DOWNLOAD PDF SET BASED CONCURRENT ENGINEERING

Chapter 1 : Concurrent Engineering/Design Process - Wikibooks, open books for an open world

How Toyota's product design and development process helps find the best solutions and develop successful products.

Mix of qualitative and quantitative criteria Fusion of multiple team member evaluations Determining what to do next to ensure the best possible decision is being made Overall, applying Bayesian models to trade studies clarifies decisions and provides direction. ARL Trade Space Visualizer ATSV [edit] Farzaneh Farhangmehr In the early stages of design, one of the main goals of decision makers is to generate trade studies of possible design options that can meet design requirements and select the best one. In traditional multi-objective optimization techniques, decision-makers have to quantify their preferences a priori. To aim this goal, a Pareto Frontier of non-dominated solutions has been defined so that decision makers can evaluate preferences. However, sometimes decision makers cannot successfully determine their preferences a priori. The goal of ATSV is generally to provide a technique for populating a multi-objective trade space using differential evolution [11] based on the reduced in aspect of dimension subset of the objectives. As a result, instead of a single solution, ATSV provides unconstrained dimensions and it allows decision makers to analyze interactions of design variables. Limitations and Benefits of ARL Trade Space Visualizer ATSV [edit] Design by shopping which was introduced by Rick Balling [12] in enables a posteriori articulation of preferences by allowing decision makers to view a variety of feasible designs and then selecting a preference after visualizing the trade space for an optimal design based on this preference. ATSV by multi-dimensional data visualization Glyph plots, Histogram plots, Parallel coordinates, Scatter matrices, Brushing, Linked views displays multiple plots simultaneously and interactively applies preferences. So, instead of a single solution, ATSV provides unconstrained dimensions and it allows decision makers to analyze interactions of design variables. Furthermore, the automation used in ATSV brings an implementation to shopping paradigm by analyzing a large number of designs in a short period of time. In spite of these benefits, as a new technique, ATSV needs to be matured in some areas. One of this area is to develope faster graphical interfaces; understand the impact of problem size and complexity on user performance and limit the size of the problem based on results of theses findings. Finally, like most of available design methods, ATSV fails with respect to provide techniques of visualizing risk and uncertainties associated with systems. Collaborative Engineering[edit] Chris Fagan Intelligent Negotiation Mechanism states that because different disciplines have different goals and knowledge, it is unavoidable that conflict will happen in the design process. Many multidisciplinary design projects require engineers to work on designs simultaneously, which can lead to confusion due to gaps in communication among engineers. Thus, it is important to design a work flow to keep the design process from being delayed. Similar to shopping for a design in ATSV, it was found that a visual representation for the management of the project could be beneficial. This first culminated in Gantt charts; a basic planning tool. These Gantt charts were only able to display information inputted from a user. As a result, this output was only as good as the information put in and if there were changes midstream the program may or may not be able to accurately predict a change. Concurrent Simultaneous Engineering Resource View ConSERV is a knowledge-based project and was built with the idea that there is a relationship between design and project management. The ConSERV software serves as a decision support system, providing schedule reminders and keeping track of progress. This allows the project manager to oversee all aspects of a multidisciplinary project with ease. Identify the project and parameters 2. Identify the main risk elements 3. Identify the most appropriate management tools 4. Establish the team and the project execution plan 5. Apply the ConSERV concept Although ConSERV is an entire software package that applies all these design strategies simultaneously, the concepts can be applied with traditional project management software and organization. PDM is a software of some type that controls and tracks data for different designs. This can save time in money in companies where data can sometimes be lost when no PDM is used. One such software tool is Team Center. Team Center allows people from all of the world to be working on one system design while keeping all the data on a central server. It also acts as a overseer for sensitive files such as CAD files, allowing only certain users to modify the part and preventing any lost data. More on PDM can be read at the wiki here: It is a language that enables unambiguous communication of geometric imperfection specifications and measurement data for mechanical parts and assemblies. The "Datum Features" that are identified as part of the process for establishing Datum Reference Frames are generally the features that mate with other parts in an assembly. Ambiguity will be encountered when attempting to control orientation or location of feaures with directly toleranced dimensions. Ambiguity will also be encountered regarding definition of directly toleranced size dimensions. A mechanical design is not complete until achievable and unambiguous specification of geometric imperfection of part features has been developed and applied to the drawings 2D or 3D that define the part s in the assembly. Lacking complete specification may lead to subsequent design changes that will be needed to accommodate the imperfection associated with physical parts. Tolerance analysis methods and also dimensional metrology methods can become quite involved for critical or complex features. A general "Profile of a Surface" tolerance is applied via a general note on the drawing, then the most critical features receive specifically applied tolerances in the field of the drawing. This method enables more focus upon the most critical functional concerns. Similar risk management cannot be achieved with only directly toleranced dimensions, due to the great ambiguity that would be encountered. This can include design for manufacturing, safety, performance, marketing, environment, cost and flexibility. DFM is usually a principle that concentrates on reducing the cost of part production. Designing for manufacturability should start at the beginning of product development and should be improved upon throughout the design process. Engineering should work with manufacturing and other functional groups to design parts that can be easily manufactured. Parts that are designed for manufacturability and assembly generally cost less to produce than parts that are not designed with these considerations. Designing spent nuclear fuel transport vessels is a perfect example of designing for safety. The transport vessels for spent nuclear fuel have to be extremely rigid and have to be able to take a massive amount of punishment. Both of these tests are to show the safety of the transport vessel. In general, designing for safety does not consider cost as one of the design parameters because it often costs a lot of money to make a part or system safe. There are many aspects of DFP that are different from other areas of design. In this case, the design is all about getting the most performance out of the product as possible. Oftentimes, little care is given to cost and manufacturability. Some examples of where this applies is the racing industry automotive and otherwise , space missions, and some military projects such as fighter jets. In order to get the best performance out of a design it is important to model and simulate the product before it is built. The modeling and simulation can be a complex process, but can save time and money later in the design process, as well as produce the best performance out of the design. Through simulation combined with DoE, trends can be found between design variables that can be used to achieve maximum performance. Complex system design is one idea that can be applied well to DFP. By looking at the product as a whole, complex interactions between large systems can be found. Blake Giles The following diagram depicts a typical design process geared toward the design of high performance complex systems such as airplanes and space vehicles. Design for Performance flow Step 1: Define the overall goal of the mission and its objectives through qualification. This statement should be referred to throughout the design cycle to ensure that the mission needs are being met. Quantify how well the mission objectives should be met to allow for success. These should be high level performance metrics of system attributes. For example, in the design of an automobile, these metrics would be acceleration, cornering, fuel economy, etc. These metrics are certainly subject to change throughout the design cycle and should be revisited. Define and characterize concepts that will meet mission objectives. This brainstorming activity should enumerate the possibilities of several different concepts that could potentially lead to mission success. Define alternate mission elements or mission architectures to meet the requirements of the mission concept. Architectures are the high level descriptions of the physical systems and sub-systems that carry out functions to accomplish the mission concepts. Identify the principle cost and performance drivers. These are the architectures which have a relatively high impact on system cost and performance. By identifying these drivers early will allow the design team to balance performance and cost. Characterization of mission architectures means to define the sub-systems of the vehicle including weight, power, and cost. Here mathematical models can be applied to describe sub-system performance. Evaluate quantitative requirements and identify critical requirements Step 8: Quantify how well requirements and broad objectives are being met relative to cost and architectural choices. A baseline design is a single consistent definition of the system which meets most or all of the mission objectives. A consistent system definition is a single set of values for all of the system parameters which fit together. Translate broad objectives and constraints into well-defined system requirements. Translate system requirements into component level requirements. Design for Marketing[edit] Adam Aschenbach Design for marketing is a way to design parts and systems so that they will be marketable. Some products may have great functionality but they may never sell in large quantities because they are not marketable. An example of design for marketing was presented in class. When Professor Burke worked at HP, marketing came to the engineers and said that they wanted the print cartridges to fit into beveled card board boxes. Although these boxes may have been slightly harder to produce and assemble they were used as a way to differentiate HP print cartridges from all their competitors. This made it easy for customers to tell which cartridges they needed for their HP printers and the boxes had more appeal due to their unique shape. PPP[edit] 3P is a product and process design tool or methodology. Derived from lean manufacturing, 3P is a "Design For Manufacturability" approach in which large consideration of the product or process design is targeted at reducing waste in the manufacturing process. Where continual improvement, also derived from lean, aims to iteratively improve the manufacturing process in small low impact increments over time, 3P allows engineers to redesign a process from scratch and seek better performing solutions. The type of overhaul is higher performing but requires more resources and capital. The goal of a 3P team is to meet customer requirements while using the least amount of resources and realizing a quick time to implementation. A cross-functional team is selected to represent multiple aspects of the product or design. In a multi-day process, the team will review customer requirements and brainstorm several solution possibilities.

DOWNLOAD PDF SET BASED CONCURRENT ENGINEERING

Chapter 2 : Concurrent engineering - Wikipedia

Toyota (and their supplier Denso) are the classical examples of set-based concurrent engineering and they did it without any software tools. They used paper-based tools such as their engineering checksheets filled with hand-drawn limit and trade-off curves and other design knowledge.

Introduction[edit] A publication described concurrent engineering as a new design management system that has matured in recent years to become a well-defined systems approach to optimizing design and engineering cycles. Beginning in the early s, CE was also adapted for use in the information and content automation field, providing a basis for organization and management of projects outside the physical product development sector for which it was originally designed. The basic premise for concurrent engineering revolves around two concepts. The idea is that the concurrent nature of these activities significantly increases productivity and product quality. By locating and fixing these issues early, the design team can avoid what often become costly errors as the project moves to more complicated computational models and eventually into the actual manufacturing of hardware. This includes establishing user requirements, propagating early conceptual designs, running computational models, creating physical prototypes, and eventually manufacturing the product. Included in this process is taking into full account funding, workforce capability, and time requirements. A study claimed that a correct implementation of the concurrent design process can save a significant amount of money, and that organizations have been moving to concurrent design for this reason. Concurrent engineering replaces the more traditional sequential design flow, or "Waterfall Model". In this design system, a design team would not quickly look backward or forward from the step it is on to fix or anticipate problems. In the case that something does go wrong, the design usually must be scrapped or heavily altered. The concurrent or iterative design process encourages prompt changes of tack, so that all aspects of the life cycle of the product are taken into account, allowing for a more evolutionary approach to design. Traditional "Waterfall" or Sequential Development Method vs. Iterative Development Method in concurrent engineering. A significant part of the concurrent design method is that the individual engineer is given much more say in the overall design process due to the collaborative nature of concurrent engineering. Giving the designer ownership is claimed to improve the productivity of the employee and quality of the product, based on the assumption that people who are given a sense of gratification and ownership over their work tend to work harder and design a more robust product, as opposed to an employee that is assigned a task with little say in the general process. If such issues are not addressed properly, concurrent design may not work effectively. Service providers exist that specialize in this field, not only training people how to perform concurrent design effectively, but also providing the tools to enhance the communication between the team members. Cross-functional teams[edit] Cross-functional teams include people from different area of the workplace that are all involved in a particular process, including manufacturing, hardware and software design, marketing, and so forth. Concurrent product realization[edit] Doing several things at once, such as designing various subsystems simultaneously, is critical to reducing design time and is at the heart of concurrent engineering. Incremental information sharing[edit] Incremental information sharing helps minimize the chance that concurrent product realization will lead to surprises. Cross-functional teams are important to the effective sharing of information in a timely fashion. Integrated project management[edit] Integrated project management ensures that someone is responsible for the entire project, and that responsibility is not abdicated once one aspect of the work is done.

Chapter 3 : Set Based Design For Lean Engineering | Alopex on Innovation

Set-Based Concurrent Engineering An approach to the design of products and services in which developers consider sets of ideas rather than single ideas. To do this, developers.

Book Reading the first edition of the book really helped me understand Lean Product Development, which led me to reading all other materials available on the subject. Despite my search far and wide, this book remains the number one book on my list by a wide margin. The second edition does absolute wonders in resolving some of the problems with the fist edition. The addition of case studies really helps in convincing the reader that it really works in the real world. The case on Scania was revealing especially on the very seldom revealed aspects of modular design. The Ford case shows a huge transformation of a gigantic organization. The Ping story reveals details on the power of causal maps and tradeoff curves in developing extremely competitive products. And the Menlo case shows the finest example of Lean concepts applied to the software industry. There have been significant additions in the main text as well. I am delighted to see more concrete examples of causal diagrams and tradeoff curves. All in all I consider the second edition as an entirely new book so those who have the first edition would benefit hugely by buying and reading the second edition. A word of caution though. This book is not only long, it has a very high idea density. This book condenses the content of 3 books into one. You have to reflect after reading every chapter to let the new and often counter-intuitive ideas sink in. This book is great for group reading and discussion. I would recommend that readers try to read this during the course of a few months to really absorb all the ideas. Being a Japanese, I am amazed that an American has been able to put the arcane practices of Toyota product developers so well. I have spoken to several Toyota development managers who seems not to understand that they are doing anything unique. They consider it common sense. And yet most Japanese companies outside the auto industry are not practicing Lean Product Development. I hope that publication of this book in Japanese will help rectify the problem. Book The second edition is well worth the investment of your time to read, even if you have read the first edition. Many people think that Lean is primarily for production operations, but the same Lean thinking can be applied effectively to knowledge work such as product development. In an era when rapid innovation is key to survival in many industries, this book is a must-read. Book Great update of the original work by al ward. Thanks for the great book!

Chapter 4 : Set-Based concurrent Engineering | calendrierdelascience.com

Set-Based Concurrent Engineering is a product development approach which offers an environment that not only permits but encourages radical innovation, increased learning and reuse of knowledge, reduces the development risk, and enable shorter and less costly development cycles.

In each of these cases, we spend time identifying and exploring the alternatives, then strive for a clear, succinct way to present the tradeoffs involved. These summaries can help you narrow in. For example, we might decide early on that we need sorting. A later decision might highlight that stable sorting is important. This would cut down the number of algorithm options. This approach, of exploring, recording, and narrowing alternatives, feels closest in spirit to the Toyota product-development approach. Consciously enable alternatives A second approach is to use modularity: The book Design Rules, Volume 1: The Power of Modularity , by Baldwin and Clark, explores modularity from a non-software-specific point of view; Design Patterns Gamma et al. Many examples of modularity are based around the ideas of abstraction and of an interface: For example, two systems communicate via SOAP, with no commitment about what technology or tool will be used on either side. Bridge driver design pattern. Each of these patterns represents a "move" that is consciously leaving room for a set of alternative designs. You could "build the persistence layer," mapping out all objects from all the tables you already have. This creates a "point-based" solution â€" this is the way persistence works. Each of those objects will have to be tested, refactored, and maintained, against that possible day in the future when all that work may finally pay off. Instead, an incremental design might commit us a little now, but it leaves to the future the set of all possible ways to implement the objects based on the current structure. And if the whole approach becomes rejected, we only pay to change objects that were already needed. I think of a simple design as set-based in this sense: Consider the program we have, doing just enough to support the current set of features. Some aspects of this program are a commitment to a particular design. But there is a set of programs out there â€" all those programs that extend the current program in a way compatible with the current design. By contrast, if we fully elaborate an implementation to support our guesses about future directions, we in effect are betting on a point solution, not a set of possible evolutions. I think there is another strategy that also fits the billâ€"duplicate implementation. This approach goes further than the first one â€" rather than evaluate the alternatives and make a guess, we carry forward on multiple implementations, thus seeing the real impact of our decisions. Conclusion There are several different design approaches that yield aspects of a set-based design:

Chapter 5 : Moving from Point-Based to Set-Based | Success AssuredÂ® | Targeted Convergence Corpora

Set-Based Concurrent Engineering Posted on December 20, by Bill Wake One of the ideas in lean product development is the notion of set-based concurrent engineering: considering a solution as the intersection of a number of feasible parts, rather than iterating on a bunch of individual "point-based" solutions.

Its development teams are not colocated. Personnel, with the exception of the chief engineer and his staff, are not dedicated to one vehicle program. Engineering and test functions rarely use quality function deployment QFD and Taguchi methods. The second paradox can be summarized in this way: Toyota considers a broader range of possible designs and delays certain decisions longer than other automotive companies do, yet has what may be the fastest and most efficient vehicle development cycles in the industry. Traditional design practice, whether concurrent or not, tends to quickly converge on a solution, a point in the solution space, and then modify that solution until it meets the design objectives. This seems an effective approach unless one picks the wrong starting point; subsequent iterations to refine that solution can be very time consuming and lead to a suboptimal design. A wide net from the start, and gradual elimination of weaker solutions, makes finding the best or better solutions more likely. As a result, Toyota may take more time early on to define the solutions, but can then move more quickly toward convergence and, ultimately, production than its point-based counterparts. We present the conceptual framework of SBCE in more detail, tying it in with other characteristics of the Toyota development system, and discuss why the SBCE principles lead to highly effective product development systems. Does Toyota really do what we claimed? Many of the challenges focused on the more extreme examples we offered. We said, for example, that Toyota broadly explored body styles and could consider anywhere from five to twenty different styling alternatives. And we suggested that final styling decisions could wait as long as the second full-vehicle prototype, at the extreme. These extreme cases were intended to be just that â€”extremes to demonstrate a point, not averages. More important than the specific numbers were the underlying principles of design that Toyota followed. We chose these examples to illustrate ideas, not to suggest that if a company makes lots of prototypes or waits until the very last minute to make decisions, its development process will improve. In fact, a good job exploring solutions on one project can lead to a very focused search and much more rapid convergence on a design in later projects. Both the novelty of the idea and the skepticism we encountered led us to develop the paradigm of SBCE further. We began by collecting more data. He interviewed managers and engineers from a broad range of design specialties including styling, body engineering, chassis engineering, power train engineering, vehicle evaluation, production engineering, and prototyping, and a number of closely affiliated Toyota suppliers. In addition, we interviewed Japanese and U. Bringing its development system to the United States has forced Toyota to make its design philosophy and principles explicit. The training materials and process for U. Research Updates from Get semi-monthly updates on how global companies are managing in a changing world. How is Toyota able to do concurrent engineering so well? Traditional, serial engineering is a series of functions, each designing to a single solution or point see top of Figure 1. Of course, this is a simplification; there are feedback loops, but the feedback from downstream functions comes later, often after upstream functions have committed to a particular solution. And, typically, the feedback consists of specific critiques that lead to minor changes to the base design. Serial engineering is fraught with shortcomings due to the delayed feedback loops. As usually practiced, CE attempts to bring more feedback upstream earlier, generally through face-to-face meetings. The typical CE process looks something like the lower part of Figure 1. A function such as styling comes up with a design solution and very early in the process shows it to other functions for input. These downstream functions analyze and critique the design from their perspective. For example, the top members of a Chrysler design team meet for an entire day every week. Since this is done early, changes to the styling design are relatively easy and inexpensive, and ideally, the design team soon arrives at a solution that will satisfy all parties. While an improvement over serial engineering, the basic picture remains the same: As the design passes from group to group for critique from different functional perspectives or even if they are critiquing it as a cross-functional team , every change causes further changes and analysis, resulting in rework and additional communication demands. There is no theoretical guarantee that the process will ever converge, and hundreds of engineers have told us that it often does not: Since the development organization never gets a clear picture of the possibilities, the resulting design can be far from optimal. Despite these drawbacks, many companies have been successful with iterative, apparently point-based models. Combining this very fast iteration with modular product architectures and extremely skilled programmers enables Microsoft to remain a leader in the software industry. Similarly, Terwiesch et al. Design participants reason about, develop, and communicate sets of solutions in parallel and relatively independently. As designs converge, participants commit to staying within the set s , barring extreme circumstances, so that others can rely on their communication. In Part A, the two functions, design engineering and manufacturing engineering, define broad sets of feasible solutions from their respective areas of expertise principle 1 â€" map the design space. In Part B, design engineering then smoothly refines the set over time by eliminating ideas not feasible from the manufacturing perspective principle 2 â€" integrate by intersection. Design engineering continues to refine the set through further design and development work, while manufacturing engineering is also designing and refining at this stage. In Part C, the two groups continue to communicate about the sets under consideration, ensuring producible product designs while enabling manufacturing to get a jump start on design and fabrication of the production process principle 3 â€" establish feasibility before commitment. The gradual convergence to a final design, Part D, helps the development team make sound design decisions at each stage. Gradual convergence also allows both functions to work in tandem with little risk of rework. Figure 2 is highly simplified, with only two actors. SBCE works in the context of many actors defining sets, communicating sets, and converging to mutually acceptable solutions that optimize system performance, not individual subsystem performance. Figure 2 Example of Set-Based Concurrent Engineering SBCE assumes that reasoning and communicating about sets of ideas leads to more robust, optimized systems and greater overall efficiency than working with one idea at a time, even though the individual steps may look inefficient. In practice, the costs of eliminating all back-tracking could probably not be justified. But a focus on convergence, rather than on tweaking a good idea to optimize it, can dramatically reduce the amount of back-tracking in the process. The following examples demonstrate a range of approaches explored in detail later that are all consistent with the underlying philosophy: Toyota maintains at least two full-scale models in parallel typically from two studios , while most competitors pick one styling design, create one full-scale clay model, and go immediately to detailed design. Simultaneous with the development of the two to three full-scale models, Toyota engineers develop structural plans for multiple styling design ideas and analyze them for manufacturability. By the time a vehicle program reaches the die-making stage, U. Toyota and other Japanese automakers , though, still views specifications as targets for die makers to refine. Manufacturing engineers then set the tolerances based on their understanding of current manufacturing capabilities. Fit and appearance to the customer override concern for exactly matching specifications. The resulting dies, then, define the final specifications for the vehicle, not the CAD database. This example illustrates a belief that a nominal dimension, which appears to be a fixed, single point, really implies a range of acceptable solutions. Die makers have developed a tacit understanding of the range allowed in the design passed on from product engineering. For every major part of the car, the engineers responsible for that part develop, maintain, and update an engineering checklist, which represents current capabilities â€" the set of feasible designs. Product engineers and production engineers also maintain checklists. When a product engineer begins a design, the production engineer sends the latest checklist so the product engineer knows the current constraints on the solutions space. This example illustrates how organizational memory can be facilitated by mapping the feasible solution space. Taking time up front to explore and document feasible solutions from design and manufacturing perspectives leads to tremendous gains in efficiency and product integration later in the process and for subsequent development cycles. All three examples involve reasoning about sets of alternatives and a sophisticated understanding of the boundaries on the solution space. Later, we describe the underlying principles of SBCE in greater depth, along with additional detailed examples from Toyota automotive development. Related Research Students of design and creativity have traditionally emphasized looking at many ideas. This method, Pugh claims, can apply to any phase of the design process, not just to concept selection. Iansiti describes a product development team at NEC that carried four distinct product concepts in parallel and worked for two years on design and development to arrive at a final concept. For example, many of the authors mentioned above seem to assume that a colocated team looks at the sets together, allowing informal communication. At Toyota, communication about sets is explicit. These authors also imply that the sets are discrete lists of alternatives, ignoring other ways of representing sets. A large body of literature looks at the evolution of fundamental science and engineering innovations at the macro-level over time. Technology cycles are defined by dominant designs and subsequent technological discontinuities. Product development of mature technologies involves integration of detailed design decisions about thousands of parts and interrelated subsystems. Toyota excels at this integration by keeping options open longer, communicating about sets, and breaking free of some of the cognitive constraints described by Nelson and Winter. SBCE is a critical aspect of the system, but it operates in concert with other, equally important principles on system design and the use of knowledge. For example, Toyota develops deep technical expertise in both its engineering and management ranks. Managers are excellent, experienced engineers who continue to view technical engineering as at least the second most critical aspect of their jobs the most critical may be developing the engineers they supervise. Outside their small staff, they have no direct authority over functional engineers who report to functional general managers. However, CEs are totally responsible for their vehicles, from the early concept stages through launch and into the initial marketing campaign. They perform vital systems integration, for while each function is responsible for its subsystem, the chief engineers are responsible for the total vehicles. The CEs make the set-based process work by controlling the narrowing process, insisting on broad exploration, resolving any disagreements across functions, and, when needed, making decisions on competing alternatives based on an analysis of trade-offs. Principles of Set-Based Concurrent Engineering In our earlier article, we described many examples of SBCE but had not yet systematized these practices into an overall framework. Together, the principles create a framework in which design participants can work on pieces of the design in parallel yet knit them together into a system. The remainder of the article discusses these principles in detail. Principles of Set-Based Concurrent Engineering 1. Map the design space. Explore trade-offs by designing multiple alternatives. Communicate sets of possibilities. Look for intersections of feasible sets. Establish feasibility before commitment.

Chapter 6 : Concurrent Engineering is best done Set-Based | Success AssuredÂ® | Targeted Convergence

The Set-Based Concurrent Engineering (SBCE) is the methodology that can improve the efficiencies and effectiveness of product development. It is found that the SBCE approach provided a suitable knowledge environment to support decision making throughout the development process.

Chapter 7 : Concurrent Product Development - Tools & Techniques

Concurrent Engineering is best done Set-Based Since the 's, concurrent engineering has played an important role in early design work of larger organizations. However, the power of concurrent engineering is tremendously limited by point-based development.

Chapter 8 : Set-Based Concurrent Engineering - XP

Concurrent engineering (CE) is a work methodology emphasizing the parallelisation of tasks (i.e. performing tasks concurrently), which is sometimes called simultaneous engineering or integrated product development (IPD) using an integrated product team approach.

Chapter 9 : Lean Product and Process Development, 2nd Edition

What is Set-Based Design? ABSTRACT team-based concurrent engineering approach, with notable successes in the *automotive (Chrysler Viper, Ford Mustang) and.*