Available online at www.sciencedirect.com
ScienceDirect
International Journal of Project Management 28 (2010) 237–244
International Journal of Project Management
www.elsevier.com/locate/ijproman

Icarus' predicament: Managing the pathologies of overspecification and overdesign

Alex Coman [a], Boaz Ronen [b,*]

[a] Academic College of Tel Aviv-Jaffa, Israel
[b] Tel Aviv University, Faculty of Management, 69978 Tel Aviv, Israel

Received 30 December 2008; received in revised form 21 April 2009; accepted 5 May 2009

Abstract

The phenomenon of overspecification and overdesign is well known in all industries: developing features that are not needed by the customer causes excess development efforts, missed due dates, terminated projects and higher lifecycle costs. The paper defines the phenomena, exploring inherent causes and prescribes solutions for both business-to-business and business-to-customer industries. It presents illustrative cases of overspecification and overdesign, proposes a self-assessment to determine the severity of these phenomena in an organization and resolves the conflicts driving these phenomena. Solutions suggested include adapting Simon's satisficer approach, resolving the marketing conflict by focusing on the 20% of features that account for 80% of the value, breaking the assumption that overspecification is beneficial for future growth potential, resolving the product manager's conflict via a global system view, implementing the 25/25 principle, freezing and stabilizing the specifications, constraining developer time to eliminate spontaneous overdesign, and piecemeal feature launch.

Keywords: Overspecification; Overdesign; The theory-of-constraints (TOC); 25/25; Research and development; Project management; Product management; ARENA

1. Introduction

Ronen and Pass (2008) define the problems of overspecification and overdesign: "Overspecification is defining product or service specifications beyond the actual needs of the customer or the market. Overdesign is designing and developing products or services beyond what is required by the specifications and/or the requirements of the customer or the market". The phenomenon of overspecification usually originates during the interaction of R&D and marketing staff people. Marketing staff members face pressure to bring the product to the market as quickly as possible, when they may not be fully acquainted with the customer or market requirements. Therefore, marketing quite often deliberately defines excessive development requirements in order to *leave all options open*. There is also a hidden understanding that some of the requirements will eventually be down-graded because the full set of marketing requirements is difficult (and sometimes impossible) to attain. There is sometimes also pressure from the customer, and the desire to increase the appeal of the product to a specific customer or to all the market. This broad definition of overspecification includes tight tolerances, unnecessary features and overwhelming complexity. It also includes "artificial complexity" – a term used to describe the needless over-engineered complexity that exists in various systems, products and components.

Overdesign occurs when developers, especially inexperienced ones, design products that are expected to satisfy every imaginable whim of every potential customer. They also have a strong technology drive that leads them to try and develop products that are in the forefront of technology (state-of-the-art). Moreover, in other situations, the developers desire to create options for future extensions of the

* Corresponding author. Tel.: +972 3 644 1181; fax: +972 3 644 1267.
E-mail addresses: Alex@coman.org (A. Coman), Boazr@post.tau.ac.il (B. Ronen).

0263-7863/$36.00
doi:10.1016/j.ijproman.2009.05.001

product (growth potential). As a result, there is the silent conspiracy (a shared interest) by marketing and R&D to introduce overspecification and overdesign into the development of new products. Contrary to the marketing–production conflict where marketing pushes for more features and versions while production pushes in the opposite direction for standardized off-the-shelf products, there are no checks-and-balances in the marketing–R&D interface. Both marketing and R&D push for more features and performance. Moreover, in certain cases the interface between marketing and R&D is fuzzily defined, adding to the overspecification and overdesign pathology.

Tight tolerances of dimensions (e.g. length, width) and performance (e.g. power, gain) are yet another manifestation of the overspecification pathology in R&D. Tight tolerances do not guarantee the delivery of a better product or service. Time pressures on R&D and the difficulty in conforming to excessive tolerances often lead to delays and extra costs. In cases where the delay is excessive, the decision to wave tolerances is taken at a lower level. Thus overspecification may actually lead to underperformance.

The same phenomena of overspecification and overdesign described above for R&D departments also occur in the development of information technology (IT) applications.

Based on our work with dozens of R&D organizations and departments worldwide, more than 25% of development efforts are invested in issues and activities that do not add value, and may be considered garbage time (Ronen and Pass, 2008).

In the marketing literature, Surowiecki (2007) defines the phenomenon of "feature creep", referring to the excessive addition of features resulting in products that are mind-boggling. Rust et al. (2006) found that when consumers were given a choice of three models, of varying complexity, of a digital device, more than 60% chose the one with the most features. Then, when the subjects were given the chance to customize their product, choosing from 25 features, they chose twenty features on average. Rust et al. (2006) found that consumers prefer to purchase feature-loaded offerings. Once they start using their purchase, the feature overload prevents them from effectively operating the functions they really need. They then return the item, and take their business elsewhere. Overspecification and overdesign result in costly returns, lost return sales, and detraction.

den Ouden (2006) found that Americans who returned a product that was too complicated for them had spent, on average, just 20 min with it before giving up. Lu et al. (2007) found that at least half of returned products had nothing wrong with them. Consumers just could not figure out how to use them.

Lu et al. (2007) assert that "for highly innovative products the actual product use is often very uncertain". They emphasize that "Under the time-to-market pressure, it is increasingly important to take into consideration the significant factors that determine product use in the early product development process".

Section 2 of the paper presents real-life cases where overspecification or overdesign destroyed value. Section 3 analyzes the causes for the overspecification and overdesign phenomena and presents the conflicts driving them. Section 4 presents solutions for the overspecification and overdesign problem. Section 5 presents a procedure to diagnose the extent to which an organization is afflicted with overspecification and overdesign. Section 6 concludes the paper and calls for further research.

2. Illustrative overspecification and overdesign cases

The following examples illustrate value destruction as a result of overspecification and overdesign. During our combined engineering, research, teaching, management and consulting experience of over 70 years we have encountered the following examples to illustrate the phenomenon:

Company A – a NASDAQ traded developer of WiMax telecom solutions, won a $42 million contract from a Japanese telephone service provider – to deploy Japan's first WiMAX network across Tokyo. The deployment of the first-of-a-kind system which offered wireless telephone and internet connection to residential and business customers ran into difficulties due to low reliability. Post-mortem analysis revealed that the cause was a feature developed for future applications. This feature was not needed to complete the $42 million contract. The feature was not operational at the time but caused the system to crash frequently. Removal of the unneeded feature solved the problem. This phenomenon whereby a feature that is planted in the product for hypothetical future applications aborts short-term functionality, leading to product termination and in some cases to the company's closure, was observed in several other cases.

Company E – a cellular handset manufacturer came up with a concept that was futuristic for its time: a cellular phone with game platform and multimedia console – music and video. The design was so innovative that the project kept delaying; the product was not delivered to customers, eventually leading to a crisis situation. The decision reached in order to salvage the project was to remove the multimedia features. The result was the critically delayed launch of a mediocre, me-too, product. To make things worse, the product's platform was exorbitantly expensive since it was designed to support a highly demanding performance envelope. It had a powerful multimedia processor, large memory capacity and a beefed up power package to drive them. The resulting product was a market failure. This pathology of "too little; too late; too costly" is manifested in three stages: (1) ambitious overspecification and overdesign of a "killer" product; (2) development is acutely delayed, the product reaches a crisis management stage; (3) the product's features are mercilessly tapered, and the product is finally launched. The launched product's features are unimpressive (much energy was spent on features that did not make it to the release) – too little; it is a "me-too" product – too late; and its platform (processor, power supply, etc.) is expensive since it was designed to support its ambitious (non existent) features – too costly.

Microsoft – in its efforts to enter the software security market Microsoft acquired an innovative startup company that specialized in data security. In the first stages of the product's specification the company went through an intensive brain-storming process, the result of which was a trailblazing design. The design did not just protect from threats but among a broad list of features also assured that illegal software was not run on the computer. The product which was aimed at penetrating the data security market was significantly delayed. A crisis management session cut out many of the features, launching a mediocre, me-too product.

Microsoft Word and Excel provide yet another bizarre example of unneeded features. Microsoft Word has a hidden pinball application and Microsoft Excel has a flight simulator stealthily incorporated into the product. Not only do these applications contribute nothing to Microsoft's value; they result in wasted developer effort, distraction, increased memory resource demand and radical increase in the product's complexity.

Company P – a cellular service provider initiated a strategic program to open "store-within-a-store" – distribution centres within drugstores. These were sales kiosks opened inside retailers such as drugstores, home products chains, etc. The legal department was in charge of closing the contract with the retailers. Time and again, it took longer to close the overspecified contract than it took to physically install the kiosk. The result was that the legal contracts were finalized after the business was already functional de-facto.

Bradley tank (Burton, 1993) – the Bradley tank was originally developed as a troop carrier to transport eleven soldiers to the battlefield. The development process exceeded 20 years and added to the Bradley a myriad of other functions such as: a missile carrier, amphibious features, weapons systems and more. As a result the Bradley turned out to be more than a tank.

Gutenberg (Britannica, 2009) – the inventor of metal movable print had a mission – to emulate the writing of contemporary scribes. In his effort to reach perfection and print in several colours, Gutenberg's project required more and more financing to complete. Gutenberg borrowed money from a lawyer, Johann Fust, making him a business partner. Gutenberg's perfectionism – the desire to improve the quality and extra features such as being able to print in colour, resulted in bankruptcy. Fust and his associate took over the business and printed the first fine books without Gutenberg's overspecified features.

Legal & General (2009) – a mortgage financial service provider, required applicants to fill out an overspecified "full life application" form which took 2 h to complete. After reducing the form to a simpler, straightforward application, Legal & General witnessed a 40% growth in applications submitted, a 13% increase in immediate acceptance applications, a 9% reduction in applications not being processed, and an overall growth in profitability.

Software overspecification – from the authors' experience most software applications developed in-house severely suffer from overspecification and overdesign. Unneeded or nice-to-have features are added to the product to be on the safe side or for future growth. Functionality is added with no economic analysis or justification. The end result is an excessively complex product, severe project overrun in terms of time and money and frequent project termination with no product delivery at all.

Consumer goods – manufacturers of consumer products continuously add new features. The result of this overspecification and overdesign is a severe reduction in product usability. By cramming in features that are seldom used, users have difficulty to focus on frequently used features that they need. This is common with audio and video remote controls that have over 50 buttons. The activation of common features such as volume control and station search becomes exorbitant. Washing machine manufacturers similarly add special programs and features that make them unusable to normal people who use one or two programs.

Table 1 summarizes the case studies. It illustrates the various pathologies, from delayed launch through excessive complexity to loss of the entire company.

We have observed that products can be classified according to their feature density. We identify three feature density zones (Fig. 1): the inferior product zone, the effective product zone, and the overspec product zone. In a given competitive arena, products that do not meet minimal customer requirements are considered inferior. They are not competitive and hence are value destroyers. Effective products are products that satisfy important customer requirements. Overspec products, cramming features that do not deliver value, destroy value by increasing costs, reducing throughput and delaying product launch. The overspec condition occurs when executives get carried away with the belief that "you can't have too much of a good thing". By contrast, the veteran product manager is familiar with the maxim: "a perfect product is the enemy of a good product".

3. Sources for overspecification and overdesign

Better is notoriously the enemy of good. Why then do marketers and engineers keep falling into this trap time and again? Overspecification and overdesign stem from the characteristics of human behaviour and from organizational measurement and compensation:

1. *Optimizer approach*: in many cases the root-cause for both overspecification and overdesign lies in the phenomenon defined by Nobel laureate Herbert A. Simon as the optimizer approach (Simon, 1957; Ronen and Pass, 2008). Simon revolutionized management by identifying a managerial phenomenon which causes decision-making failures. He claimed that executives, engineers, and decision makers strive to be optimizers, that is, to achieve the best possible solution, without consideration of time constraints.

Table 1
Case studies and implications.

Case	Industry	Pathology
Company A	New product development	Future feature kills project
Company E	Consumer goods	Late launch due to overspecification results in a mediocre and overpriced product
Microsoft	Software development	Extreme overspecification and resulting crisis launch an outdated antivirus and protection product
Microsoft	Software development	Excessive complexity as a result of overspecification prevents timely updates and stabilization
Company P	Retail services	Perfectionist legal process is completed later than actual business transaction
Bradley tank	Defence	Product looses focus, diverges from original purpose and lasts for years
Gutenberg	Invention/new product development	Overspecification results in bankruptcy and loss of intellectual property
Legal & General	Mortgage bank	Simplified application form adds value
Software applications	Software development	"Growth potential" with no economic justification
Consumer goods	Consumer goods	Feature overload damages usability

2. *Option overkill*: for both overspecification and overdesign the tendency of marketing and engineering to anticipate the product's growth potential has been observed (Rust et al., 2006). Features are crammed into the product to assure compliance with potential future demand. All features that can be conceived along the product's lifecycle are incorporated in its first version.
3. *One size fits all*: the product is developed to comply with requirements from radically different customer segments. When a standard, off-the-shelf product is designed, rather than developing distinct product versions, the result is expensive and highly complex. The authors encountered a hi-tech company that developed a universal power-supply component. The product was designed to accept a broad range of input power sources and deliver a broad range of power output, aiming to simplify the development, logistics and testing of the product. The result was an overly complex product which could not receive certification-of-compliance. Moreover, it carried a significantly higher price tag than single-purpose power supplies and was therefore scrapped.
4. *Lack of knowledge and leadership*: marketing people do not know with certitude which features will generate market demand and will differentiate the product in the customer's eyes. Similarly, engineering people do not know with certainty which standards, protocols, components and features will dominate the future. As a result both marketing and R&D people spread their bets across all slots in the roulette wheel. Many marketing and R&D executives do not have the leadership to determine which features should be included in the product's initial release, which should be deferred to future releases along the product's roadmap, and which features should be eliminated altogether.

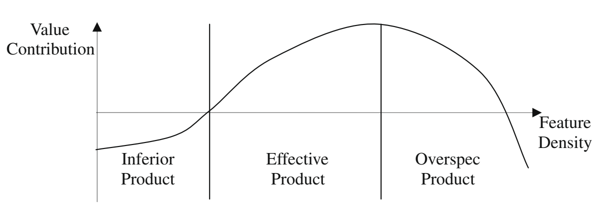

Fig. 1. Feature density zones.

5. *Measurement, incentives and compensation*: marketing and R&D executives should be measured by the value contribution of the product or project along its lifecycle. Unfortunately in many cases marketing people are measured by their creativity and therefore concentrate on dreaming up as many potential segments and features as possible. They are measured by the product's media exposure and at best the short-run acceptance of the product. They therefore often tend to concentrate on exotic applications. R&D people are measured by the traditional performance trinity: scope, cost and time. Of these three elements, scope is most tangible during the initial product specification. Hence, R&D people initially incorporate as many features as they can possibly imagine. As the product's launch is delayed, the product enters crisis-management mode and features are eliminated to shorten time-to-market. The features that are eliminated are often important, value - creating features that were delayed due to the waste of resources on marginal features that were not eliminated early enough (den Ouden, 2006).
6. *Organizational culture*: engineering schools train engineers to deliver the "best" product from a technological perspective. Only a minority of schools stress design-to-cost and teach product lifecycle principles; the majority of young engineers do not see their objective as increasing the company's value through their product or project. Engineers' self-esteem and peer-appreciation are derived from technological brilliance rather than value delivery breakthrough (Rust et al., 2006).The same applies to marketing people. Their goal is to be more creative and expose the product to the media. Their culture does not involve value creation. Since both R&D and marketing people have motivation for overspecification and overdesign, a

"conspiracy" is forged between the "customer" and the "supplier". This is true for both solicited projects – where a specific customer orders the product and for unsolicited projects – where the product is conceived by the marketing department.

7. *Grey R&D*: every R&D department is characterized by a certain amount of "grey R&D" – unauthorized projects or features that are developed by highly motivated R&D people. The R&D department is a "permanent bottleneck" (Ronen and Pass, 2008) – i.e. demand for their services permanently exceeds supply. The authors' experience shows that the demand for software and hardware applications and products is three to five times greater than the resources available. Thus, the formal strategic gating process eliminates low priority projects. In certain cases developers excited by new technologies and infrastructure develop these features in a covert mode – under management radar, with no control or supervision. This results in product overdesign.
8. *Manipulative budgeting*: new technologies, infrastructure, platforms and feature introduction should normally be part of the R&D budget. However, in certain cases the bulk of the budget is assigned to customer-specific projects. Thus, when R&D people wish to pursue new technologies they manipulate projects to use these technologies, even when they are not needed from the customer's point of view. This results in features that are unneeded for the project.
9. *Inertia*: this pathology was defined by Christensen (2003) as the "Innovator's Dilemma". Christensen describes innovators locked on the improvement of a single performance measure when it no longer makes a difference to the customer. Engineers and computer scientists perpetually wish to improve performance. As a result they release improved products on a continuous basis. In some cases inertia causes the engineers to pursue performance improvement even when the improvement cost exceeds the value to the customer. This phenomenon is sometimes intensified by competitive market conditions. Examples include the race to improve a processor's clock rate between Intel and AMD. Intel eventually realized that they had reached overspecification and moved to another more important parameter – namely power consumption.
10. *The misconception of the linearity of effort*: common human thought is linear, leading to the belief that the addition of each new feature results in a proportional increase in effort. However, complexity added to the system results in an exponential increase in effort. New features complicate the product's architecture, compete for common constrained resources and cause a multitude of unanticipated quality problems. As a result developers miscalculate the effort associated with added features underestimating its impact on the project. Thus, for over 50 years Eli Lilly locked itself into an effort to manufacture purer and purer insulin. The company's efforts eventually produced Humulin – 100% perfectly pure human insulin, only to discover that the purity differentiation was insignificant since only a fraction of people develop insulin resistance. Thus highly purified pork insulin is good enough for the majority of the population. At this level the race for purity becomes insignificant.

We apply Goldratt's conflict-resolution-diagram (Goldratt, 1991) in the analysis of the underlying conflicts leading marketing and R&D to commit the overspecification and overdesign (Ronen and Pass, 2008) mistake.

Fig. 2 describes the marketing organizational conflict underlying overspecification.

Box A designates the undisputed goal to increase the company's value. From the marketer's point of view the conflict is between satisfying the needs of more customer segments thus increasing throughput (box B) vs. reducing operating expenses and complexity (box C). The conflict is evident between satisfying more customers via a multitude of features (box D) or reducing expenses and complexity via few features (box D′).

Fig. 3 describes the R&D organizational conflict underlying overdesign.

This goal can be achieved by trying to gain long-term benefits (box B) or by completing the project on time (box C). The actual conflict is presented in box D vs. box D′: Develop with overspec and overdesign (box D) vs. Develop just-in-spec (box D′).

Fig. 4 describes the underlying personal conflict for overspecification and overdesign experienced by R&D professionals.

As before, box A designates the developer's individual goal which is to increase self-accomplishment. This goal can be achieved by being on the edge of technology and personal satisfaction; (box B) or by completing the project on time (box C). The actual conflict is presented in box D

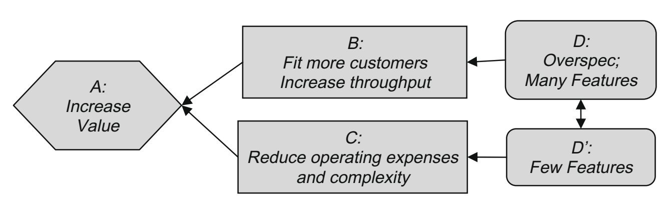

Fig. 2. The marketing-overspecification conflict.

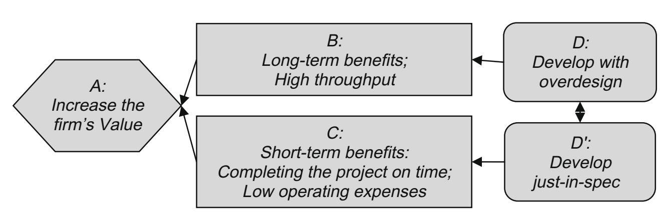

Fig. 3. The R&D-overdesign organizational conflict.

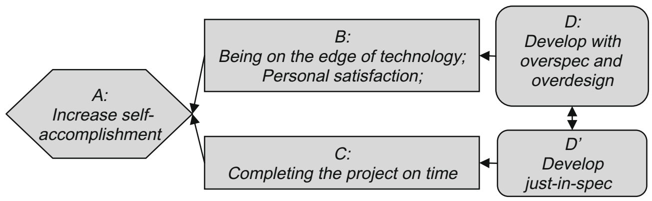

Fig. 4. The personal conflict for overspecification and overdesign.

vs. box D′: Develop with overspec and overdesign (box D) vs. Develop just-in-spec (box D′).

4. Solutions for overspecification and overdesign

a. The "optimizer" approach, a major source for overspecification and overdesign is remedied by Simon's (Simon, 1957; Ronen and Pass, 2008) "satisficer" approach. The satisficer sets a "level-of-aspiration", a threshold he or she aspires to achieve. The objective is no longer to maximize or minimize some performance measure, but to achieve a solution that will improve the measure beyond the predefined level-of-aspiration. The satisficer need not exhaustively examine all possible alternatives. The satisficer examines some alternatives until one that satisfies the level-of-aspiration is found. Once the level-of-aspiration has been met, the satisficer may set a new level-of-aspiration. This iterative process delivers attainable, continuous improvement. The authors' experience shows that whenever the satisficer concept is spread within a company, it resolves a significant part of the problem. The satisficer will not seek "the best solution". Rather, he or she will look for a practical solution that will reach a certain "level-of-aspiration" that will represent the customer's actual needs.

b. The overspecification-marketing conflict can be resolved via the differentiation principle. The assumption that overspecified products increase throughput is often erroneous. Differentiation applying the Pareto principle (20% of the features account for 80% of the value) resolves the majority of conflicts. Eighty percent of the features can be developed just-in-spec and only 20% of the features need be incorporated in the product platform.

c. The overdesign-R&D conflict can be resolved via assumption breaking (Ronen and Pass, 2008). The assumption of growth potential is often over-optimistic and naïve. Technology changes rapidly while the product lifecycle becomes shorter. Therefore significantly fewer design features are smoothly recycled into next generation products. The organizational conflict should be resolved at the senior executive level – which features should be included on a product platform level, which should be postponed for later releases along the product's roadmap and which should be eliminated altogether. This decision is often resolved de-facto by the engineering level.

d. The personal conflict for overspecification and overdesign of the product management and development person can be relaxed through globalization (Ronen and Pass, 2008). A global view of the organizational project portfolio and the R&D human resource personalities can relax this conflict. The desire to be at the cutting edge of the technological knowledge and to achieve personal satisfaction can be met by occasionally assigning these development people to innovative high-risk projects. This enables them to occasionally face cutting edge technology and fulfil their technological interest. However, low-risk more-of-the-same projects or components within projects should follow just-in-spec principles. The conflict may also be resolved by challenging the developers with the economic need of design-to-cost.

e. 25/25 principle: The 25/25 (Ronen and Pass, 2006) principle states that management should periodically terminate 25% of the projects/products and taper (trim down) 25% of the features in the projects/products that are not terminated. While the company's innovation process is responsible for the steady addition of products and features, no organizational function is charged with the evaluation of products and projects and the removal of "white elephants". This results in the proliferation of projects and products and generates a "high-mix low-volume" product portfolio. During the product's specification phase the tendency of both marketing and development is to brainstorm as many features as possible. This "conspiracy" between marketing and R&D results in overspecification and later in overdesign. The 25/25 mechanism establishes checks-and-balances for this tendency. The 25/25 is chaired by senior business executives and operationalized by marketing, sales, R&D and operations people. For a specific project, trimming down the unneeded features, as a part of a 25/25 process can reduce overspecification and overdesign dramatically.

f. Freeze and stabilize: Product managers tend to leave as many options open as possible throughout the project's lifecycle – to respond to new ideas and events in the competitive arena. This results in unmanageable feature creep. Two milestones must be established: freeze – the point in time after which no changes to the product specification are accepted from marketing, and stabilize – the point in time after which only fixes are accepted from R&D. For low uncertainty projects the project is frozen at the project launch. For high uncertainty projects the product may be frozen at a later phase determined by senior management. Breakthrough projects are often frozen at a very late phase.

g. Controlling the developer: In order to reduce the engineers' tendency for overdesign, management should construct a straight-jacket – a gating mechanism containing the engineering feature explosion. Such mechanisms include design-to-launch and critical-chain buffer management (Goldratt, 1997). Typically, projects are managed in a design-to-spec priority. This results in project delays, which in-turn cause further features to be added in response to competitive moves in the arena. The design-to-launch mechanism subordinates the features to the pre-defined launch date, constraining the engineers' ability to add unplanned features. Within the project's timetable, critical-chain buffer management further constrains feature creep. Rather than assigning spare time to individual activities controlled by engineers, critical-chain clusters all spare time resources in a project buffer managed by the senior project manager. Individual engineers are allocated tight activity durations precluding them from making unauthorized feature additions.

h. Piecemeal feature launch: Today's dynamic business arena causes multiple feature value and effort assessment mistakes. False-positive mistakes apply to features that were perceived valuable in the specification phase but proved to be redundant in the implementation phase. False-negative mistakes apply to features that were classified as white elephants during the specification phase but proved to be valuable later in the product's lifecycle. R&D people are often detached from the real user in the field and are as a result prone to make both types of mistakes. Piecemeal feature launch enables the timely release of features while retaining the real-option to remedy gating mistakes. False-positive features are eliminated as soon as their real value is ascertained. False-negative features are reinstated as soon as the mistake is identified. This technique is particularly applicable for software features. The methodology of software-as-a-service (e.g. Google or Salesforce.com) enables gradual and granular increments of the product or the service's features. The extreme-programming and scrum methodologies prescribe minute feature launches and frequent feature re-evaluation.

i. Using the quality-function-deployment (QFD) methodology (Yoji, 2004; Chan and Wu, 2005): QFD should be used to prioritize investments in product features. The QFD methodology starts with the customer's quality functions – i.e. the set of features that define the product's quality in the customer's eyes. These features are weighed based on their importance to the customer. The quality function is deployed across the various organizational departments that are accountable for defining the quality function. These departments include development, design, purchasing, manufacturing, quality, logistics, customer support, etc. Next the product is compared with its competitors to prioritize feature value creation. The QFD methodology is incorporated into systems and software engineering lifecycle standard ISO 15288 (Chan and Wu, 2005), and into quality management standard ISO 10006 (ISO 10006, 2003).

5. Self-assessment

The following questions are used by senior executives to assess the severity of overspecification and overdesign in their organization. The questionnaire indicates whether overspecification and overdesign are severe pathologies in the organization.

Consider the following questions:

a. Are most product milestones or projects delivered on time? If "yes" then overspecification and overdesign are not significant pathologies in your organization. If "no" then consider the following questions:
b. Are most of your products excessively ahead of your competition?
c. What percentage of effort invested in new features is targeted for long-term potential?
d. What percentage of effort invested is designated for potential new customers?
e. What percentage of effort invested is for new, potential product applications?
f. What percentage of development effort exceeds specified requirements?
g. What is the proportion of overspecification in your project?
h. What is the proportion of overdesign in your project?
i. What is the extent of excessively tight tolerances in your project?
j. What is the extent of artificial complexity?

6. Conclusions

The pathology of overspecification and overdesign is critical for organizations dealing with frequent new product introduction. Though very important, little has been published in the academic literature on this topic. This paper defines the problems of overspecification and overdesign, demonstrates their occurrence, investigates their roots and causes, and suggests practical solutions. Finally, the paper proposes a simple methodology to diagnose the severity of the phenomena in a given organization. Further research should follow this paper: empirical studies should quantify the amount of effort wasted on overspecification and overdesign in various industries and arenas. Next, quantitative modelling studies should evaluate the dimensions that determine these pathologies and their interactions. Case studies that illustrate the phenomena should follow.

References

Britannica, 2009. <http://www.britannica.com/EBchecked/topic/249878/Johannes-Gutenberg>. March 2.

Burton, J.G., 1993. The Pentagon Wars: Reformers Challenge the Old Guard. Naval Institute Press, Annapolis, MD.

Chan, L.K., Wu, M.L., 2005. A systematic approach to quality function deployment with a full illustrative example. Omega 33 (2), 119–139 (April).

Christensen, C.M., 2003. Innovator's Dilemma. HarperCollins Publishers, NY.

den Ouden, E., 2006. Development of a design analysis model for consumer complaints: revealing a new class of quality failures. Doctoral Dissertation. Technische Universiteit Eindhoven, The Netherlands.

Goldratt, E.M., 1991. It's not Luck. North River Press, Great Barrington, MA, USA.

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

ISO 10006, 2003. Quality Management Systems – Guidelines for Quality Management in Projects, Geneva, Switzerland.

ISO/IEC 15288, 2008. Systems and Software Engineering – System Life Cycle Processes, International, Geneva, Switzerland.

Legal & General, 2009. <http://www.legalandgeneralgroup.com/investors/presentations.cfm?year=2006>. March 2.

Lu, Y., den Ouden, E., Brombacher, A.C., Geudens, W.H.J.M., Hartmann, H., 2007. Towards a more systematic analysis of uncertain user-product interactions in product development: an enhanced user-product interaction framework. Qual. Reliab. Eng. Int. 23 (1), 19–29.

Ronen, B., Pass, S., 2006. The 25–25 principle. In: The Fourth Worldwide TOCICO Conference, 4–7 November, Miami, Florida, USA.

Ronen, B., Pass, S., 2008. Focused Operations Management: Doing More with Existing Resources. John Wiley and Sons, New York, NY, USA.

Rust, R.T., Thompson, D.V., Hamilton, R.W., 2006. Defeating feature fatigue. Harvard Bus. Rev. (February), 98–107.

Simon, H.A., 1957. Models of Man. John Wiley and Sons, New York, NY, USA.

Surowiecki, J., 2007. Feature presentation. The New Yorker, (May 28), p. 8.

Yoji, Akao, 2004. Quality Function Deployment: Integrating Customer Requirements into Product Design. Taylor and Francis, London.