEE Transactions on Engineering Management **44**(1): 20-30.

Carley (1999). "On Generating Hypothesis Using Computer Simulation." Systems Engineering **2**: 69-77.

Carlock, P. G. and R. E. Fenton (2001). "System of Systems (SoS) Enterprise Systems Engineering for Information-Intensive Organizations." Systems Engineering **4**(4).

Chang, T. M. and Y. Yih (1994). "Generic kanban systems for dynamic environments." International Journal of Production Research **32**(4): 889-902.

Chapman, W. L., J. Rozenblit, et al. (2001). "System Design is an NP-Complete Problem." *Systems Engineering* **4**(3): 222-229.

Cooney, R. (2002). "Is "lean" a universal production system? Batch production in the automotive industry." International Journal of Operations & Production Management **22**(10): 1130-1147.

Corrêa, H. L. and I. G. N. Gianesi (1996). Just in Time, MRP II e OPT: um enfoque estratégico. São Paulo, SP, Editora Atlas S.A.

Disney, S. M., M. M. Naim, et al. (1997). "Dynamic simulation modelling for lean logistics." International Journal of Physical Distribution & Logistics **27**(3/4): 174-196.

Dong, M. and F. F. Chen (2001a). Modeling and Performance Analysis of Manufacturing Logistic Chains: An Integrated Framework. 5th International Conference on Engineering Design and Automation, Las Vegas, Nevada USA.

Dong, M. and F. F. Chen (2001b). "Process modeling and analysis of manufacturing supply chain networks using object-oriented Petri nets." Robotics and Computer Integrated Manufacturing **17**: 121-129.

Fine, C. H. (1998). Clockspeed - Winning Industry Control in the Age of Temporary Advantage. Reading, Massachusetts, Perseus Books.
Forrester, J. W. (1961). Industrial Dynamics. Cambridge, MA, The MIT Press.
Forrester, J. W. (1975). Collected Papers of Jay W. Forrester. Cambridge, Massachusetts, Wright-Allen Press, Inc.
Forrester, J. W. (1996). The Beginning of System Dynamics, Massachusetts Institute of Technology.
Forrester, J. W. (1998). Designing the Future. Universidade de Sevilla, Sevilla, Spain.
Fowler, A. (1998). "Operations Management and Systemic Modeling as Frameworks for BPR." *International Journal of Operations & Production Management* **Vol. 18**(No. 9/10): 028-1056.
Fowler, A. (1999). "Feedback and feedforward as systemic frameworks for operations control." International Journal of Operations & Production Management **19**(2): 182-204.
Fung, K. K. (1999). "*Follow the laggard? - Not all bottlenecks are created equal*." System Dynamics Review **15**(4): 403-410.
Gianesi, I. G. N. (1998). "Implementing Manufacturing Strategy through Strategic Production Planning." International Journal of Operations and Production Management **18**(3): 286-299.
GlobalSpec (2004). About Servo Drives. **2004**.
Gonçalves, P. (2003). The Impact of Shortages on Push-Pull Production Systems. Sloan School of Management - MIT, Working Paper.
Goranson, H. T. (1997). "Design for Agility using Process Complexity Measures." Agility & Global Competition **1**(3): 47-55.
Goranson, H. T. (1999). The Agile Virtual Enterprise: Cases, metrics, tools. Westport, CT, Quorum Books.
Groesbeck, R. L. (2001). An Empirical Study of Group Stewardship and Learning:
Implications for Work Group Effectiveness. Industrial and Systems Engineering. Blacksburg, VA, Virginia Polytechnic Institute & State University**:** 242.
Grosfeld-Nir, A., M. Magazine, et al. (2000). "Push and pull strategies for controlling multistage production systems." International Journal of Production Research **38**(11): 2361-2375.
Gupta, S. M. and Y. A. Y. Al-Turki (1997). "An algorithm to dynamically adjust the number of Kanbans in stochastic processing times and variable demand environment." Production Planning and Control **8**(2): 133-141.
Gupta, S. M. and Y. A. Y. Al-Turki (1998a). "Adapting just-in-time manufacturing systems to preventive maintenance interruptions." Production Planning and Control **9**(4): 349-359.
Gupta, S. M. and Y. A. Y. Al-Turki (1998b). "The effect of sudden material handling system breakdown on the performance of a JIT system." International Journal of Production Research **36**(7): 1935-1960.
Gupta, S. M., Y. A. Y. Al-Turki, et al. (1999). "Flexible kanban system." International Journal of Operations & Production Management **19**(10): 1065-1093.
Gupta, Y. P. and M. Gupta (1989). "A system dynamics model of a JIT-kanban system." Engineering Costs and Production Economics **18**: 117-130.
Hoek, R. I. v. (2000). "The thesis of leagility revisited." International Journal of Agile Management Systems **2**(3): 196-201.
Hornby, A. S. (1995). Oxford Advanced Learner's Dictionary of Current English. Oxford, UK, Oxford University Press.

Huq, F. and W. E. Pinney (1996). "Impact of short-term variations in demand on opportunity costs in a just-in-time kananban system." Production and Inventory Management Journal **Fourth Quarter**.
INCOSE (2002). Intelligent Enterprises Working Group. **2002**.
Kamath, M., N. P. Dalal, et al. (2001). Process-Modeling Techniques for Enterprise Analysis and Design: A Comparative Evaluation. Institute of Industrial Engineering Annual Conference, Dallas, TX USA.
Keating, E. K., R. Oliva, et al. (1999). "Overcoming the Improvement Paradox." European Management Journal **17**(2): 120-134.
Keough, M. and A. Doman (1992). The CEO as organization designer: an interview with Prof. Jay W. Forrester. The McKinsey Quaterly**:** 03-30.
Kosanke, K., F. Vernadat, et al. (1999). "CIMOSA: enterprise engineering and integration." Computers in Industry **40**: 83-97.
Kotter, J. P. (1996). *Leading Change*, Harvard Business School Press.
Kumar, R. and P. Vrat (1989). "Using Computer Models in Corporate Planning." Long Range Planning **22**(2): 114-120.
Kurstedt, H. A. (2000). Management Systems Theory, Applications, and Design. Blacksburg, VA, Author.
Law, A. M. and W. D. Kelton (2000). Simulation Modeling and Analysis, McGraw-Hill.
Leedy, P. D. and J. E. Ormrod (2001). Practical Research: Planning and Design. Upper Saddle River, New Jersey, Prentice-Hall, Inc.
LEI (2003). Technical Workshops and Value Stream Management Seminars. **2003**.
Liles, D. H., M. E. Johnson, et al. (2002). The Enterprise Engineering Discipline, Automation and Robotics Research Institute - University of Texas at Arlington. **2002**.
Liles, D. H. and A. R. Presley (1996). Enterprise Modeling Within an Enterprise Engineering Framework. Winter Simulation Conference, San Diego, CA USA.
Little, D., M. Peck, et al. (2001). "Responsive manufacturing demands alignment of production control methods to business drivers." Integrated Manufacturing Systems **12**(3): 170-178.
Mason-Jones, R., B. Naylor, et al. (2000). "Lean, agile or leagile? Matching your supply chain to the marketplace." International Journal of Production Research **38**(17): 4061-4070.
Mason-Jones, R. and D. R. Towill (1998). "Time compression in the supply chain: information management is the vital ingredient." Logistics Information Management **11**(2): 93-104.
Matson, J. B. and D. C. McFarlane (1999). "Assessing the responsiveness of existing production operations." International Journal of Operations and Production Management **19**(8): 765.
Moeeni, P., S. M. Sanchez, et al. (1997). "A robust design methodology for Kanban system design." International Journal of Production Research **35**(10): 2821-2838.
Monden, Y. (1981). "Adaptable Kanban System Helps Toyota Maintain Just-In-Time Production." Industrial Engineering **May**: 29-46.
Moore, K. E. and S. M. Gupta (1999). "Stochastic coloured Petri net (SCPN) models of traditional and flexible kanban systems." International Journal of Production Research **37**(9): 2135-2158.
Naylor, J. B., M. M. Naim, et al. (1999). "Leagility: Integrating the lean and agile manufacturing paradigms in the total supply chain." International Journal of Production Economics **62**: 107-118.
Nightingale, D. J. (2002). Lean Enterprises - A Systems Perspective, Massachusetts Institute of Technology**:** 18.

O'Callaghan, R. (1986). A System Dynamics Perspective on JIT-Kanban. The 1986 International Conference of the System Dynamics Society, Sevilla, Spain.
Oliva, R. (2001). "Tradeoffs in Response to Work Pressure in the Service Industry." California Management Review **43**(4): 26-43.
Petrie Jr., C. J., Ed. (1992). Enterprise Modeling and Integration: Proceedings of the First International Conference. Cambridge, MA, The MIT Press.
Powell, S. G., M. Schwaninger, et al. (2001). "Measurement and control of business processes." System Dynamics Review **17**(1): 63-91.
Presley, A. R. (1997). A Multi-View Enterprise Modeling Scheme. 6th Industrial Engineering Research Conference, Miami, FL USA.
Pressman, R. S. (1995). Engenharia de Software. São Paulo, Makron Books.
Razmi, J., H. Rahnejat, et al. (1998). "Use of analytic hierarchy process approach in classification of push, pull and hybrid push-pull systems for production planning." International Journal of Operations & Production Management **18**(11): 1134-1151.
Rees, L. P., P. R. Philipoom, et al. (1987). "Dynamically Adjusting the Number of Kanbans in a Just-in-Time Production System Using Estimated Values of Leadtime." IIE Transactions **19**(2): 199-207.
Rentes, A. F. (2000). Transmeth - Proposta de uma Metodologia para Condução de Processos de Transformação de Empresas. Industrial Engineering. São Carlos, SP, University of São Paulo**:** 229.
Rentes, A. F. (2002). Future State Design. Blacksburg, VA, Author.
Rentes, A. F., E. M. Van Aken, et al. (1999). An organizational assessment method for transformation efforts. *Portland International Conference on Management of Engineering and Technology*, Portland, OR.
Repenning, N. P., P. Gonçalves, et al. (2001). "Past the Tipping Point: The Persistence of Firefighting in Product Development." California Management Review **43**(4).
Repenning, N. P. and J. D. Sterman (2001). "Nobody Ever Gets Credit for Fixing Problems that Never Happened: Creating and Sustaining Process Improvement." California Management Review **43**(4): 64-88.
Resende, M. O. and J. B. Sacomano (*1997*). *Princípios dos Sistemas de Planejamento e Controle da Produção*. São Carlos, SP, Universidade de São Paulo.
Ritchie-Dunham, J. and E. G. Anderson (2000). A Strategic Supply Chain Simulation Model. Winter Simulation Conference.
Ritchie-Dunham, J. L. and H. T. Rabbino (2001). Managing from Clarity: Identifying, Aligning, and Leveraging Strategic Resources, John Wiley & Sons, Ltd.
Rolstadas, A. (1998). "Enterprise performance measurement." International Journal of Operations & Production Management **18**(9/10): 989-999.
Rother, M. and J. Shook (1999). Learning to See: value stream mapping to create value and eliminate muda. Brookline, MA USA, Lean Enterprise Institute.
Scheer, A.-W. (1991). Principles of Efficient Information Management. Berlin, Germany, Springer-Verlag.
Scheer, A.-W. (1992). Architecture for Integrated Information Systems, Springer-Verlag.
Scheer, A.-W. (1993). CIM: Evoluindo para a fábrica do futuro. Rio de Janeiro, RJ Brazil, Qualitymark Editora Ltda.
Scheer, A.-W. (1994). Business Process Engineering: Reference models for industrial enterprises. Berlin, Germany, Springer-Verlag.

Scheer, A.-W. (1999a). ARIS - Business Process Frameworks, Springer.
Scheer, A.-W. (1999b). ARIS - Business Process Modeling. Berlin, Germany, Springer-Verlag.
Senge, P. M., A. Kleiner, et al. (1994). The Fith Discipline Fieldbook. New York, NY, Doubleday Dell Publishing Group, Inc.
Shewchuk, J. P. (2003). Advanced Topics in Production Planning and Control, Author.
Sipper, D. and R. L. Bulfin Jr. (1997). Production Planning Control and Integration. New York, NY USA, McGraw-Hill.
Smalley, A. (2004). Creating Level Pull. Brookline, MA, The Lean Enterprise Institute.
Sousa, G. W. L. (1999). Aplicação de Conceitos de Modelagem e Integração de Empresas no Gerenciamento de Projetos de Transformação Organizacional. Industrial Engineering. São Carlos, SP, University of São Paulo**:** 134.
Sousa, G. W. L., L. C. R. Carpinetti, et al. (2003). Enterprise Analysis and Conceptual Redesign Using System Dynamics Modeling and Simulation. International Conference on Industrial Engineering and Operations Management, Ouro Preto, MG Brazil.
Sousa, G. W. L., E. M. Van Aken, et al. (2001). Using Enterprise Modeling to Facilitate Knowledge Management in Organizational Transformation Efforts. Portland International Conference on the Management of Engineering and Technology, Portland, OR.
Spearman, M. L., D. Woodruff, et al. (1990). "CONWIP: A Pull Alternative to Kanban." International Journal of Production Research **28**: 879-894.
Stephens, S. (1999). *Supply Chain Council & SCOR Model Overview*. **1999**.
Sterman, J. D. (1991). A Skeptic's Guide to Computer Models. Managing a Nation: The Microcomputer Software Catalog. G. O. B. e. al. Boulder, CO USA, Westview Press**:** 209-229.
Sterman, J. D. (2000). Business Dynamics - Systems Thinking and Modeling for a Complex World, McGraw-Hill.
Sterman, J. D. (2001). "System Dynamics Modeling: Tools for learning in a complex world." California Management Review **43**(4): 8-25.
Stratton, R. and R. D. H. Warburton (2003). "The strategic integration of agile and lean supply." International Journal of Production Economics **85**: 183-198.
Takahashi, K. and N. Nakamura (1999). "Reacting JIT ordering systems to the unstable changes in demand." International Journal of Production Research **37**(10): 2293-2313.
Takahashi, K. and N. Nakamura (2000a). "Agile control in JIT ordering systems." International Journal of Agile Management Systems **2**(3): 242-252.
Takahashi, K. and N. Nakamura (2000b). "Reactive logistics in a JIT environment." Production Planning and Control **11**(1): 20-31.
Takahashi, K. and N. Nakamura (2002a). "Comparing Reactive Kanban and Reactive CONWIP." Production Planning & Control **13**(8): 702-714.
Takahashi, K. and N. Nakamura (2002b). "Decentralized reactive Kanban system." European Journal of Operational Research **139**: 262-276.
Tardif, V. and L. Maaseidvaag (2001). "An adaptive approach to controlling kanban systems." European Journal of Operational Research **132**: 411-424.
Taylor, L. J. (1999). "A simulation study of WIP inventory drive systems and their effect on financial measurements." Integrated Manufacturing Systems **10**(5): 306-315.
Towill, D. R. (1996a). "Industrial dynamics modeling of supply chains." Logistics Information Management **9**(4): 43-56.

Towill, D. R. (1996b). "Time compression and supply chain - a guided tour." Logistics Information Management **9**(6): 41-53.
Vensim (2002). User's Guide. United States of America, Ventana Systems, Inc.
Ventana (2003). Frequently asked questions about Vensim., Ventana. **2003**.
Vernadat, F. B. (1996). Enterprise Modeling and Integration: Principles and Applications. London, UK, Chapman & Hall.
Vollmann, T. E., W. L. Berry, et al. (1997). Manufacturing Planning and Control Systems, Irwin McGraw-Hill.
Walizer, M. H. and P. L. Wienir (1978). Research Methods and Analysis: Searhing for relationships. New York, NY, Harper & Row, Publishers.
Wang, D. and C.-G. Xu (1997). "Hybrid push/pull production control strategy simulation and its applications." Production Planning and Control **8**(2): 142-151.
White Jr., K. P. (1998). "Systems Design Engineering." *ystems Engineering* ***1***: 285-302.
Whitman, L. E., B. L. Huff, et al. (1998). The needs and issues associated with representing and integrating multiple views of the enterprise. DIISM.
Wiendahl, H.-P. and J.-W. Breithaupt (1999). "Modeling and controlling the dynamics of production systems." Production Planning and Control **10**(4): 389-401.
Wilding, R. (1998). "The Supply Chain Complexity Triangle - uncertainty generation in the supply chain." International Journal of Physical Distribution & Logistics **28**(8): 599-616.
Womak, J. P. and D. T. Jones (1996). Lean Thinking: banish waste and create wealth in your corporation. New York, NY USA, Simon & Schuster.
Wortmann, J. C., D. R. Muntslag, et al., Eds. (1997). Customer-driven Manufacturing, Chapman & Hall.
Yavuz, I. H. and A. Satir (1995). "A kanban-based simulation study of a mixed model just-in-time manufacturing line." International Journal of Production Research **33**(4): 1027-1048.
Yu, B., J. A. Harding, et al. (2000). "Supporting enterprise design through multiple views." International Journal of Agile Management Systems **2**(1): 71-82.

Vita

George W. L. Sousa was born on May 11 1975 in São Paulo, Brazil. He received BS and MS degrees in production engineering from EESC USP – São Carlos Engineering School, University of São Paulo in 1997 and 1999 respectively. His research interests focus on the joint application of system dynamics and systems engineering methods, in particular for the analysis, (re)design, and implementation of enterprise management processes. Prior to graduate studies he worked as a business process engineer at IDS Prof. Scheer South America. As a Ph.D. student he has worked for the past four and a half years as a graduate research assistant at the Enterprise Engineering Research Lab at Virginia Tech. During the course of graduate studies he engaged in various industrial projects, including work with Fultec Ltda. and TRW in Brazil, and ElectroTec, AT&T, Vinten, and Danaher Motion in the USA. He is co-founder and former president of the Enterprise Systems Engineering Society of Virginia Tech, an affiliate of the International Council on Systems Engineering. George can be reached at **gsousa@vt.edu**.