LAI Paper Series
"Lean Product Development for Practitioners"

Waste in Lean Product Development

Version 1.1 – July 2010

Prepared by:

Dr. Josef Oehmen
Dr. Eric Rebentisch

1 About LAI

The Lean Advancement Initiative (LAI) at MIT, together with its Educational Network (EdNet), offers organizational members from industry, government, and academia the newest and best thinking, products, and tools related to lean enterprise architecting and transformation. LAI is a unique research consortium that provides a neutral forum for sharing research findings, lessons learned, and best practices.

LAI offers:

- unique opportunities to engage with customers, suppliers, and partners to solve problems and share organizational transformation experiences
- a portfolio of thought-provoking knowledge exchange events and meetings
- innovative enterprise transformation products, tools, and methodologies

LAI researches, develops, and promulgates practices, tools, and knowledge that enable and accelerate enterprise transformation. LAI accelerates lean deployment through identified best practices, shared communication, common goals, and strategic and implementation tools honed from collaborative experience. LAI also promotes cooperation at all levels and facets of an enterprise to eliminate traditional barriers to improving industry and government teamwork.

The greatest benefits of lean result when the operating, technical, business, and administrative units of an enterprise strive for enterprise-wide lean performance. LAI is completing its fifth Enterprise Value phase, during which LAI has engaged in transforming aerospace entities into total lean enterprises and delivered more value to all stakeholders than would have been possible through conventional approaches.

Contact Information

Lean Advancement Initiative
Massachusetts Institute of Technology
77 Massachusetts Avenue
Building 41-205
Cambridge, MA 02139

Homepage: http://lean.mit.edu
Phone: +1 (617) 258-7628
Email: lean@mit.edu

2 About this Series

A vast amount of research has been conducted at MIT´s Lean Advancement Initiative (LAI) on Lean Product Development in the last 15 years. For the first time, this series of papers makes this research accessible to practitioners in a condensed form.

The aim is to provide an application-oriented, readable, concise and comprehensive overview of the main fields of Lean Product Development. The papers follow LAI´s understanding and philosophy regarding Lean Management concepts and especially their integration into large and complex Enterprise settings.

Level of Analysis	Type of Process		
	I. Processes for Value-Orientation	II. Processes for Enterprise Integration	III. Processes for Efficient Execution
Project Portfolio ↕ Single Project	1. Stakeholder needs generation 2. Trade space exploration & decision making 3. Value & waste in core PD processes	4. Enterprise management 5. Program management 6. Multi-project management 7. Performance metrics and measurement 8. Product architecture & commonality management 9. Risk management 10. IT systems in PD 11. HR development & intellectual capital 12. Teams in PD	13. Enterprise process improvement 14. Enabling factors in Lean PD 15. Core PD process principles

Figure 1: Topics of the Paper Series - LAI's Three Main Areas of Lean Product Development

The papers draw mainly on the research done by LAI. Where necessary to ensure a comprehensive presentation of a topic, findings of other researchers and research groups from the field of Lean Product Development are integrated into the papers.

The series focuses on 15 topics in three major areas of Lean Product Development that LAI identified (see Figure 1). The processes span the space from single project to project portfolio management. This paper addresses topic 3, Value and Waste in Core PD Processes.

2.1 I: Processes for Value-orientation

The processes for value-orientation address those types of processes that ensure a focus on the creation of value and the elimination of waste in Lean Product Development. This covers the areas of stakeholder needs generation, trade space exploration and decision making, as well as the identification and handling of value and waste in the core PD processes.

2.2 II: Processes for Enterprise Integration

Enterprise Integration is one of the main challenges in developing a Lean Enterprise. Product Development plays a central role in this integration effort, as it interfaces with all main Enterprise processes. This larger group therefore consists of the processes of enterprise, program and multi-project management, performance metrics and measurement, product architecture and commonality management, risk management, IT systems, HR development and human capital, and teams in Product Development.

2.3 III: Processes for Efficient Execution

This group addresses the challenges surrounding the efficient execution of PD processes. It includes the relationship of PD to overall Enterprise process improvement initiatives and enabling organizational factors within Lean PD, as well as addressing alternative Lean PD core process principles.

3 Introduction to Value and Waste in Lean PD

3.1 Objectives and Intended Readership of this Paper

The main objective of this paper is to make the work that has been done at LAI in the area of waste in product development easily accessible to the consortium members. The focus of the discussion in this paper is therefore on past LAI work. Non-LAI work is integrated into the presentation where it is necessary to complete the picture.

The intended readership is engineers and managers in the areas of product development, product design, systems engineering and program management. The paper is also intended to provide a first overview to students and others interested in the field.

Reading this whitepaper provides a concise overview of the most important waste drivers in product development, that is, the most common project deficiencies that lead to cost and schedule overrun, as well as to performance issues. It will enable those involved in process improvement initiatives to include specific lean-related factors into their process analysis. It provides both managers and engineers with a common language and concepts to enhance the efficiency of their product development projects.

3.2 The Concept of Value and Waste in Lean Management

Lean Management sets itself apart from other management approaches by its clear focus on value delivery to the stakeholders. This is reflected in the five Lean Principles (Womack and Jones, 1998) that all aim at optimizing the delivery of value:

1. Define value to the program stakeholders
2. Plan the value-adding stream of work activities from raw materials until the product delivery while eliminating waste
3. Organize the value stream as an uninterrupted flow of work pulsed by the rhythm of takt time, and proceeding without rework or backflow
4. Organize the pull of the work-in-progress as needed and when needed by all receiving workstations
5. Pursue perfection, i.e. the process of never ending improvement

The delivery of value is the reason for support from customers and other stakeholders, and ultimately a measure of how much they will be willing to pay. Waste, on the other hand, is "all elements […] that only increase cost without adding value" (Ohno, 1998). Lean Product Development focuses on defining and creating successful and profitable product value streams.

Activities in product development can be categorized as either value creating (e.g. interacting with a customer to elicit his or her requirements), necessary but non-value creating (e.g. performing a necessary handover), or waste (e.g. over-engineering a component). In addition, core project activities are often idle, as engineers are waiting for necessary input. Based on LAI's research and experience, the central value-creating engineering tasks are idle for most of

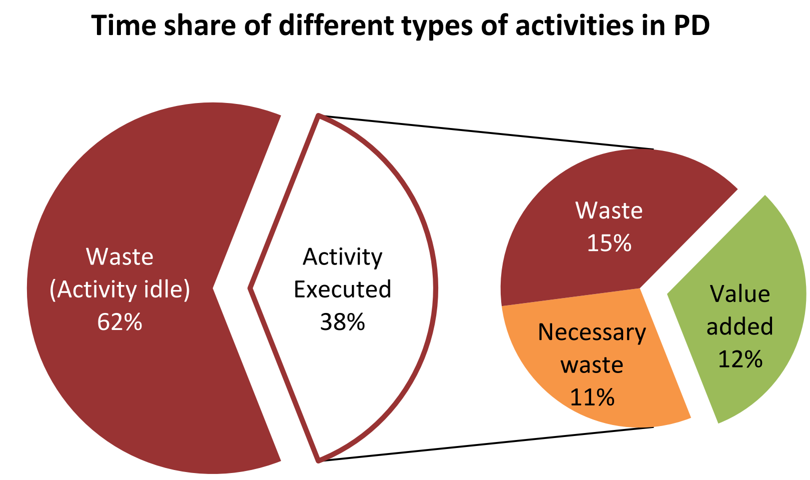


Figure 2: Only about 10% of a project's time is spent creating value

the time (62%). If the tasks are active, engineers spend 40% of their time on pure waste, 29% necessary but non value adding activities, and only 31% on value adding activities (McManus, 2005, Oppenheim, 2004). Combining these two findings (see Figure 2), on average only 12% of the time during the execution of a project are spend on value adding activities, 11% on necessary, but ultimately non value adding activities, and 77% of the time is wasted (the project either being idle, or the engineers working on waste).

If the time that project activities are being idle can be cut in half, even without eliminating any other waste in the project, the time needed to execute the project would almost be cut in half. Eliminating half of the waste now would again add the equivalent of the original time spend on value adding activities.

But discerning waste and value adding activities in PD is not trivial. A meeting with colleagues would be considered value-adding, if important information was exchanged. The same meeting however can be wasted time if a critical piece of information is still missing and the meeting is cancelled and postponed after half an hour. Therefore, the figures provided here can only convey a rough estimate. For example (Evers et al., 1998) report the time spend by engineers during the design phase as follows: value adding activities 33%; necessary waste 38%; and waste 29%.

In this paper, we introduce different categories of waste and explain their relevance in the context of Lean PD. The aim is to provide engineers and managers with the necessary knowledge to identify and eliminate waste during process improvements.

3.3 Scope and limitations of this document

Both "value" and "waste" are very broad topics. To clarify the intent and applicability of this document, the following limitations are important:

- The scope of analysis is the single development project, or one value stream. In reality, organizations often have to manage multiple projects and value streams at the same time, some of them depending on each other (e.g. technology development that is relevant for multiple projects), some of them competing for resources. This creates additional management challenges and potential for waste. These issues are all very important, but – with the exception of firefighting discussed in the following section – not in the scope of this document. They are discussed in a separate white paper on program and multi-project management.

- The generic discussion of different types of wastes must always be critically reflected and interpreted in the concrete project context. For example, overengineering is discussed as a waste in this paper; but in the case of changing requirements, overengineering might in fact be important and value-creating, as it decreases rework when the requirements are increased.
- The minimization of waste is part of a Lean management strategy, but not its sole component. The minimization of waste is one cogwheel in successful Lean enterprise performance (also see Figure 1).
- Information generation and transfer are means to generate value, not ends in themselves. Generating value means fulfilling stakeholder requirements. While in manufacturing the stream of material through the system is used as a proxy for the value stream, in product development the information stream serves this purpose. It is also ultimately important to transform “dead” information into “actionable” knowledge.

In the following section, the eight categories of waste are introduced. Each is discussed in detail, as well as the relationship between the different waste drivers discussed. The next section focuses on methods for the identification and description of waste, while also pointing out approaches to minimize waste. The final section summarizes past work at LAI that is related to the management of waste in PD.

4 Categories of Waste

Figure 3: The Eight Types of Waste in Lean PD

Lean Product Development can be understood as creating a "product recipe" or set of specifications that are then transformed into a physical product or service. In manufacturing, different raw materials and product parts, or in general physical goods, flow through the Lean Production system. In Lean Product Development, this hardware is replaced by information in its various forms. As much as Lean Production focuses on waste associated with the transformation of physical goods, the waste identification in Lean Product Development focuses on the transformation of information. The resulting eight categories of waste are similar to those of Lean Production, but not identical (see Figure 3). Table 1 gives an overview of the different categories that different authors use to describe waste in Lean PD, as well as Taiichi Ohno's "classic" seven categories of waste. The discussion in the following sub-sections gives an overview description of the different types of waste. The descriptions are mostly based on (Bauch, 2004, Kato, 2005, McManus, 2005, Pessôa, 2008). Some typical examples for the eight types of waste are shown in Figure 4.

PD processes are very complex, and so is the waste that can occur within these processes. The details of the dependencies between the different types of waste are difficult to model exactly, but that does not mean that they should not be discussed in general terms. These complex relationships should also not hinder anyone from mapping a value stream or analyzing the waste occurring in their project (also see the examples in the following sections): While no analysis will ever be perfect and represent everything with every dependency, it is usually accurate enough to pinpoint the most important trouble spots.

Figure 5 gives one of the many possible interpretations of the complex relationship of the different types of waste (more details and different interpretations can for example be found in (Bauch, 2004, Kato, 2005, Pessôa, 2008)).

Table 1: The Different Types of Waste in Lean Product Development of Selected Authors

Categories of Waste	**(Pessôa, 2008)**	**(Bauch, 2004)**	**(Kato, 2005)**	**(McManus, 2005)**	**Millard (2001) [cited from (Oppenheim, 2004)]**	**Morgan (2002) [cited from(Oppenheim, 2004)]**	**(Morgan and Liker, 2006)**	**(Anand and Kodali, 2008)**	**(Ohno, 1998)**
Overproduction of information	Overproduction	Overproduction Unsynchronized processes	Overproduction of information	Overproduction	Overproduction: Creating Unnecessary information	External quality enforcement	Overproduction	Overproduction or early production	Overproduction
Overprocessing of information	Overprocessing	Overprocessing Re-invention	Overprocessing Re-invention	Overprocessing	Overprocessing: Working more than necessary to produce the outcome	Re-invention waste	Processing	Inappropriate processing or poor process design	Processing itself
Miscommunication of information	Transportation	Transport Handoff	Transportation of information Hand-offs	Transportation	Transportation: Inefficient transmittal of information	Hand off: Transfer of process between parties Transaction waste Ineffective communication	Conveyance	Transportation or movement	Transportation
Stockpiling of information	Inventory	Inventory	Inventory of Information	Inventory	Inventory: Keeping more information than needed.		Inventory	Unnecessary inventory	Stock on hand
Generating defective information	Defects	Defects	Defective information	Defective product	Defects: Insufficient quality of information, requiring rework	High process and arrival variation		Defects	Making defective products
Correcting information	Correction		Rework				Correction		
Waiting of people	Waiting	Waiting	Waiting of people	Waiting	Waiting: For information, data, inputs, approvals, releases etc	Waiting Large batch sizes System overutilization and expediting Unsynchronized concurrent processes	Waiting	Waiting or delays	Time on hand
Unnecessary movement of people	Motion	Movement	Motion of people	Unnecessary movement	Unnecessary movement: People having to move to gain access to information		Motion	Unnecessary motion or inefficient performance of design	Movement
Not part of this framework	Wishful thinking External events	Limited IT resources Lack of system discipline				Lack of system discipline			

Overproduction of information	•Two different groups creating the same deliverable •Delivering information too early
Overprocessing of information	•Overengineering of components and systems •Working on different IT systems, converting data back and forth
Miscommunication of information	•Large and long meetings, excessive email distribution lists •Unnecessary hand-offs instead of continuous responsibility
Stockpiling of information	•Saving information due to frequent interruptions •Creating large information repositories due to large batch sizes
Generating defective information	•Making errors in component and architecture design •Delivering obsolete information to following tasks
Correcting information	•Optimization iterations •Reworking deliverables due to changing targets
Waiting of people	•Waiting for long lead time activities to finish •Waiting due to unrealistic schedules
Unnecessary movement of people	•Obtaining information by walking up and down the hallway •Travelling to meetings

Figure 4: Examples for the different types of wastes

Overproduction of information directly leads to longer waiting times, as processing times increase. It also leads to a stockpiling of information, as the generated information cannot be processed in sync and must be stored. During storage, it is subject to deterioration and thus becomes defective. These defects, along with the defects occurring during the generation of the information, have to be corrected, leading to increased capacity utilization, resulting in the need to store more information, and also increasing the wait time of people for information. Miscommunication both supports the stockpiling of information, as well as generating defective information. Overprocessing occurs when processes are executed based on defective information, and may itself generate defective information (or increase the likelihood that defective information is generated). Information in inventory might encourage unnecessary movement of people, if the stored information is difficult to obtain. This in turn will lead to longer waiting periods. A possible interpretation of this system of interdependent wastes and their resulting dynamics is:

- The different types of waste form a complex causal network.
- This causal network is highly linked and contains only positive feedback loops. That means that regardless where and what type of waste occurs first, it will quickly trigger a cascade of other types of waste.

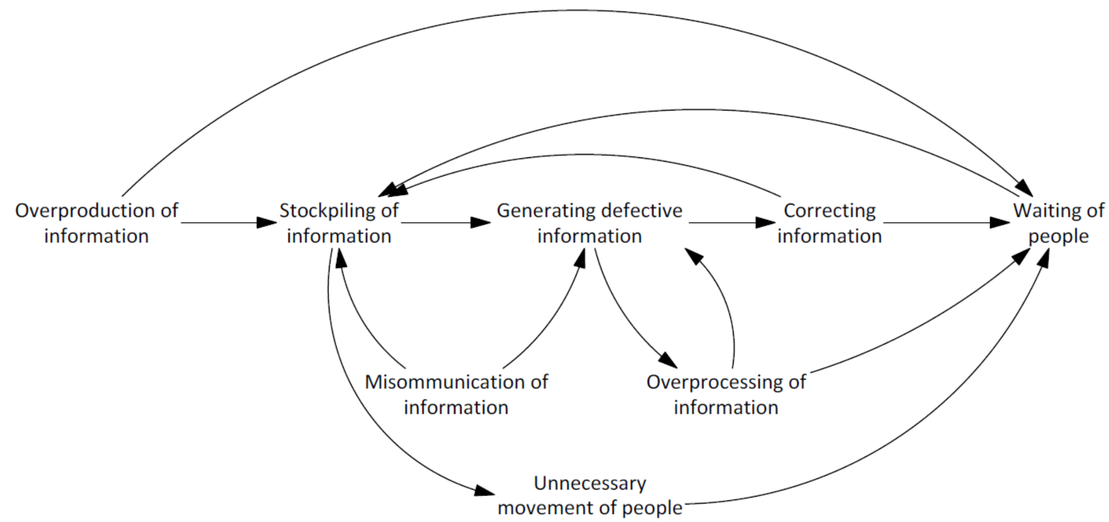
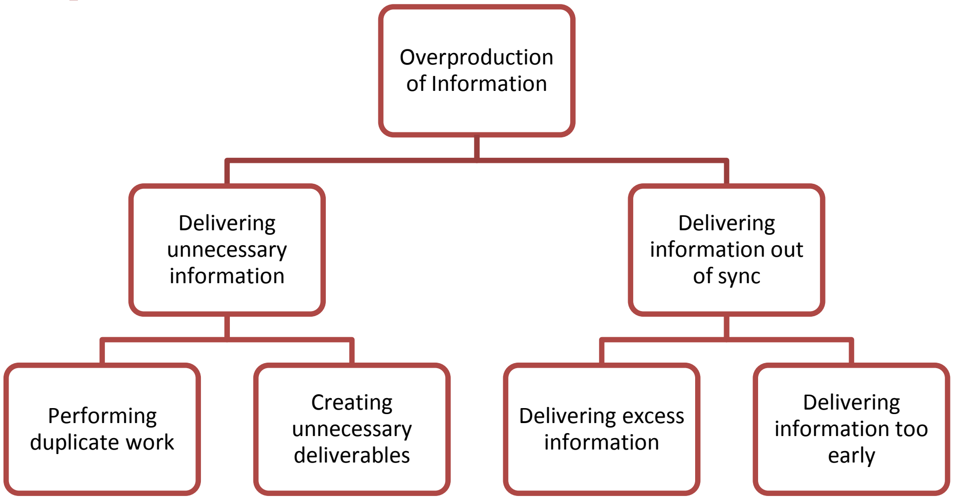


Figure 5: Possible interpretation of the complex relationship of different types of waste

- The positive feedback loops will drive the system towards ever increasing instability; the greater the instability, the faster the deterioration.
- The system is thus inherently unstable. It must be actively balanced by negative feedback loops. These consist in proactive practices to minimize the occurrence of waste, as well as quickly reacting to the occurrence of waste. They must also actively avoid reinforcing feedback loops that actually include waste countermeasures, such as the "firefighting virus" described in the information stockpiling section
- Two fundamental root causes are overproduction and miscommunication of information. Waiting of people can be interpreted as the corresponding end point of the dynamics. This can also be read as suboptimal processes, causing overproduction and miscommunication of information, leading, over several steps, to quality deterioration of the information, cost increase due to additionally required effort, and ultimately to schedule slippage due to a high amount of waiting times. This reinforces the importance of high quality PD processes that avoid overproduction and miscommunication of information.
- This interpretation is supported by the example data given at the beginning of the document, claiming that about 2/3 of the time is spend waiting.

4.1 Overproduction of information



Overproduction of information is usually only perceived in a subsequent process step. The waste of overproduction occurs whenever either superfluous or unnecessary information is delivered, or if information is delivered out of sync and therefore not useful.

Delivering unnecessary information includes any process output that is not necessary and has to be separated from useful information. Unnecessary information is different from defective information. Unnecessary information is not needed at all, whereas defective information is generated as a response to a valid information need, but that need is not fulfilled properly. The development of solutions in parallel or alternatives is not unnecessary information – as long as it is perceived as value-adding by the stakeholders (e.g. to generate data for decision-making on alternatives, or to create backup solutions for risky technologies).

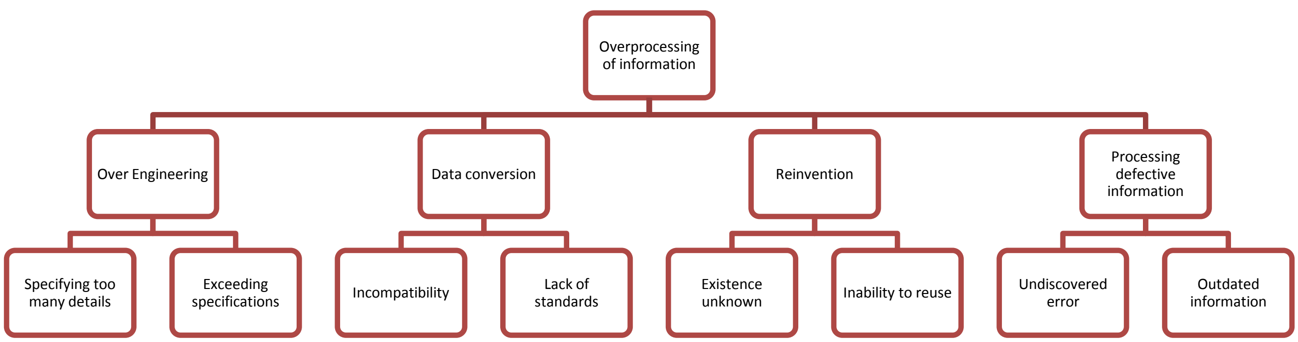
Not needed information might be the result of doing duplicate work or simply the creation of unnecessary deliverables. Duplicate work may occur when the company or the development team structure has redundant functions, the division of labor is unclear, communication and coordination is insufficient, or due to the company's or team's inability to conduct necessary adjustments to the division of labor when the need arises during a project. This waste might also occur during organizational or process changes, when people have to relearn their network of communication and collaboration. Resistance to change or a lack of specific training to implement the new processes may play a role.

Creating unnecessary deliverables might be a consequence of an insufficient standard PD process, an immature or incomplete Statement of Work, or due to errors in the project planning and implementation. Team members may create unnecessary deliverables out of their own initiative, to support for example "pet projects" or to hedge against uncertainty (e.g. development of an alternative solution before a primary solution was tested). Creating unnecessary deliverables differs from overprocessing, where the information is still needed but exceeds quality or functional specifications.

Delivering information out of sync means that the information cannot be used immediately, either because of lack of capacity to process it (delivery of excess information) or because other, complementary information needed to proceed is unavailable (delivery too early). It is important to judge whether information delivery that occurred too early is actually the result of poorly synchronized processes – if early information delivery was planned, for example to create buffers on the critical path, or because of remaining slack time after careful planning, it may be acceptable or even valuable by creating process robustness.

Unsynchronized processes are either the consequence of a poorly planned schedule or the result of issues during the development execution. A schedule may be originally unsynchronized due to a non-optimized standard process, the lack of the needed resources to define a smooth work flow, or simply by a lack of planning effort to synchronize information flows. Batch processing of information can overload the next process, resulting in idle wait times followed by large inventories of waiting information when it is delivered too early. Due to the high number of uncertainties in PD, unexpected events might occur at any time and their effects disrupt a synchronized plan.

4.2 Overprocessing of information



Overprocessing of information addresses unnecessary information processing (while overproduction is related to the output of a process and delivery of information). Overprocessing can be divided into overengineering (generating information beyond the required specifications), data conversion (converting data between information systems or between people), reinvention (existing components or technical solutions are not reused) and processing defective information (defective information is used as process input). The ability to judge whether or not overprocessing occurs is strongly dependent on the stability and accuracy of the relevant requirements. If these requirements are still fluid, or can only be loosely defined due to technological uncertainty, designing components to perform above base-case assumptions adds robustness to the design and can thus be value-creating.

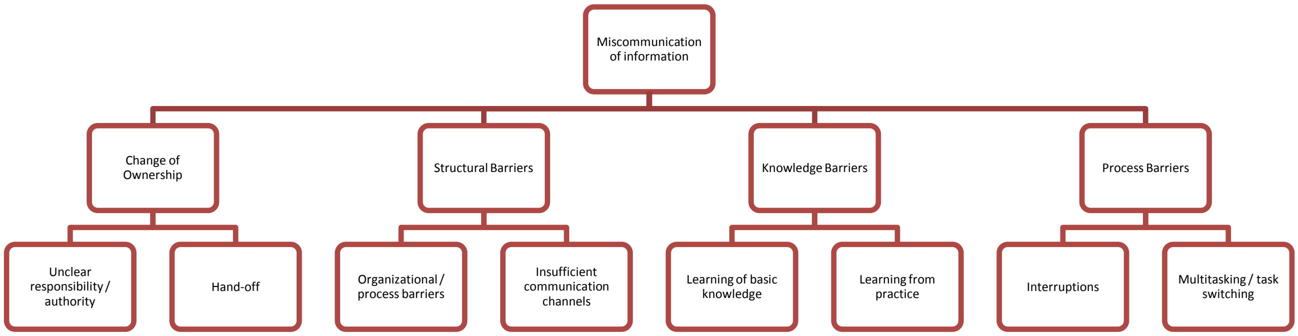
When ***overengineering***, a designer specifies too many details or exceeds the required specifications. The designer not only wastes his own time and resources, but overengineering can also set unnecessarily rigid tolerances or other requirements that constrain the following development activities and later also the production process. Overengineering might be attributed to an organizational or personal trait of perfectionism, but is often a consequence of a lack of knowledge regarding required functionality and detail. This might be due to poor understanding of downstream tasks, lack of constructive discussions with other team members and colleagues (both up- and downstream), or insufficient specifications or process standards.

Data conversion addresses the processing of data to make it fit for communication between different reference systems, without changing the actual content (also see the discussion of miscommunication). It includes for example converting information between different measurement systems, IT systems or the translation into different languages. It may also refer to converting information more broadly, for example converting detailed measurement data into a common-language management presentation. Reformatting and reprocessing reports for different stakeholders, without actually changing the content, is also waste through data conversion. While data conversion is necessary at different stages, it is waste if it occurs due to avoidable incompatibility and connectivity (e.g. different types or versions of IT systems), or due to a lack of clear process guidelines that specify how and what information is to be represented (e.g. timing, number and detail of reports, or conflicting formatting requirements).

Re-invention applies to processes, solutions, methods, components and products which already exist or rather would only require some modifications to make them fit for the use at hand. Re-invention may be a consequence of a simple lack of knowledge of existing solutions, or the inability to reuse existing knowledge. This can result from poor expertise sharing, an insufficient system to capture lessons learned and manage existing knowledge, security issues (such as ITAR restrictions) preventing reuse, or product architecture concepts that do not favor modularity and reuse on a technical level. Re-invention might also cause overprocessing in downstream activities, as these are forced to re-invent certain solutions as well (e.g. at the interfaces to the re-invented component).

Processing defective information is one of the wastes caused by generating defective information (next to rework and waiting, discussed in the respective sections). Processing defective information creates waste through overprocessing, as the resulting information is in most cases useless, and the processing therefore wasted. The use of defective information can either be attributed to receiving defective information from upstream processes and not noticing the error, or by using information that is outdated (“rotten”) at the time of continued processing (also see the discussion of information stockpiling).

4.3 Miscommunication of information



Communication of information includes its encoding, transmission and decoding. Fundamentally, as information is not changed, this process does not add value. However, effective and efficient communication is of paramount importance for successful PD projects. The waste of ***miscommunication*** occurs either through inefficient communication, i.e. more resources than needed are used to conduct the communication (e.g. data format conversion instead of using unified IT systems, drafting of memos where phone calls would have sufficed), or ineffective communication, i.e. information is communicated unnecessarily (e.g. discussions in large meetings that only concern a fraction of the people present, excessive email distribution lists). Communication is closely linked to organization and processes, as these largely define the need for communication.

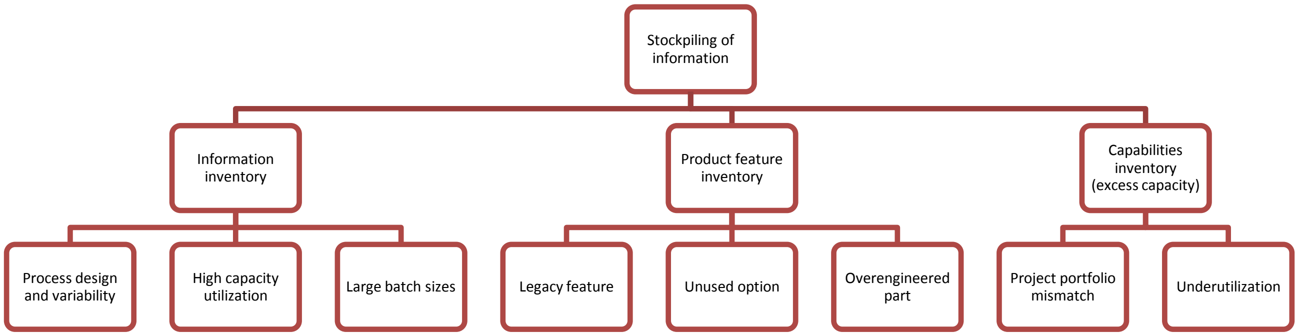
Change of ownership addresses miscommunication occurring when the responsibility for a certain piece of information changes. Change of ownership has two main causes: unclear responsibility or authority, and hand-offs. By not having a clear assignment of responsibility for outcomes, feedback and review processes, people are tempted to send and receive pieces of work as a way to prevent blame for failures and mistakes. If responsibility and authority are not aligned, people have to ask permission excessively to continue to work or release information to subsequent process steps. Hand-offs occur when the ownership of information changes. In this case, communication must occur. Information has to be encoded (writing of reports, translating notes from native language to working language and back, converting data etc.), transmitted (electronically or via mail), and decoded (re-translating, understanding etc.). If hand-offs are unnecessary, these activities are wasted. If they are necessary but then imperfectly executed (e.g. data incompatibility or conversion, incompleteness), waste through rework or overprocessing occurs.

Structural barriers cause waste by organizational and process barriers, or by nonexistent or unreliable communication channels. Organizational barriers can be created by the physical location and distribution of team members, creating barriers to communication through language, time zones (co-location vs. international teams) and communication technology (e.g. awkward video conferencing systems). Lengthy approval processes (e.g. security clearance and/or intellectual property safeguards) can also result in waste, if these processes are not absolutely necessary to conduct communication. In general, the hierarchical and process structure of the organization and the written and unwritten communication rules between functions can have a significant impact (e.g. having to communicate two levels up, and then two levels down again to reach a colleague down the hall). The communication channel itself creates miscommunication waste if it is not adequate for the task at hand regarding bandwidth (text/audio/video) and synchronicity (real time vs. asynchronous), e.g. conducting group discussions via email.

A ***knowledge barrier*** exists whenever a person does not have the necessary knowledge to perform a task. This results in additional communication, either within or outside the team, to acquire the necessary knowledge and skills to perform the task. Whether or not this activity is value creating, necessary waste or waste depends on the circumstances: If it is a learning activity that creates a new and valuable capability within the company, it adds value as it now allows executing a new type of value creation for the stakeholders. If the capability already exists, but is unavailable (e.g. capacity constraints and re-scheduling of activity not possible), it is necessary but non value adding. If the capability existed but was not used (e.g. due to a lack of transparency regarding competencies of team members and colleagues), it is waste.

Work continuity barriers are caused by interruptions that require a person to change the subject of their work (also see the discussion of information stockpiling). It may require communication with oneself, in the form of notes, memos or codifying current implicit information in any way, or transferring the task to somebody else with the ensuing communication process. After the interruption, the process is then usually reversed, leading to additional set-up time due to the re-communication and understanding process involving the changed information. Both interruptions as well as scheduled multi-tasking and task switching may create continuity barriers, creating waste in the form of miscommunication. Interruptions are mostly related to solving not directly related issues inside or outside the project. Multitasking and task switching can occur within a project, when different activities are executed by one person, between parallel projects if a person is allocated to more than one project at any time, or between functional and project activities in a matrix organization. Whereas there are justifications for all of these organizational constructs, these advantages have to be evaluated in the light of the waste that they create.

4.4 Stockpiling of information



The ***stockpiling of information*** leads to the build-up of information inventories, and, in the broader sense, inventories of product features and capabilities. In general, inventory is considered waste, as it incurs cost to maintain (storage costs and cost of bound capital), exposes the stored item to obsolescence risk ("information rot"), and in many cases also indicates that the underlying processes are not well synchronized. If information is stockpiled deliberately, for example in the form of currently unused product features that enable a re-use in later assemblies, the value stream analysis has to account for these stakeholder needs and ensure that these types of information inventory are treated accordingly as value. Similar to the waste of overprocessing, stockpiling of information can be value-adding in a broader multi-project system, where for example features may not be necessary for a single product, but necessary to execute an overall platform or commonality strategy.

Information inventory exists between process steps. When one activity is finished, the resulting information is stored until the downstream activity continues with the processing. In a perfectly balanced PD system, information inventory is zero, as all information is currently being processed. While storing information in itself is not wasteful, it is considered a waste in product development for two main reasons: a) the size of information inventory is an indicator for process quality: the better a process is run, the better different steps are synchronized, the shorter information needs to be stored between processing, and the smaller the amount and time of stored information; b) stored information "rots", thus creates rework and in turn destabilizes the PD process even further, resulting in more information needing to be stored: Information is subject to obsolescence at a rate of about 11% per month, as the PD project progresses while part of the information is frozen in inventory. The required rework per "rotten" information item is on average 54% of the initial effort, leading to an average of 6% of effort of the inventory wasted per month in storage (Kato, 2005). This in turn creates unexpected rework in the PD process, leading to firefighting, as people are re-assigned from their routine tasks to take care of the rework. This destabilizes the process, increasing not only the probability to make errors (create defective information), but also the size of information in inventory, as activities are interrupted more frequently and information has to be stored between processing. This in turn increases the amount of information being exposed to obsolescence risk, increasing the amount of rework.

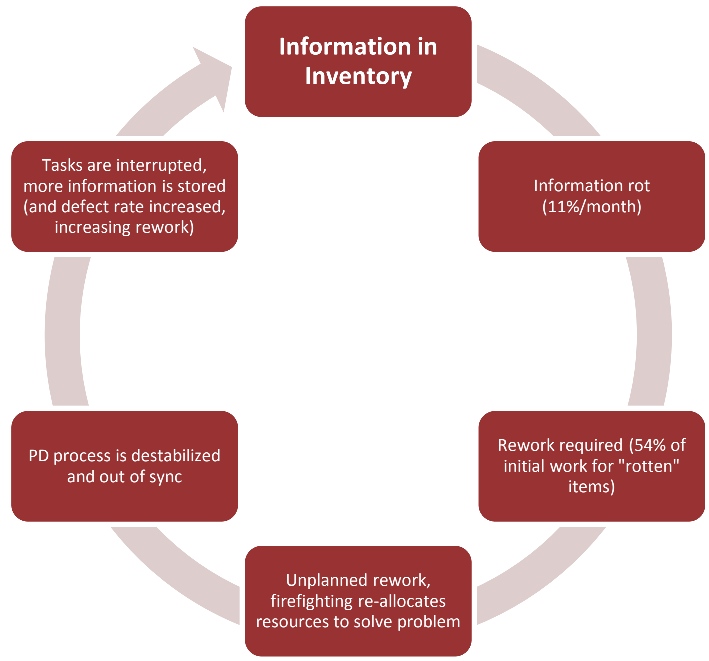


Figure 6: The Vicious Circle - Information in Inventory Creating Information in Inventory

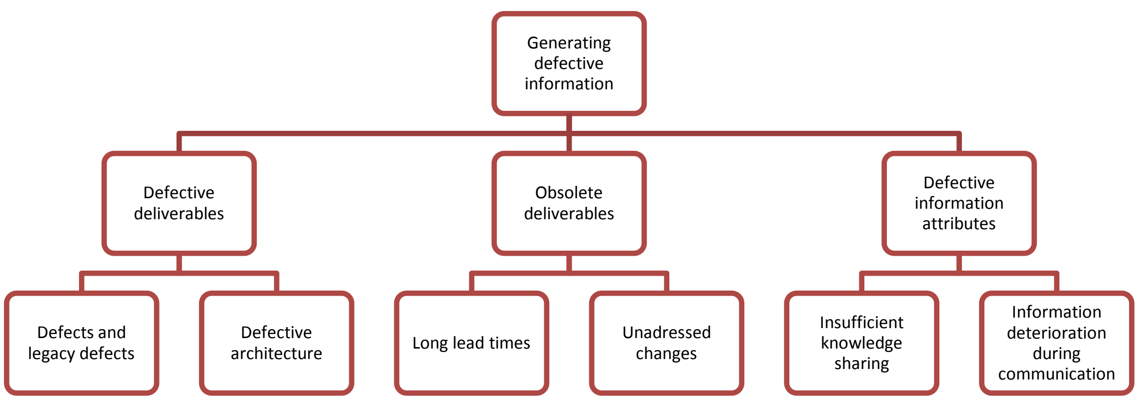
This vicious circle of information in inventory creating more information in inventory overlaps with the vicious circle of firefighting creating more firefighting: Many organizations are trapped in a mode of operation where problems in a project lead to a re-allocation of resources to solve that problem, only creating more problems in the areas where the resources have been drawn from (often other projects, thus spreading the "inventory and firefighting virus").

Storing information between process steps occurs when the execution of information-processing activities is not continuous and synchronized. Underlying reasons might be a high variability in process execution, leading to widely varying process times and thus a low degree of synchronization. A high utilization of capacities (e.g. engineers) can be a cause for such variations, as small disturbances cannot be buffered and immediately lead to delays. Large batch sizes in information processing may also lead to waiting times. If processes are not executed concurrently, i.e. a downstream process has to wait until an upstream process is completely executed (e.g. finishing the complete detailed design before production is involved), this not only leads to significant iterations, but also to wait times (also see the discussion of the waste 'waiting').

Product feature inventory is the result of unnecessary (often legacy) features, options, or overengineered components. As many products represent incremental improvements of existing systems, information or design standards can 'accumulate' as legacy systems. Mobile phones for example looked and handled a lot like mobile handsets of fixed line phones, until the iPhone marked a radical break from legacy features. The market success shows how customers appreciate the reduction of the old 'feature inventory'. Options for future modifications of a system (e.g. hull extension of an aircraft) are potential waste, if these are not driven by concrete or anticipated future customer needs. Similarly, if overengineered parts are incorporated into the final system, these also represent waste in the form of excessive embodied information in the product. In the best case, these excess specifications only raise development and production cost, in the worst case they lower the delivered value by increasing the complexity of using the system, increasing life cycle cost for the customer, or decreasing its reliability.

An ***inventory of capabilities and / or capacities*** is also a special type of information inventory: Here, information in the form of knowledge is embodied in enterprise capabilities (e.g. people's skills, processes, culture, IT systems). These 'stored capabilities' are either value creating (or necessary waste), if they are important to deliver value according to the current stakeholder needs (i.e. match the current portfolio of projects and stakeholders). But as the environment and the stakeholder needs keep evolving, capabilities become outdated (e.g. being able to develop life support systems is no longer relevant if the company's new focus is unmanned vehicles). If they are not adapted to the new scenarios of project challenges and stakeholder needs, they are wasted (they are also closely linked to the continued use of legacy system components and technologies, see above). Similarly, an inventory of capabilities is created when new capabilities for emergent project situations and stakeholder needs are developed: They remain 'in inventory' until the new situation or stakeholder need actually manifests itself, and might be wasted if it never happens (e.g. developing the capability to share development costs in international collaborative R&D programs). Similarly, value-creating capabilities might exist at excess capacity, causing waste. As discussed in the section of 'waste through waiting', the decision to have (over-) capacity available must be traded off against an over-utilization of capacity, leading to instability and delays.

4.5 Generating defective information



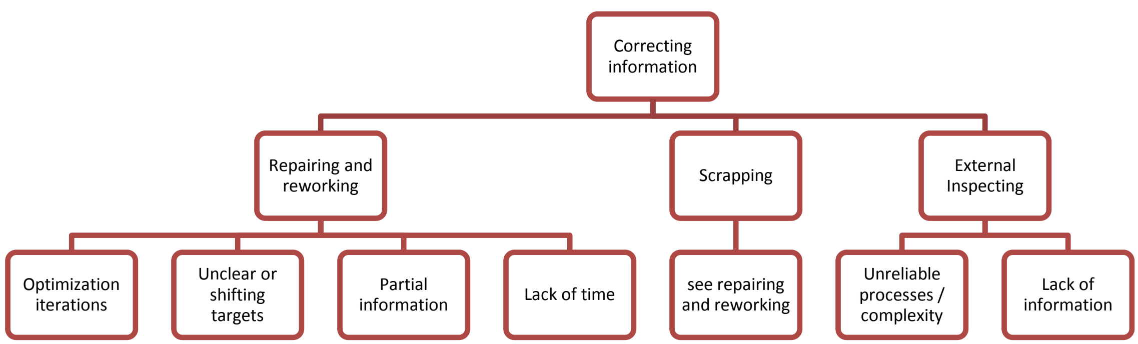
Generating defective information applies to the quality of system components and architecture, the deliverables being up-to-date, and the defects that might occur during communication. It directly leads to three types of other waste: Processing defective information, rework and waiting.

Defective deliverables include not only the final product, but also parts, subsystem and the general system architecture that are created during the development project. Legacy defects from reused components might occur, where defects either remained dormant or their discovery was not documented or corrected. They might only become apparent when new features are added or the components are subjected to more stress. The architecture itself might also contain defects, either increasing the complexity of the task unnecessarily and thus increasing cost and reducing reliability. It might also hinder components to perform their intended function, or lead to unnecessary interference.

Obsolete deliverables were defect-free at an earlier stage, but now have to be considered defective as they do not meet current requirements. Long lead times, long periods in inventory or unaddressed changes might be reasons for obsolescence (also see the discussion regarding the waste of information stockpiling).

Defective information attributes relate to contextual information, or the quality of information sharing. During communication, information is necessarily transformed and often condensed, opening the possibility of generating defects (e.g. in data conversion, but also in reports or team meetings). (Graebsch, 2005) discusses four categories to characterize information quality: 1. Intrinsic information quality: Accuracy, objectivity, believability and reputation; 2. Availability of the information: Accessibility, security, transferability, originality and availability; 3. Contextual information quality: Relevancy, value-added, timeliness, completeness and amount of information; and 4: Representational information quality: Interpretability, ease of understanding, concise representation and consistent representation. Deficiencies in one or more of these attributes do not necessarily mean that the information itself is defective or useless; some deficiencies may be even compensated by the designer's knowledge. Defective information attributes might however lead to a wrong interpretation of information (e.g. regarding the reliability of measurements or the relative importance of different stakeholder preferences).

4.6 Correcting information

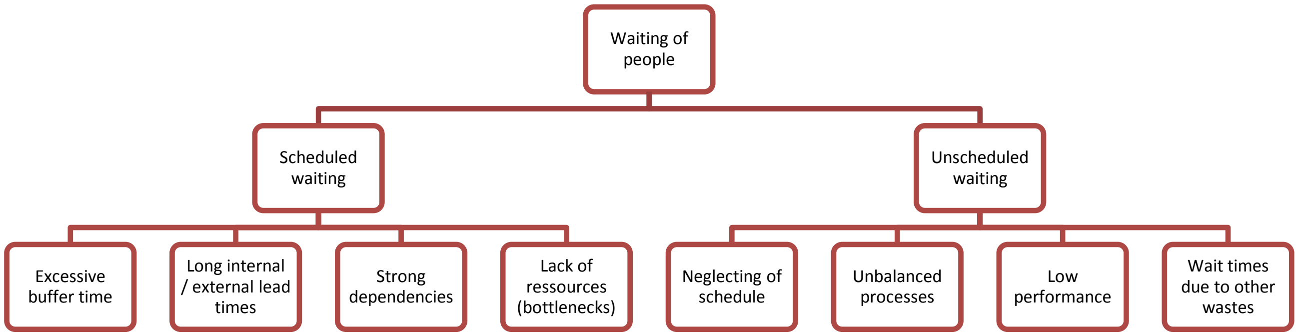


Correcting information includes waste associated with the repairing or reworking of information, the scrapping (and subsequent complete replacement) of information, and inspections associated with ensuring information correctness. Both reworked and scrapping of information have the same underlying causes; the decision on whether information can be reworked to elevate it to the necessary level of quality, or has to be scrapped and redone completely, depends on the extent of the deficiencies.

Repairing and reworking aims at improving existing information to make it meet requirements and quality standards. ***Scrapping*** has the same goals, but involves starting from a clean sheet. Reworking is mainly driven by optimization iterations that become necessary as new information emerges. These iterations might be planned or unplanned. Whether small and fast (but more) iterations are more efficient than long and slower (but therefore also fewer) iteration cycles depends on the specific project, the level of uncertainty and the level of achievable integration between different functions. Shifting targets or unclear targets (e.g. stakeholder needs) also often lead to rework, as the value definition changes. Partial or incomplete information (which might already be considered 'defective') also leads to defective output information. This not only applies to the information necessary to generate the desired output information, but also to information that is necessary to analyze and judge the quality of one's output, and determine whether or not it is defect-free and can be passed on to the next function. Time pressure is an important factor that leads to defects in several ways: First, time pressure entices people to take 'short cuts' and ignore established processes and best practices, thus leading to defects. Quick fixes and patchwork are preferred to finding and fixing error sources. The psychological effects of stress also encourage errors. Also, more often than not time pressure is combined with large information inventories, as processes got out of sync, thus increasing the probability of working on defective or outdated data (also see the discussion on information stockpiling). Time pressure also encourages passing on information that has not been verified, or where the person in charge is unsure itself in regard to its quality.

External inspecting of information refers to checks and quality audits of information by people other than those involved in its generation. Fundamentally, these activities are waste, as the goal must be to ensure that information can be produced error-free, and that the persons responsible for their generation have the ability to verify the quality of their work. However, unreliable processes or high levels of complexity might lead to external inspections and reviews. These become also necessary if the quality of the information can only be judged by other persons. The amount of "necessary waste" through reviews depends on the quality of process execution and the ability and willingness to judge the quality of one's own work. In general, waste in the form of additional, early inspections prevent significant amounts of waste through random discoveries of errors further downstream. Still, perfect quality each and every time must remain the goal.

4.7 Waiting of people (for information)



Waiting means that a person is idle, i.e. the value stream is not flowing. Waiting of people occurs when critical information to execute a task is not available. This may include an authorization to perform the work; information input to be processed; or resources to be used during execution. Waiting for authorization means that, although the process has all the necessary inputs and resources, it is not authorized to start. Authorization might be a pre-defined moment or control input to trigger the process. Waiting for input or resource means that the previous processes did not deliver necessary information on time or that another process is keeping a resource for longer than anticipated. Wait can be "scheduled" or "unscheduled", i.e. occurring expectedly or unexpectedly.

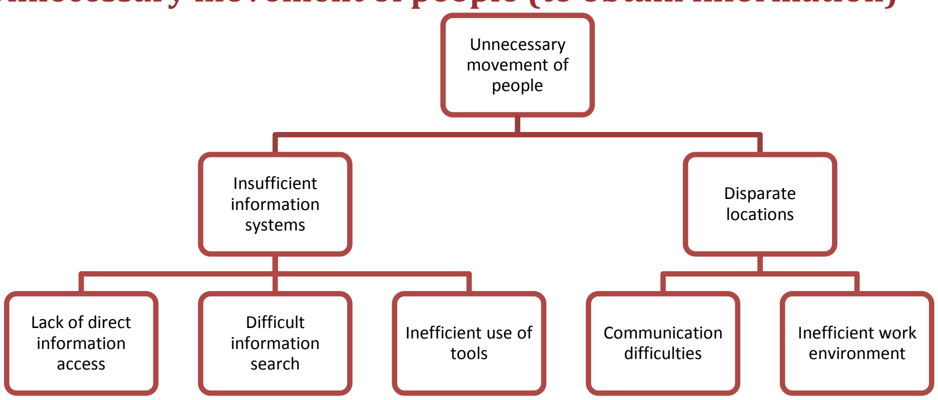
In the case of ***scheduled waiting***, people, information and / or resources are planned to stay idle during some time. Wait times are defined during planning as a consequence of for example:

- Excessive buffer time added by a planner. More than necessary reserve time is included between activities due to perceived risks, historical delays, or imposed by the standard process and guidelines (i.e. buffer must be equal to X% of the critical path). In the case of significant uncertainty, the decision of what is an appropriate and what is an excessive buffer will probably be somewhat subjective. The challenge then is to create a work breakdown structure that allows waiting engineers to perform other important work.
- Long internal or external lead times. Internal and / or external lead times for key activities at certain stages may be very long (e.g. delivery of prototypes, safety approvals, specialized tests). This imposes wait time on other tasks until the project is cleared to proceed.
- Existence of strong of dependencies between tasks. The complex interdependencies between tasks often do not allow creating a plan without wait times.
- Lack of resources. There are no available resources to perform independent tasks in parallel, thus one has to wait until the other is completed.

Unscheduled waiting is the unexpected wait time that may occur during the development due to one of the following factors:

- Lack of schedule discipline. The scheduled is either neglected by team member and / or not enforced by project leaders and management.
- Requirement or Design Changes. Changes can cause the duration of activities to differ from planning, thus leading to wait times in the following process steps.
- Unbalanced processes or planned schedule is too tight. The processes are not coordinated and / or their capacity not dimensioned properly during planning. This leads to unrealistic schedules, if insufficient or unbalanced capacities are squeezed to fit to targets or expectations. The original schedule is then unrealistic and prone to delays. While excessive buffer result in planned wait times, schedules that are too tightly planned result in unscheduled wait. Too tight schedules may lead to a complete lack of directions during execution, since they are quickly disregarded by team members.
- Performance lower than expectations. Performance estimation may have been overly optimistic. Excessive performance requirements over prolonged periods of time may lead to fatigue in people and equipment alike. Also, people or equipment assigned to the execution of a task may not be suitable, causing delays.
- Other wastes. Other wastes, like overproduction, transportation, overprocessing, motion, reworking defective information etc. usually disturb the value flow in PD and thus lead to additional wait times.

4.8 Unnecessary movement of people (to obtain information)



Unnecessary movement of people can be typified as unnecessary human motion to obtain information due to insufficient information systems, disparate locations, and inefficient use of equipment, tools and techniques. It wastes time, as people are in transit rather than creating value, and money, and it may incur travel costs.

A ***badly designed, insufficient information system*** may contribute to motion by not allowing the needed and available information to be directly accessible to the user, or by requiring timely searches (information hunt) to be found. People have to either leave their workplace to make a physical search or directly interact with people, or search for examples through complex project directory structures on a server. The inefficient use of equipment, tools and techniques that are designed to support the people using them to optimize their access to information leads to additional waste. Often, tools to manage information can become very complex (e.g. integrated 3D CAD and product data management systems or collaboration and workflow tools). They represent not only large investments, but by using them improperly, they do not yield the expected benefits and might even contribute significantly to wasted time. The tools and the processes they are supposed to support have to be optimized for seamless integration, the tools itself must be of manageable complexity (regarding introduction, use and continuous servicing), and their proper use must be ensured by adequate training and feedback cycles.

In the case of ***disparate locations***, the information owner or its storage place is not directly accessible from the working environment. The local distance of departments and facilities can have negative impacts on the project team work. A loss of time occurs (and depending on the distance, travel expenses occur) to move to and from the remote location. The distance acts as a barrier and encourages people not to make the trip or other effort to obtain necessary information or interact with colleagues (for a detailed discussion on the effect of distance on communication, see (Allen, 1984)). Also, remoteness may inhibit team formation by making direct and personal interaction rare that is required to build a personal professional relationship.

5 Identification of Waste

5.1 The Value Stream Mapping Process

One process for identifying value and waste in PD processes is Product Development Value Stream Mapping, or PDVSM. As LAI published a dedicated guide, the 'Product Development Value Stream Mapping (PDVSM) Manual' (McManus, 2005), the technique is only briefly introduced here to give a general overview. The manual addresses product development (PD) personnel working on improving their own processes, and lean change agents working with them. Its aim is to provide practical guidance for applying lean concepts to PD process improvement. The manual concentrates on processes on the scale of component-level

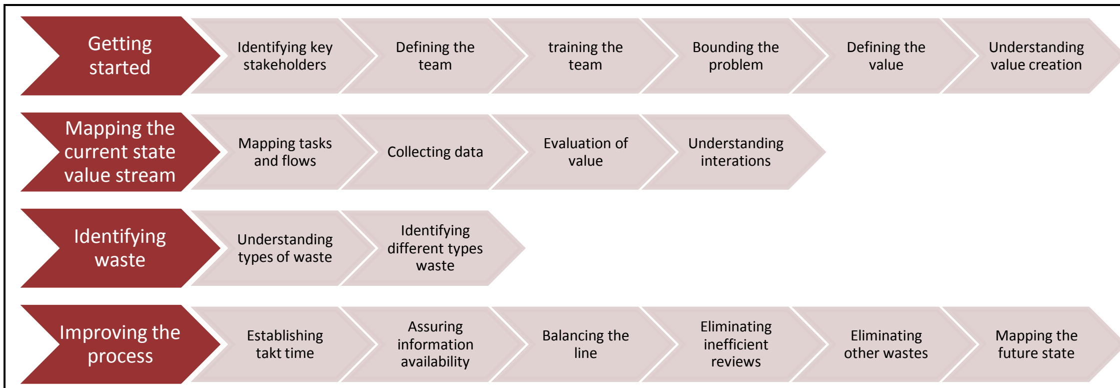


Figure 7: The PD Value Stream Mapping Process

hardware development.

The PDVSM consists of four main processes (also see Figure 7): A preparatory phase (‘Getting started’), the mapping of the current value stream, the identification of the different types of waste in the current value stream, and lastly the improvement of the current process.

During the ***preparatory phase***, first, the relevant stakeholders are identified. Then, the team that conducts the value stream analysis is defined and trained. The problem is then bounded, i.e. the scope of the analysis is delimited. Based on scope and stakeholders, the value is defined. Lastly, the processes that create value are analyzed. During the ***mapping of the state of the current value stream***, first the tasks and flows between the tasks are mapped. Then, the necessary data to describe the value stream are collected and the generated value evaluated. As iterations play an important role in PD, those are analyzed in detail in the last step. The third step focuses on the ***identification of the different types of waste***. First, the different types of waste have to be discussed, understood and agreed upon by all team members, before they are identified along the value stream in the second step (for an expanded set of waste notations, see (Kato, 2005)). The fourth step, ***improving the process***, may include the establishment of takt time to synchronize processes. Other specific actions include assuring information availability, balancing of the workload and capacity of activities, and eliminating unnecessary or inefficient reviews. Other measures addressing more types of waste are also discussed in detail. Finally, the future state of the value stream is mapped to serve as guidance for the implementation of the improvement actions.

5.2 Example of Value Stream Analysis and quantification of waste

The following analysis of a PD project illustrates the concept of the different types of waste. It is based on (Kato, 2005). The identified types of waste as well as their importance cannot be generalized; they strongly depend on the type of project, as well as the executing company's capabilities. The following example is based on detailed observations of three projects in two companies in Japan. The development projects dealt with embedded software for high tech equipment. The development groups consisted of 6-7 people each (engineers and project manager). The projects were observed between 17 and 50 weeks.

The projects were observed in detail; the processes were mapped according to the value stream mapping method and the occurrence of the different types of waste noted for every activity and engineer. The data was quantitatively collected through detailed interviews and observations (see Figure 8 for an example).

The graph in Figure 9 shows the results of the quantitative analysis of the different types of waste that were measured in wasted engineering hours per 50 engineering weeks: In this example, correcting defective information consumed an average of 1343 hours per year. The second most important waste – also being responsible for the first – was generating defective information with 697 hours. Unnecessary movement of people consumed 605 hours, about the same as overprocessing of information (600 hours). Miscommunication, overproduction and waiting of people were less important, probably due to the relatively small team size.

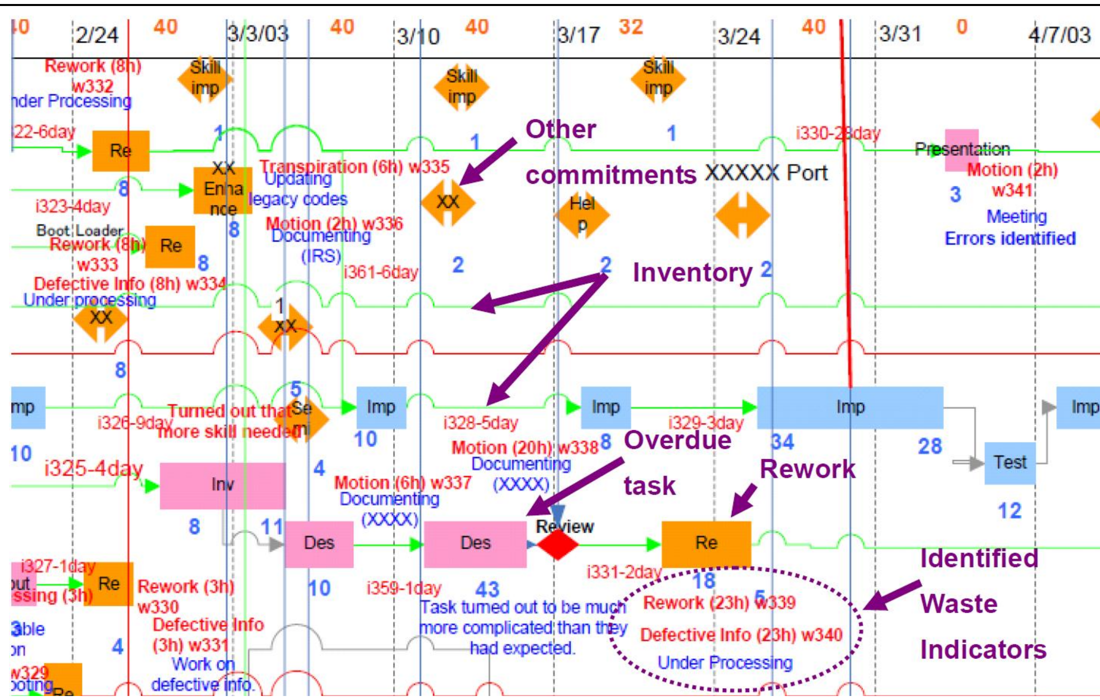


Figure 8: Example of detailed and quantitative Value Stream Map (Kato, 2005)

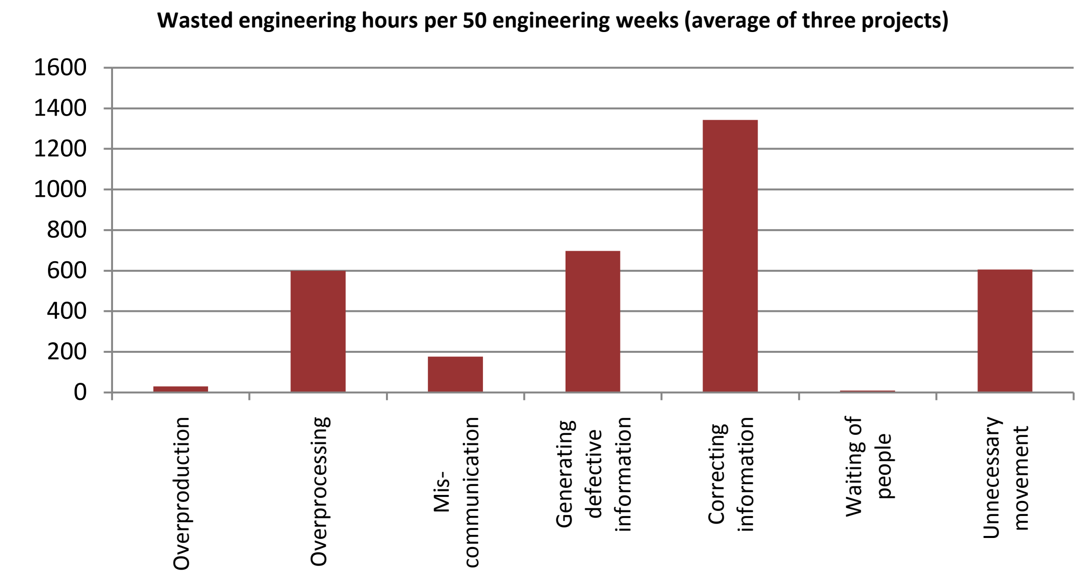


Figure 9: Example results of quantitative Value Stream Analysis (Stockpiling of information separately)

Figure 11 shows the trend for the waste of 'correcting information' in project three (other two projects show similar results): The amount of time spent on rework increases strongly with the progress of the project. While no time was wasted on rework in the first five weeks of the project, that value increases to almost 50 hours in weeks 26-30.

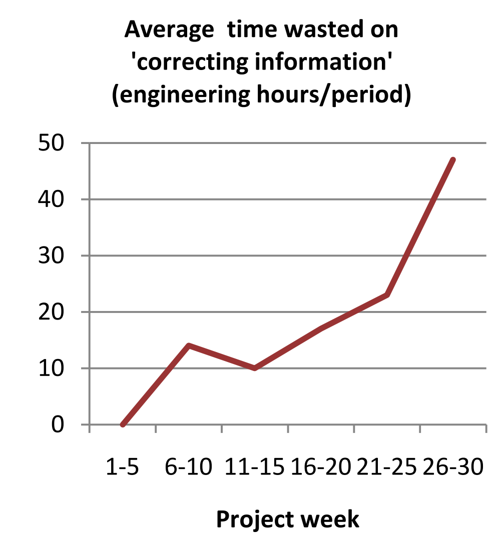


Figure 11: Development of time wasted on 'correcting information'

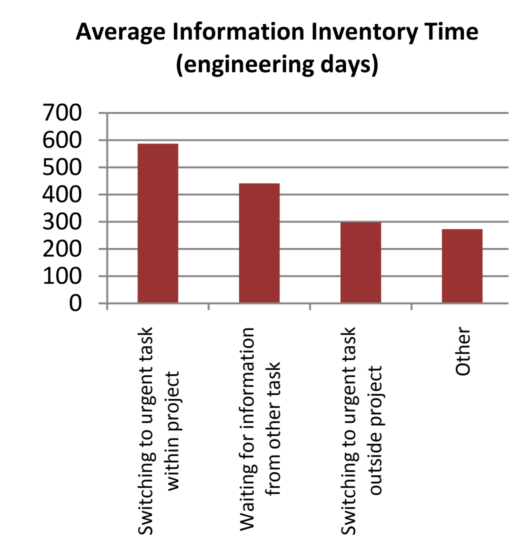


Figure 10: Main causes for 'stockpiling information'

The waste associated with stockpiling of information was measured by days of information in inventory (see Figure 10). Process instabilities were the most important drivers, both unplanned task switching within the project as well as across projects (firefighting) being in the top 3 sources of waste. The second most important reason – waiting for information from other tasks – is also associated with asynchronous processes.

The concern with information in inventory is the fact that as the information becomes obsolete, it 'rots' (for several reasons, see discussion of 'information stockpiling'). The analysis of the projects has shown that information once it is in inventory starts to rot with about 0.54% per day. Of the information affected by rot, the rework ratio was on average 53%. That means that 53% of the time spent to originally create the information had to be spent again to correct it. The effect for 100 engineering days is illustrated in Figure 12.

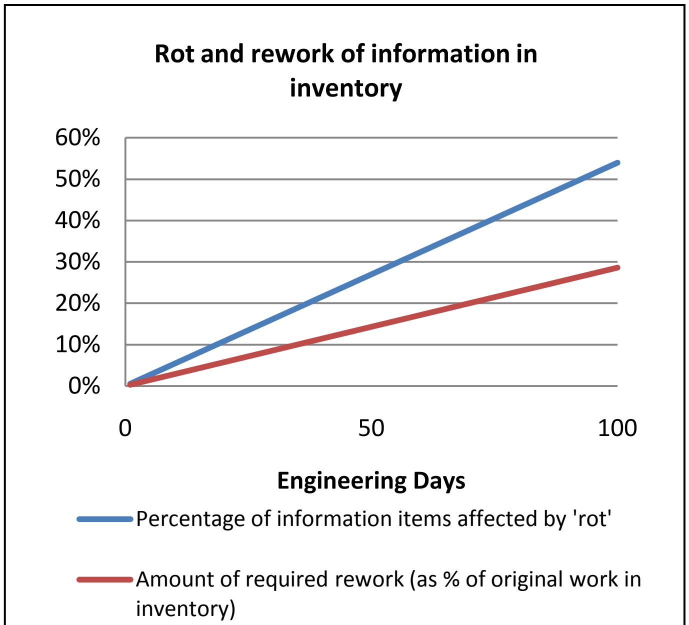


Figure 12: Degradation of information in inventory

5.3 Overcoming Waste in Product Development – Some Starting Points

Waste in product development is a symptom for a system not running at peak efficiency and effectiveness. In order to optimize the PD system, a PD value stream analysis helps, as a first step, to identify different types of waste and their relative impact on the value creation. In order to reduce the amount of waste, systemic actions – that is improvements that are effective in different parts of the process – can be powerful. For example, improving the skill set and motivation of the engineers working in a project will for example not only reduce the numbers of errors they make, but also improve their ability to check and quality control their own work, thus enabling faster corrections.

A number of improvement suggestions relating to systems engineering activities (Oppenheim et al., 2010) are summarized in Table 2, structured along Lean Principles of (Womack and Jones, 1998).

Table 2: Lean Enablers to Minimize Waste in PD (Oppenheim et al., 2010)

Lean Principle	Lean Enabler for Systems Engineering (top level categories)
Value	• Follow all practices for the requirements capture and development of the INCOSE handbook process • Establish the value of the end product or system to the customer • Frequently involve the customer
Map the value stream	• Plan the program according to the recommended INCOSE handbook process • Map the systems engineering and product development value stream and eliminate all non-value adding elements • Plan for frontloading the program • Plan to develop only what needs developing • Plan to prevent potential conflicts with suppliers • Plan leading indicators and metrics to manage the program
Flow	• Execute the program according the INCOSE handbook process • Clarify, derive, prioritize requirements early and often during execution • Front load architectural design and implementation • Encourage system engineers to accept responsibility for coordination of product development activities • Use efficient and effective coordination and communication • Promote smooth systems engineering flow • Make the program visible to all • Use lean tools
Pull	• Tailor for a given program according to the INCOSE handbook process • Pull tasks and outputs based on need, and reject others as waste
Perfection	• Pursue continuous improvement according to the INCOSE handbook process • Strive for excellence of systems engineering processes • Use lessons learned from past programs • Develop perfect communication, coordination and collaboration policy across people and processes • For every program use a chief engineer role to lead and integrate the program from start to finish • Drive out waste through design standardization, process standardization and skill-set standardization • Promote all complementary continuous improvement methods to draw best energy and creativity from all employees
Respect for people	• Pursue people management according to the INCOSE handbook process • Build an organization based on respect for people • Expect and support engineers to strive for technical excellence • Nurture a learning environment • Treat people as the most valued assets, not as commodities

6 An Overview of LAI's Research on Value and Waste

This section briefly describes the content of the various relevant LAI papers.

Table 3: Overview of research at LAI related to the concepts of Value and Waste in Lean PD

Area / Author	Publication	Citation
Types of waste and identification		
Pessôa, Marcus V.P.	"Weaving the waste net: a model to the product development system low performance drivers and its causes", LAI White Paper 08-01.	(Pessôa, 2008) **Download link**
Pessoa, Marcus V.P., Warren Seering and Eric Rebentisch.	"Understanding the Waste Net: A Method for Waste Elimination Prioritization in Product Development." ASME 2008 International Design Engineering Technical Conferences (DETC), New York, NY, August 3-6, 2008.	(Pessôa et al., 2008) **Download link**
Pessôa, Marcus V. P., Warren Seering, Eric Rebentisch and Christoph Bauch	"Understanding the Waste Net: A Method for Waste Elimination Prioritization in Product Development" In: Global Perspective for Competitive Enterprise, Economy and Ecology, Ed. Shuo-Yan Chou, Amy Trappey, Jerzy Pokojski and Shana Smith, Springer London, 2009	(Pessôa et al., 2009) **Link**
Kato, Jin	"Development of a Process for Continuous Creation of Lean Value in Product Development Organizations." S.M. Thesis in Mechanical Engineering, Massachusetts Institute of Technology, May 2005.	(Kato, 2005) **Download link**
Graebsch, Martin	"Information and Communication in Lean Product Development." Diploma Thesis, LAI and Technical University of Munich, January 2005.	(Graebsch, 2005) **Download link**
Graebsch, Martin, Warren P. Seering and Udo Lindemann	"Assessing Information Waste in Lean Product Development." 16th International Conference on Engineering Design, Paris, France, August 28 - 30, 2007.	(Graebsch et al., 2007) **Download link**
Bauch, Christoph	"Lean Product Development: Making Waste Transparent" Diploma Thesis, LAI and Technical University of Munich, 2004.	(Bauch, 2004) **Download link**
Hsu, Teng-Cheng	"Causes and Impacts of Class One Engineering Changes: An Exploratory Study Based on Three Defense Aircraft Acquisition Programs" Master Thesis, LAI and Massachusetts Institute of Technology, June 1999	(Hsu, 1999) **Download link**
PD Value and Value Stream Mapping Method		
McManus, Hugh	"Product Development Value Stream Mapping (PDVSM) Manual, Release 1.0," MIT Lean Aerospace Initiative, September 2005	(McManus, 2005) **Download link**
Kato, Jin	see above	
Chase, James P.	"Value Creation in the Product Development Process" Master Thesis, LAI and Massachusetts Institute of Technology, December 2001	(Chase, 2001) **Download link**
Chase, James P.	"Measuring Value in Product Development", LAI Working Paper WP00-05, March 2000	(Chase, 2000) **Download link**
Millard, Richard L.	"Value Stream Analysis and Mapping for Product Development" Master Thesis, LAI and Massachusetts Institute of Technology, June 2001	(Millard, 2001) **Download link**
Slack, Robert A.	"The Application of Lean Principles to the Military Aerospace Product Development", Master Thesis, LAI and Massachusetts Institute of Technology, December 1998	(Slack, 1998) **Download link**
Slack, Robert A.	"The Lean Value Principle in Military Aerospace Product Development", LAI Report Series RP99-01-16, July 1999	(Slack, 1999) **Download link**
Application of PD Value Stream Mapping		
MacKenzie, Scott	"Utilizing Value Stream Mapping In Air Force Acquisition Program Offices." S.M. Thesis, LAI and Massachusetts Institute of Technology, February 2006.	(MacKenzie, 2006) **Download link**
Whitaker, Ryan	"Value Stream Mapping and Earned Value Management: Two Perspectives on Value in Product Development." S.M. Thesis, LAI and Massachusetts Institute of Technology, September 2005	(Whitaker, 2005) **Download link**
Frenkel, Yuliya M.	"Enterprise Level Value Stream Mapping and Analysis for Aircraft Carrier Components". MBA and Master Thesis, LAI and Massachusetts Institute of Technology, June 2004	(Frenkel, 2004) **Download link**
Davis, Mark J.	"Synchronization of System-of-Systems Interfaces in Military Satellite Communications." S. M. Thesis, LAI and Massachusetts Institute of Technology, January 2008.	(Davis, 2008) **Download link**
Carreras, Carmen M.	"Opportunities for Lean Thinking in Aircraft Flight Testing and Evaluation" Master Thesis, LAI and Massachusetts Institute of Technology, June 2002	(Carreras, 2002) **Download link**
Kato, Jin	see above	

6.1 Types of waste and identification

Several studies undertaken by LAI focused on analyzing and describing the different types of waste that can occur in Lean Product Development (see Section 4 above).

The work by Marcus Pessôa (Pessôa, 2008, Pessôa et al., 2008, Pessôa et al., 2009) is based on an extensive review of LAI and non-LAI literature to develop an overview of different types of waste. The descriptions in Section 4 are based on this body of work. His structure also included two more types of waste, 'wishful thinking' and 'external events'. In the overview presented here, these are not treated as types of PD waste as such, but as possible underlying root causes. His work also includes a method to prioritize different types of waste according to the extent by which they influence and are influenced themselves by other types of waste, as well as an approach to identify system-level root causes of wastes in order to prioritize improvement efforts.

The thesis of Jin Kato (Kato, 2005) is based on a very detailed empirical analysis of three projects, conducting a value stream analysis, waste identification and waste quantification. His work therefore belongs to three categories: Defining and identifying different types of waste, extending the method of value stream analysis, and conducting and documenting a very detailed value stream analysis (also see the example in Section 0). In addition to the elements of his work already part of this overview, of special interest are the additional notations that he develops to discern different types of waste graphically in a value stream map, as well as the detailed root cause analyses that he performs for several types of waste.

Martin Graebsch analyses in detail the role of information in product development (Graebsch, 2005, Graebsch et al., 2007). He develops over 100 information and communication-related requirements for displaying a Lean PD value stream. The work analyses waste in communication in more detail and discovers that several types of waste often occur simultaneously (see Figure 13). The most important types of waste were miscommunication, defective information attributes, and inefficient communication (creating structural barriers for example by using email instead of phone).

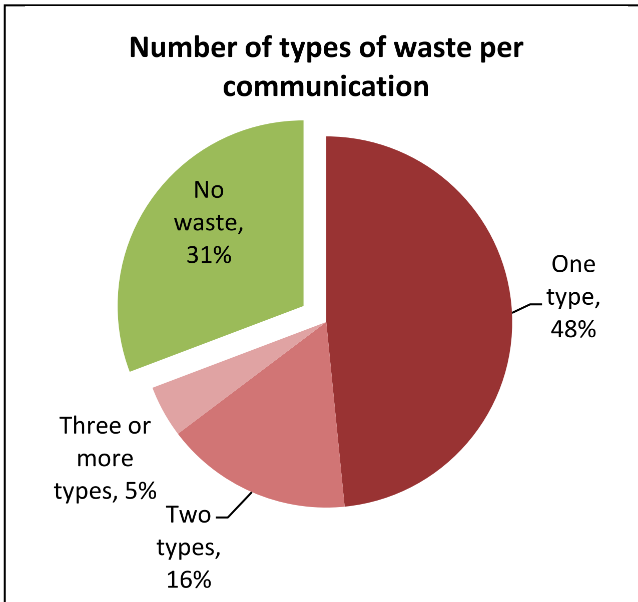


Figure 13: Number of types of waste occurring in one communication

In the work of Christoph Bauch, different types of waste in PD are discussed in detail (Bauch, 2004). Similar to the work of Pessôa presented above, this thesis can also be used as a source for more in-depth description and examples of different types of PD waste. Starting from the fundamental PD goals of performance and quality, time to market, and life cycle cost, general categories of waste are developed: Wasting resources (manpower and equipment), wasting time, wasting information and knowledge, wasting potential, wasting money and wasting people's motivation. These categories are then fueled by different types of waste, as discussed in this document in Section 4 above.

Teng-Cheng Hsu analyzes in his work root causes that lead to a high level of rework, class I engineering changes (Hsu, 1999). This large-scale rework in turn has cost, schedule and potentially also performance impact and impact on organizational processes. The research focused on three defense aircraft acquisition programs. Requirements definition issues, changes in user needs, the need to fix deficiencies, and technological changes were found to be the four dominant causes of engineering changes. The combination of thorough requirements definition facilitated by the use of integrated product teams (IPTs), prioritization on program schedule, and the use of mature technologies allowed one program to make frequent engineering changes to accommodate evolving user needs and changes in technology without any program schedule delay. It was also found that had IPTs been used during the development phases of the other two progams, the prime contractors and their suppliers might have been able to avoid some engineering changes.

6.2 PD value and value stream mapping methods

The work by Hugh McManus (McManus, 2005) provides a detailed guideline to conduct value stream mapping in product development. It is discussed in Section 5.

James Chase (Chase, 2000, Chase, 2001) discusses in his work the value in product development in terms of cost, schedule, performance and risk. It is important to consider processes' value contribution in PD, instead of just only a product's value contribution. It is difficult to judge how much value exactly every activity in PD contributes to the final product. It is therefore sensible to focus the assessment of value creation on the execution of the PD process itself. This leads to the recommendation of strict schedule-focused earned value management in PD. For detailed analysis, the PD process is broken down into the four categories of tasks, resources, environment and management.

Richard Millard analyzed nine different approaches to value stream mapping in the aerospace industry (Millard, 2001). The tools were evaluated regarding their level of sophistication, as well as the business context surrounding the application of the tools. The findings show that tool sophistication correlates positively with the success of a value stream mapping activity. Also, a positive business environment is important, as it encourages and enables the use of more sophisticated tools. A general approach to value stream mapping of the current state, as well as the development of a desired future state, is presented. This approach uses high level Gantt charts, detailed process flow maps and a design structure matrix on the detailed level.

Value in product development is also the subject of the work of Robert Slack (Slack, 1998, Slack, 1999). The meaning of value is investigated first in a general context, in the context of Lean Thinking, and finally in the context of other product development and business literature. This investigation found the value principle to be pertinent in the product development context and a specific definition of value was developed which facilitates an understanding of customer value in the product development arena, and assists in creating a customer focus in the lean transition process (see Figure 14). Although Lean Thinking is customer value focused, the existence of other value perspectives is investigated. The linkage between these value perspectives is also discussed and the need to understand these linkages during the lean transition is established. Finally, specific high level attributes of customer value in the product development setting are established. The work also explores the value stream and flow principles in the context of a jet engine PD case study.

Figure 14: Decomposition of Customer Value in PD

6.3 Applications of PD value stream mapping

A number of LAI publications focus on the application of the value stream mapping technique that then also led to improvements of the method.

PDVSM Application Context	Source
Development of embedded software	(Kato, 2005)
Analysis of PDVSM methods at 9 sites	(Millard, 2001)
F119 Jet Engine Development	(Slack, 1998)
Air Force acquisition system program offices	(MacKenzie, 2006)
Software requirements development (radar system) Hardware prototype development (antenna of radar system)	(Whitaker, 2005)
Aircraft carrier components (enterprise level)	(Frenkel, 2004)
Management of System-of-Systems interfaces in satellite communication systems	(Davis, 2008)

The work by Scott MacKenzie (MacKenzie, 2006) examines the applicability and usefulness of Value Stream Mapping as a tool for program managers in Air Force acquisition System Program Offices. Data were gathered from two Air Force program offices. The key finding was that Value Stream Mapping could potentially be used in a System Program Office with some modifications to the traditional tool. The work being accomplished appears to be a combination of administrative and product development efforts thereby requiring a different type of tool than what has been traditionally used. In addition, the use of Value Stream Mapping is complicated by multiple stakeholders all vying for the program office's resources to fulfill their needs with most of them believing that their needs are more important that the other stakeholders.

Ryan Whitaker (Whitaker, 2005) contrasts in his work two different perspectives on value in the PD process: Value Stream Mapping and Earned Value Management. The activities on the Value Stream Map were analyzed regarding their value contributions, and these findings contrasted with official data from an Earned Value Management system. The two analysis approaches yielded comparable results. Therefore, a hybrid management system, utilizing both Earned Value Management and Value Stream mapping, is developed. Earned Value Management did prove to be very difficult in early project phases; here, a value-stream based approach might be more useful.

The thesis of Yuliya Frenkel (Frenkel, 2004) analyses the value stream of pipe assemblies for an aircraft carrier by creating a global, high level value stream map. The value stream map is used to identify and prioritize the most significant types of waste, as well as explore their root causes. Among other types of waste, the findings helped reduce major time delays, inventory buildups, re-work and excessive processing. The late delivery of piping assemblies was one of the main issues. The PD-related identified root causes include late revisions of engineering drawings, an underestimation of the production duration, as well as late material purchase orders. Among others, the recommended improvements are a re-alignment of critical metrics and prioritization system for rework.

The improvement of System-of-System interfaces in military satellites communication programs is analyzed by Mark Davis (Davis, 2008). Unsynchronized interface design and development caused large amounts of scrap and rework, leading to adverse impacts including large cost growth and schedule delays. The Air Force began to put a framework in place to manage SoS interfaces The research collected data to quantify the performance of the then-current change management process. A value stream mapping and analysis effort along with a discrete event simulation model was conducted to identify areas for improvement in the as-is change management process and suggest an improved future-state change management process. The future-state change management process drew on best practices from the lean and SoS engineering literature to improve interface synchronization and significantly reduce process cycle time. This leaner and more effective future-state change management process could be applicable to many government acquisition program offices to save cost and schedule on programs by reducing the amount of rework due to engineering changes.

7 Acknowledgements

The authors wish to thank John Carroll, Anthony Eastland, John Evers, Paul Fehrenbach, Hugh McManus, Kurt Mittelstaedt, Bohdan Oppenheim and Keith Weiss for their insightful comments and suggestions. The sole responsibility for any errors and inaccuracies remains with the authors. The views expressed in this document are those of the authors alone.

8 Literature

Research publications by LAI (see Table 3) are available on our website at http://lean.mit.edu or can be requested at LAI.

Allen, T. J. (1984) *Managing the Flow of Technology,* Cambridge, MA, MIT Press.

Anand, G. & Kodali, R. (2008) Development of a Conceptual Framework for Lean New Product Development Process. *International Journal of Product Development,* 6**,** 190-224.

Bauch, C. (2004) *Lean Product Development: Making Waste Transparent*, Diploma Thesis. LAI and Technical University of Munich.

Carreras, C. E. (2002) *Opportunities for Lean Thinking in Aircraft Flight Testing and Evaluation*, Master Thesis, LAI and Massachusetts Institute of Technology.

Chase, J. P. (2000) *Measuring Value in Product Development*, LAI Working Paper WP00-05.

Chase, J. P. (2001) *Value Creation in the Product Development Process*, Master Thesis, LAI and Massachusetts Institute of Technology.

Davis, M. J. (2008) *Synchronization of System-of-Systems Interfaces in Military Satellite Communications*, S. M. Thesis, LAI and Massachusetts Institute of Technology.

Evers, J. H., Oehler, G. M. & Tucker, M. G. (1998) Improving Engineering Productivity: A Time Study of an Engineer's Typical Workday. *Proceedings of the IEEE International Conference on Engineering and Technology Management, October 11-13, 1998, San Juan, Puerto Rico***,** 377-381.

Frenkel, Y. M. (2004) *Enterprise Level Value Stream Mapping and Analysis for Aircraft Carrier Components*, MBA and Master Thesis, LAI and Massachusetts Institute of Technology.

Graebsch, M. (2005) *Information and Communication in Lean Product Development*, Diploma Thesis. LAI and Technical University of Munich.

Graebsch, M., Seering, W. & Lindemann, U. (2007) Assessing Information Waste in Lean Product Development. *16th International Conference on Engineering Design, Paris, France, August 28 -30*.

Hsu, T.-C. (1999) *Causes and Impacts of Class One Engineering Changes: An Exploratory Study Based on Three Defense Aircraft Acquisition Programs*, Master Thesis, LAI and Massachusetts Institute of Technology.

Kato, J. (2005) *Development of a Process for Continuous Creation of Lean Value in Product Development Organizations,* Cambridge, MA, LAI / MIT Master Thesis.

MacKenzie, S. (2006) *Utilizing Value Stream Mapping In Air Force Acquisition Program Offices*, S.M. Thesis, LAI and Massachusetts Institute of Technology.

McManus, H. (2005) *Product Development Value Stream Mapping (PDVSM) Manual,* Cambridge, MA, Lean Advancement Initiative (LAI) at MIT.

Millard, R. L. (2001) *Value Stream Analysis and Mapping for Product Development*, Master Thesis, LAI and Massachusetts Institute of Technology.

Morgan, J. M. & Liker, J. K. (2006) *The Toyota Product Development System,* New York, Productivity Press.

Ohno, T. (1998) *Toyoto production system,* New York, Productivity Press.

Oppenheim, B. W. (2004) Lean Product Development Flow. *Systems Engineering,* 7**,** 352-376.

Oppenheim, B. W., Murman, E. M. & Secor, D. A. (2010) Lean Enablers for Systems Engineering. *Systems Engineering,* DOI 10.1002.

Pessôa, M. V. P. (2008) *Weaving the waste net: a model to the product development system low performance drivers and its causes,* Cambridge, MA, LAI White Paper 08-01.

Pessôa, M. V. P., Seering, W. & Rebentisch, E. (2008) Understanding the Waste Net: A Method for Waste Elimination Prioritization in Product Development. *ASME 2008 International Design Engineering Technical Conferences (DETC), New York, NY, August 3-6, 2008*.

Pessôa, M. V. P., Seering, W., Rebentisch, E. & Bauch, C. (2009) Understanding the Waste Net: A Method for Waste Elimination Prioritization in Product Development. IN CHOU, S.-Y., TRAPPEY, A., POKOJSKI, J. & SMITH, S. (Eds.) *Global Perspective for Competitive Enterprise, Economy and Ecology.* London, Springer.

Slack, R. A. (1998) *The Application of Lean Principles to the Military Aerospace Product Development*, Master Thesis, LAI and Massachusetts Institute of Technology.

Slack, R. A. (1999) *The Lean Value Principle in Military Aerospace Product Development*, LAI Report Series, RP99-01-16.

Whitaker, R. (2005) *Value Stream Mapping and Earned Value Management: Two Perspectives on Value in Product Development*, S.M. Thesis, LAI and Massachusetts Institute of Technology.

Womack, J. & Jones, D. (1998) *Lean thinking,* New York, Simon Schuster.