TechniCom, Inc.
66 Mt. Prospect Avenue
Clifton, NJ 07013 USA
(973) 470 9110
http://www.technicom.com
staff@technicom.com

NX Systems Engineering (from UGS PLM Solutions) Powers the Product Lifecycle

December 2003

Prepared by
TechniCom, Inc.

NX Systems Engineering Powers the Product Lifecycle

Requirements and specifications drive the design

Introduction

In early October, 2003 we traveled to Cincinnati, OH to meet with several technical experts from UGS PLM Solutions. Our objective was to find out more about the capabilities of NX 2 to perform systems engineering tasks in mechanical design projects. This paper will review how Systems Engineering can begin to be implemented by using the current tools available within the NX product development environment. We hope to be able to point out its importance to users, and what engineers might achieve using such technology.

Real designers usually admit that CAD is used more for documenting the design than developing it. We need to rethink the design process now that we have all these exotic computer based mechanical design tools. We believe that Systems Engineering is the next step in product design. (See the box for a description)

> Systems Engineering uses a combination of theories and tools, along with a methodology and procedures, to address real world problems that are often of large scale and scope. Systems engineering activities vary from requirements definition or specification to the conceptual and functional development of systems. The systems engineer's perspective is different from that of a product engineer. Whereas the product engineer deals with details, the systems engineer takes a "top down" perspective dealing with details only as needed to guarantee successful implementation. Whereas the product engineer deals with system internals, the systems engineer also addresses the external view of the system through the system's interface to other systems, users, and managers.

Today's technology limits

Today's CAD systems are quite facile at making geometric changes driven by parameters. But assemblies typically connect using only geometric mating constraints. If these mating constraints change, how is the corresponding sub-system to react? Suppose two co-linear shafts connect via a coupling. If one shaft increases in diameter, how is the coupling or shaft supposed to change? Since no knowledge of the behavior of the coupling is known, a human designer is required to intervene.

Advanced product development systems, after more than a decade of working on embedding knowledge rules, are still in their infancy.

Solving interdependencies by embedding each connection with knowledge rules could solve the problem. BUT such rules only work for predetermined changes! And a rule would have to be written for every connection. Clearly an impossibility. On the other hand, knowing the characteristics of the coupling might enable the system to compute such a change. Yet mechanical systems have no way to describe what happens at each interface. Several problems exist in making such a concept operable.

1. No language exists to clearly define what subsystems do. Even in the simple case of a bolt, we know it has thickness, thread pitch and depth, and material type. But what does it do? How does its performance change as its parameters and characteristics change?
2. Beside geometry, changes include operational characteristics, geometric sizing and tolerances, external stresses and reactions, motion behavior, and on and on. These changes are quite complex to describe, and in fact, often impossible to describe with a regular language. Many systems can only be described using non-linear charts, such as RPM vs. power output.

Nevertheless, there are CAD systems that enable a substantial subset of the components necessary to provide systems engineering tools. Leading among them is NX from UGS PLM Solutions.

The NX Product Line

While the NX product line consists of many products, the ones we are particularly interested in for this paper relate to those that enable systems engineering. These include the following:

NX solid modeling - available in Designer bundles that form the basis for mechanical product designs.

Teamcenter Engineering - for managing complex parts, assemblies, revisions, and engineering work orders.

Teamcenter Requirements - groupware for defining, parsing, and managing all important system requirements.

WAVE - for top down design of interpart relationships within assemblies. Included with

NX modeling. Interpart dependencies managed with a high level control structure.

WAVE Diagramming - added module providing diagrammatic management of WAVE links.

Knowledge Fusion Author - provides a rules based modeling and knowledge based engineering development kit

Knowledge Fusion sharing - allows editing of engineering models created with Knowledge Fusion. For a modest fee this can be added to every seat of NX.

Digital Simulation - includes the wide breadth of analysis tools directly coupled to the CAD models.

How NX assists in Systems Engineering

Unique among its peers, UGS PLM Solutions is aggressively tying together its solutions, starting from the beginning of the design process and working toward a capability that offers unique tools to help customers solve these problems. Systems Engineering concepts will work more viably when tied directly into the CAD tool. Teamcenter Requirements provides one of the best ways we have seen to make specific product requirements visible and offers a way to interpret these requirements so that they can directly drive geometry. Extending, then, directly into top down design by controlling key assembly constraints allows building and investigating overall designs early.

If detailed designs allow their parts and assemblies to be "knowledgeable" then such top down designs can be moved rapidly to detailed design.

No matter the level of detail, tightly integrated NX analysis offerings allow careful evaluation of design alternatives, and whether and how the product meets the initial requirements.

Closing the loop between the design requirements will prove if the design meets the requirements!

Customer benefits

Volvo Aero wanted to speed its development process and improve handling of design changes. The company implemented NX Knowledge Fusion(KF), which allows embedding rules within a design, because of its tight integration with feature-based CAD. The results included shorter lead times, more accurate bids, more design iterations (four per day instead of one per week), and a dramatic decrease in the time required to complete design changes (many went from a month to hours).

The Timken Company, using KF, captured design rules from a variety of sources. Their bearing design application runs as an integral part of NX and their engineers can generate geometry from scratch or manipulate existing CAD models. Using Teamcenter to manage their bearing design environment, with captured knowledge stored in Teamcenter Engineering, this system automates the "grunt work" of repetitive calculations and table look-ups, and ensures that design rules aren't violated. New designs now take hours, as compared to days, can have more iterations and analyses per design, and have better product consistency.

At a recent conference GE Aircraft Engines stated that using KF techniques and WAVE for top down sketches driven by spreadsheets, they were able to morph one engine size to another, yielding savings in time of 3-6 months, the ability to perform 20 times the number of analysis iterations in 25% of the time, and a 50% savings in manufacturing costs.

All these implementations produced excellent payback! These are just the tip of the iceberg because customers like those cited above have only begun to realize the results of systems engineered design.

Executive Summary

Systems Engineering addresses the effective design of reliable systems within cost and time constraints. Companies that employ Systems Engineering techniques are the ones that will best be able to understand the effect of and reaction to changes anywhere in the product lifecycle.

The next challenge facing manufacturing companies is to progress from cost cutting to increasing revenue and margin through innovative product development. Often such innovative designs require rethinking the product delivery processes. Simulating the essence of these processes allows product innovations to be proven before deployment. Companies that employ Systems Engineering techniques to understand and control product lifecycles will be the ones most likely to succeed in gaining market share through innovative product development.

We strongly recommend that all users engaging in product development seriously consider how they approach the task. Considering an approach that employs systems engineering is a logical, well thought out alternative to the bottoms up thinking that has prevailed in the computer aided design industry for years. In the past we have been unable to wed the tools necessary to perform such a system approach. Only today have we begun to see the ability to use this approach. Users who adopt these techniques will be in a unique position in the future to develop, deliver, and service products. They will be able to respond to rapidly changing conditions in markets, the supply chain, and manufacturing that traditional companies will not. UGS PLM Solutions' NX enables the use of Systems Engineering concepts. Customers need to make the decision to change their business practices to accommo-

date a systems engineering, product lifecycle planning approach. The time is now for such decisions!

About the Author

Raymond Kurland is President of TechniCom, Inc., a market research and analysis organization that specializes in understanding, consulting, and writing about Mechanical Engineering product development software. TechniCom produces a monthly newsletter, specialized reports, and offers a continuing research program for software vendors. Ray also speaks at conferences on the subject and frequently consults with users considering embarking on re-evaluating their product development systems. He can be reached via email at rayk@technicom.com.

More about NX tools and how they enable Systems Engineering

NX's "digital decision making" includes capturing product, engineering and performance knowledge, allowing its re-use early to validate and optimize design decisions. This includes total product engineering, system based modeling, knowledge driven automation, integrated digital simulation, and via Teamcenter, the ability to manage and control the critical product data.

The NX components we are most interested in and that make up the systems engineering capability of NX 2 include: WAVE, requirements capture management, Knowledge Fusion (often abbreviated as KF) for part design and high level design, and simulation.

Below we discuss each of these individually and then discuss how they can be used together.

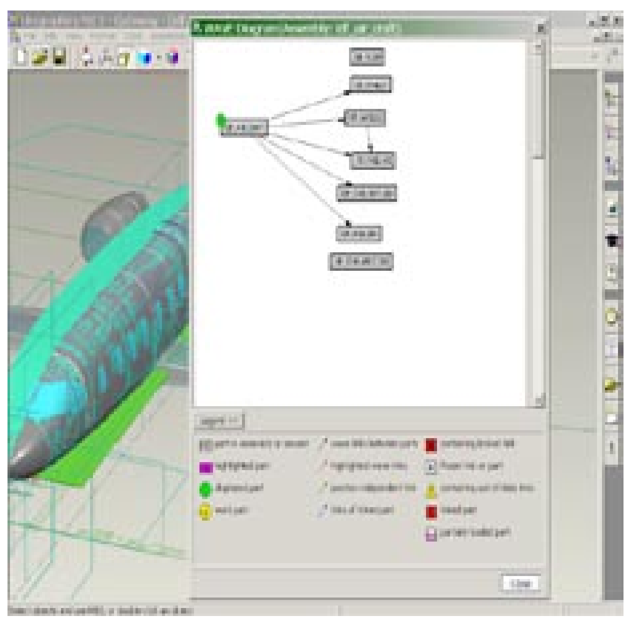
1. NX WAVE enables top down design and helps manage complex changes

An integrated module within the NX portfolio, WAVE allows control of interpart modeling by elevating interpart dependencies to a high level control structure. Complex parametric systems are often unable to be successfully modified because of interpart relations buried deep within part and assembly structures. The problem is that most systems have no way of understanding how changing the values of these relationships affect the final model. Even worse, when such models fail to execute after changing these values, the reason for the failure may be difficult to unravel, much less repair. The dichotomy is that such relationships are necessary to build the intrinsic knowledge of how the geometry is related. Consequently, users often find it difficult to provide libraries of reusable objects, a critical success factor for building a knowledge base.

WAVE allows the logical development and management of interpart geometric dependencies. It does this by using control structures, which in themselves are a simple representation depicting relationships. When using a high level control structure, the individual component parts, in their definition, refer back to the control structure for their parameters. Thus, modifying the control structure drives changes down to the individual components. Different than some other systems, such interpart modeling uses geometric entities rather than complex expressions, and thus provides a more general solution.

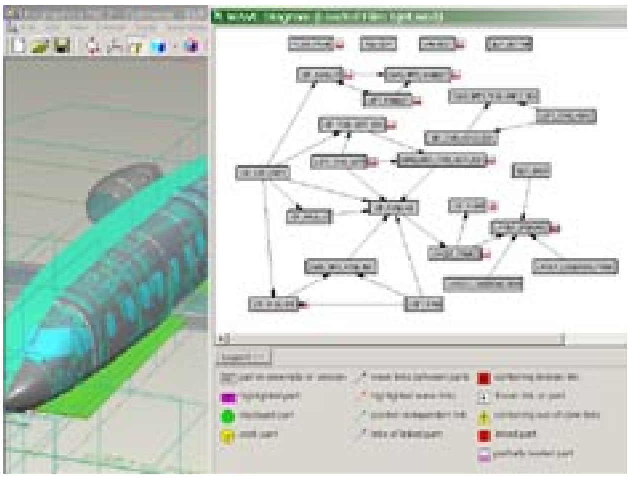
An additional module, called WAVE control, offers other capabilities that allow top level structure management such as wave diagramming, link mapping, and link searching. Re-parenting (moving objects elsewhere in an assembly) is part of WAVE and provides the powerful ability to edit links between parts to resolve any conflicts.

Teamcenter Engineering adds to WAVE's flexibility by allowing users to manage these links and control structures more easily.



The image above depicts a simple first-level WAVE diagram for a small business jet. Only a portion of the control diagram is displayed. Note the clarity of the controlling parameters and their notation.

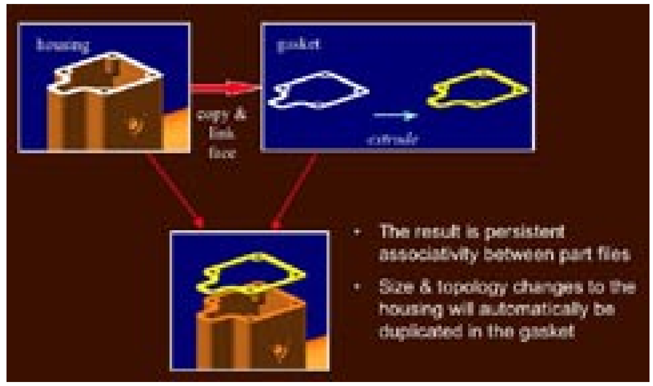
UGS PLM Solutions recommends, and we wholeheartedly agree, that prospective customers plan in advance as how best to use (WAVE) control structures in a top down implementation. Users planning such an implementation (and we cannot think of any who should not do so) should seek out the advice of UGS PLM Solutions for its recommended best practices before proceeding. Their experience should greatly improve the transition to top down design - the only way projects are effectively designed!



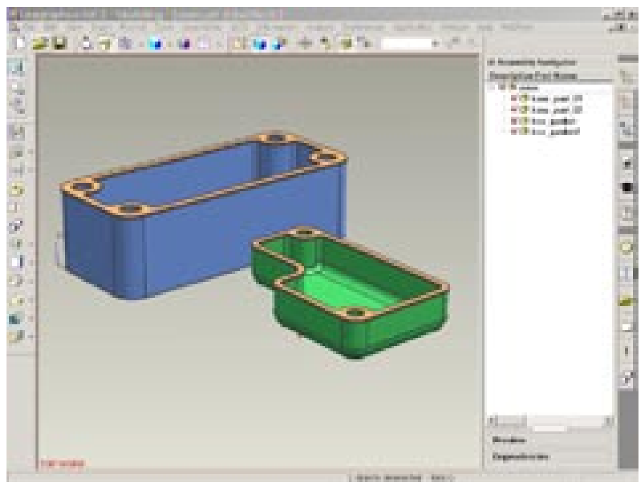
The image above displays a WAVE diagram that expands the detail of the previous image; it can show as much or as little detail as desired. This diagram shows two levels.

A WAVE example illustrates top down design

To illustrate how geometric links are used within WAVE to control dependencies we looked at a simple example - adding a gasket to a base part. Note in the image below an extruded block with four holes and a pocket. The objective is to build a gasket related to the top face. We did so by copying the top face to become the base of the gasket by selecting the face and choosing "link to work part," where the work part was a newly named blank part named "gasket." Simple, yet powerful, this technique allows the system to continue to operate even if the parent is unavailable. When available it checks and updates the parent link. Since the linked geometry is copied, complicated history models are not required - merely the link.



As can be seen from the image above, modifying the base part by changing the face causes the gasket to automatically update. In the event of errors in regenerating the part, we can choose to accept these changes or alter the base part, to arrive at the correct final part. *We liked this flexibility.* A generic template can be developed by using a linked part and simply changing the original link to point to a new part, as we have done for the second, green part gasket below.



The image above shows completed box assemblies, each with their intelligent gaskets built by linking the gasket part to different faces.

2. NX Knowledge Fusion enables system configuration within the CAD context

Historically, implementations of knowledge capture software have been used "upstream" from the product development (CAD) system, because the communication from the knowledge system to the design system was strictly a serial, one-way path. While knowledge capture systems were a powerful engine for configuring complex systems, their geometry creation tools were limited. Traditionally, a knowledge system would create a low-fidelity framework upon which detailed design work could be layered. In more dynamic design environments, the serial nature of the connection between the knowledge system and the design system became a major liability. Once downstream design work began on a design, changes or additions to the requirements or knowledge base could no longer drive design changes in an automatic way. Indeed, incorporation of such changes would require "starting over" at the beginning of the requirement-driven first stage – with regeneration from scratch of the knowledge system output. Thus, while knowledge systems were powerful engines for high-level configuration and evaluation of requirements, and CAD systems were powerful tools for the creation of complex detailed geometry, the lack of a two-way interaction between these two worlds was a major problem. UGS PLM Solutions fused together the knowledge and CAD worlds in a uniquely powerful way.

A knowledge-based engineering (KBE) rules evaluation engine embedded directly inside NX eliminates the traditional wall between knowledge systems and detailed design systems. UGS PLM Solutions calls this combination NX Knowledge Fusion (KF). First introduced a few versions ago, this technology has matured to an impressive level, and now provides a rich capability with an enormous potential for productivity. Engineers can now not only drive designs "up-front" using external requirements and knowledge databases, but also encapsulate stimuli/results sequences by embedding knowledge rules inside existing designs – or do both simultaneously.

This very point highlights the departure of the UGS PLM Solutions' knowledge tools from the historical norm. Traditional knowledge-based engineering usually concerned itself with "generative" application of knowledge – the creation "from scratch" of geometry to drive designs. Knowledge Fusion still supports this generative approach, and several customers have created fully generative knowledge-based applications using KF. However, the patent-pending concept of "adoption" (see example 1 below) allows knowledge rules and connections to requirements to be added to existing designs, without starting over, and then <u>continue</u> to influence the design throughout its lifecycle.

Incorporation of knowledge in system designs can proceed in at least five major ways:

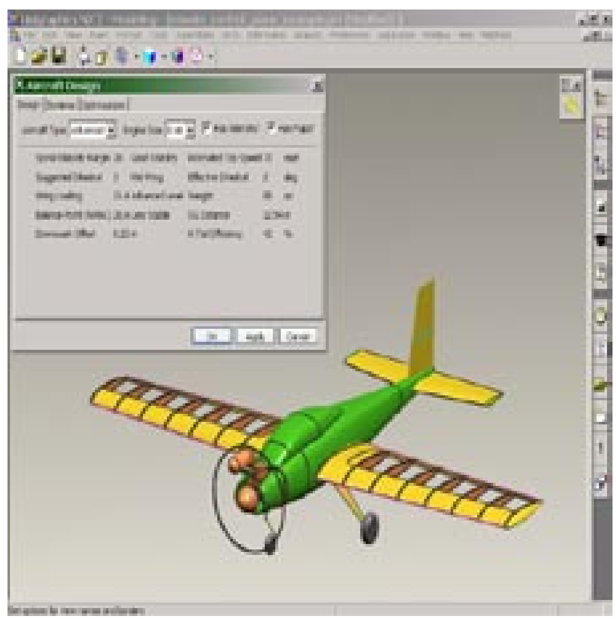
Encapsulation using existing subsystems: For a systems engineer, KF provides a mechanism for encapsulating sub-systems and defining interfaces such that a system can be configured and driven based on any combination of geometric or non-geometric requirements. Whereas traditional design systems were primarily geometry-driven, KF opens the door to non-geometric inputs, outputs, and relationships between subsystems. Knowledge Fusion can wrap subsystem interfaces around the major subsystems of their product, as a starting point to develop system configurations, while preserving investments in existing design data.

Developing configurations: Once subsystem interfaces have been adequately described, KF can be used as the engine to drive configurations of the system. Even those where major topological changes occur – a capability not available using traditional CAD-based methods.

Analysis results can drive KF decisions: Successful systems require analyzing results. Knowledge Fusion allows performing novel, non-traditional types of system evaluation and other analyses not commercially available elsewhere. A rich set of tools allows KF to incorporate and link designs to FEA, kinematic/dynamic analysis, and other external analysis codes.

Designs can be checked for compliance: KF-based tools like NX Check-Mate (a real-time system for monitoring corporate design standards compliance) help enforce compliance with design standards and warn users when designs start to deviate from acceptable standards or level of performance. (See example 2 below)

Design alternatives and processes can be captured: Because it is an integral part of every single seat of NX, KF allows ready interaction with internal data structures and interaction methods within NX. The KF toolset, coupled with the User Interface Styler in NX, allows building easy to use knowledge-driven design automation applications that both simplify user input and guide them through the process.

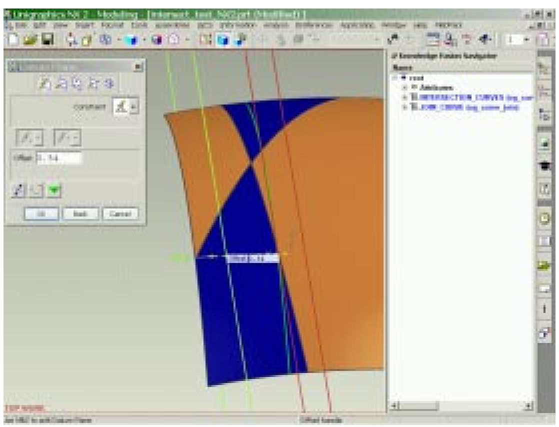


The above image shows a custom designed user dialog built with NX Knowledge Fusion tools to improve usage of a pre-defined KF function for "model airplane design."

Many customers have built unique knowledge-driven design applications using the KF toolset. UGS PLM used exactly these KF tools and techniques to develop most of the new modules that debuted with the release of NX and NX 2, including Mold Wizard, Optimization Wizard, Strength Wizard, Progressive Die Wizard, a suite of automotive body design wizards, and Check-Mate.

KF Example 1 - Changing the process flow for geometry evaluation allows users to extend NX

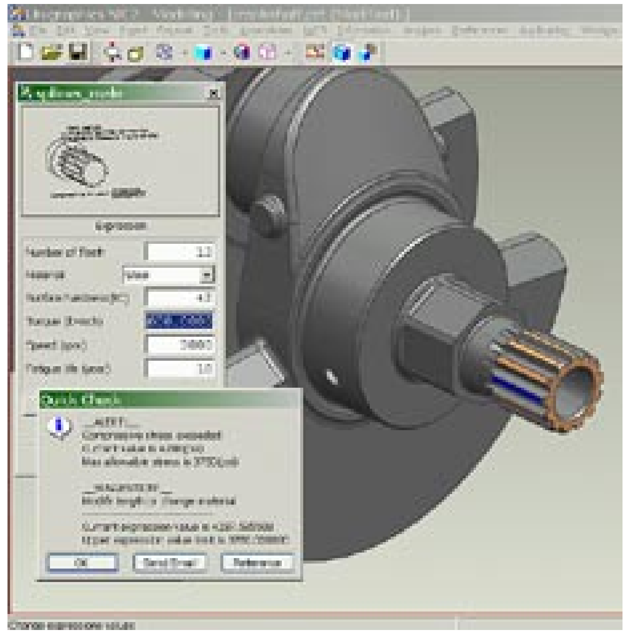
In this real-life customer example, a sporting goods manufacturer was struggling with a certain parametric edit that created a particularly tricky topological change in a family of complex (over a thousand features) parts. Though the user was able to manually "help" NX resolve the design change during each update, NX was unable to resolve the problem automatically for all members of the family of parts. However, the designer understood the "rule" behind the manual topological edit he was making, and after this relationship between two features in the design history was added, NX was able to completely resolve the model using the new rule. *The important thing here is that no rebuilding or "starting over" was necessary for inclusion of design rules in a model. Users can simply "adopt" features into the KF environment along with all their attributes.*



In the image above the new NX Knowledge Fusion rule is being incorporated into the design.

KF Example 2 - Embedding a sanity check within a design process catches errors early

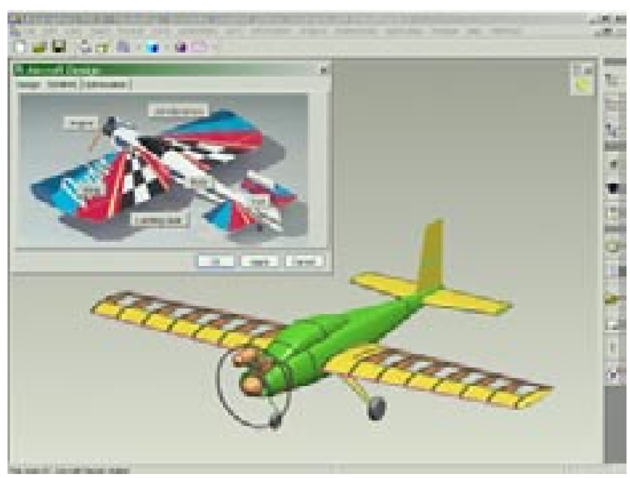
In this example, a performance requirement (torque, in this case) for a splined shaft is changed. The user is immediately notified that the compressive stress limit has exceeded its allowed value. Quantitative feedback about the current state is provided to the designer, along with recommendations for potential remedies.



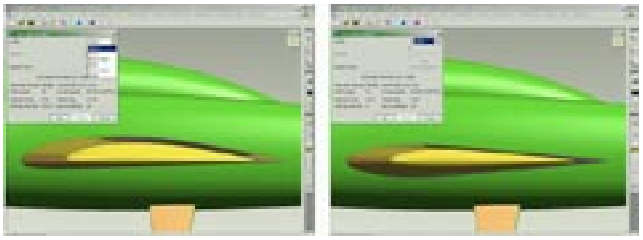
In the image above a pre-planned NX Quick Check shows that the user has violated a simple analysis check on the allowed compression value. Performing this check during the assembly process ensures product validity at design time rather than later - a better technique.

KF Example 3 - Model airplane illustrates subsystem decomposition, UI Styler, and using simple analyses to analyze the design.

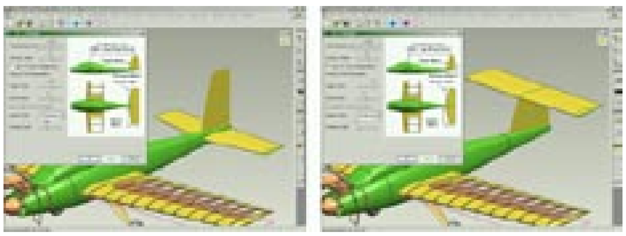
In this example of a system design, a remote-control airplane design is decomposed into major subsystems and driven from a Knowledge Fusion interface.



Changes like swapping airfoils can be made at the subsystem level, while predicting top system-level performance characteristics in real-time.



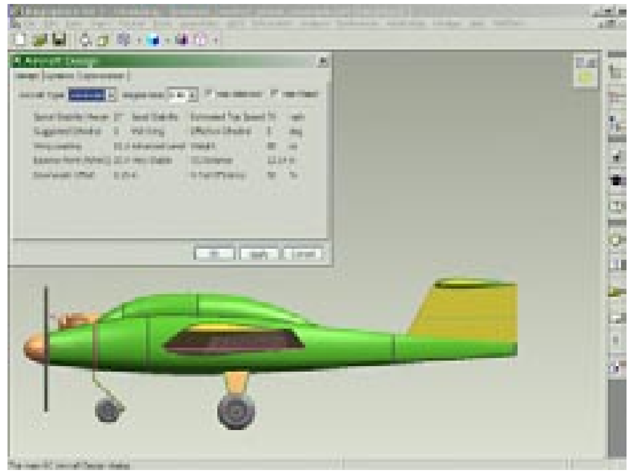
Major topological swapping of subsystems like the replacement of this tail assembly or the replacement of the landing gear from a tricycle configuration to a tail-dragger can also be performed. Also, an optimization loop for positioning the landing gear relative to the center of mass of the plane (to which the landing gear contributes, of course) is included in the system model.



Note the dramatic changes made to the tail assembly above.

In the first (tail low), our design tab indicates that the efficiency of the tail is quite low (42%) because the downwash off the wing is interfering with the tail. Sensing this, the analysis is rerun after changing the tail configuration to a "T" tail. This change requires that several calculations are done in a reverse order from the calculations for sizing a normal tail. Because the system has been decomposed into "self-sufficient" objects, the calculation method for each tail type can be self-contained within the tail object, and adding new tail types does not require editing of the other types. *The flexibility of the Knowledge Fusion rules engine to determine correct dependencies between rules and to adapt the calculation order to react to changing conditions is important to system design, where dependencies are not necessarily unidirectional.*

The result is an optimized remote-control plane with better performance and stability characteristics, as reported by the dialog showing metrics for the top-level system.



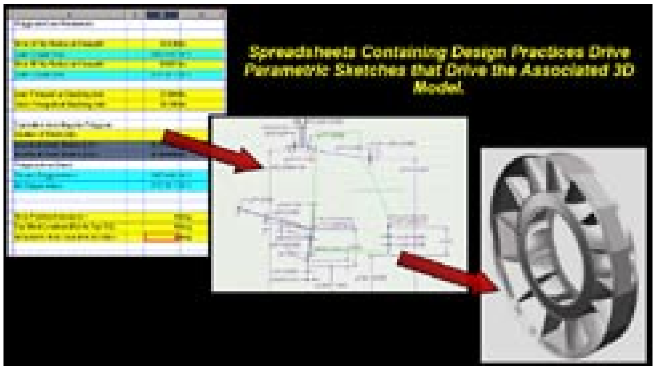
The analysis depicted above shows markedly improved handling characteristics for the plane.

The steps required to build KF functionality are usually taught in a one or two day Knowledge Fusion Introductory Class. **Typical users should be able embed engineering rules and knowledge into the actual design using these KF rules.** Such rules can be made available to other users on the system for "knowledge reuse" through libraries or via Teamcenter.

3. Requirements capture aids Systems Engineering

Teamcenter Requirements, via groupware, delivers product requirements to all entitled users who participate in the product lifecycle. It can also bring customers directly into the extended enterprise and reflect their concerns from the start of the product lifecycle to its conclusion. This is most important because product requirements are the common language for communicating product expectations. Effective communications across the extended enterprise are crucial for optimizing the product lifecycle. Not only that, but product requirements provide the best measures for knowing when products are finished, whether they meet business targets, and how they conform to requirements - a measure of quality.

Teamcenter Requirements enables centralized capture, viewing, and organization of product requirements. The software provides web based access into a groupware environment that facilitates enterprise requirements management. Requirements cannot influence product development if they are not actively connected into your design process using connected requirements. You can enable the requirements that you manage under Teamcenter Requirements to flow into other Teamcenter solutions (including Teamcenter Enterprise and Teamcenter Engineering). Teamcenter Requirements enables you to hook a standard Excel spreadsheet into your requirements database.



The example above illustrates just that. An Excel spreadsheet drives a WAVE diagram, which in turn drives the model.

Particularly interesting is work being done by the UGS PLM Solutions Automotive Center of Excellence. Teams are actively engaged in rethinking the design process by connecting the bill-of-material to requirements, such that if requirements change, a more automated system will assist not only in replacing key design elements and parts, but also making the sometimes dramatic changes in the design to accommodate them. For instance, replacing an automotive door will also require new window mechanisms, new electrical controls, body changes to fit the new door, changes in the supporting underbody structure, and others. Thus, massive design changes could more easily be evaluated.

4. Simulation validates and optimizes the digital design

Validation and simulation components offer three categories to validate designs: analyst tools, tools for early stages of design validation, and tools for checking against corporate standards and processes. Not just for validation, simulation can also be used to set requirements if used in early design stages to test alternatives. A wide variety of analysis tools is available, including intrinsic Unigraphics NX simulation and analysis tools as well as those of their partners. Tools typically used by CAE experts might include linear and nonlinear, thermal, response, noise, dynamics, and crash analyses. The forthcoming tight integration of NX Nastran will only serve to amplify these choices as well as expanding the optimization possibilities.

Some tools that have been designed specifically within NX for early stage validation include Strength Wizard, Molded Part Validation, and Scenario for Structures and Motion.

The third category, corporate standards and processes, includes Quick Check and Check-Mate. Check-Mate validates that the model meets corporate standards, AND can be extended using Knowledge Fusion capabilities to add additional and custom checks.

Within NX, the CAE digital simulation models are always up to date with the CAD model, allowing validation to be an integral part of the system design. Once the design revisions process starts, NX CAE users are advised if the design model changes. They can choose to update the CAE model automatically to accommodate changes. *This associativity persists for individual components as well as assemblies, and ensures that the CAE models and analysis results remain synchronized with the design.*

How to get started

Does it fit?

Keep in mind that systems engineering is likely happening within your company for many of your current products. However, without a specific systems engineering organization, it is likely an informal process with mixed or unrecorded results. We recommend that management consider how design trade-offs are made and by whom. A formal Systems Engineering process should greatly benefit the visibility of how key product decisions are made. Any decision to evaluate these processes can greatly benefit from independent advice.

When you are ready to proceed on an implementation, carefully consider a stepped approach that delivers value during each phase. UGS PLM Solutions offers a combination of software and services that will assist in implementing a formal Systems Engineering process in your company.

About this paper

This paper was sponsored by UGS PLM Solutions. The content of this paper was written and produced by TechniCom, Inc. and the conclusions are solely those of TechniCom, Inc.

For more information

See the Executive Summary above; contact UGS PLM Solutions at www.ugsplm.com/nx or at the contact information on the next page.

TechniCom, Inc. - 66 Mt. Prospect Avenue - Clifton, NJ 07013 USA
(973) 470 9110 - http://www.technicom.com

Corporate Headquarters

United States
5400 Legacy Drive
Plano, Texas 75024
1 972 605 6000

Regions

Americas
13690 Riverport Drive
Maryland Heights, MO 63043
United States
800 498 5351
Fax 314 264 8900

Europe
Norwich House Knoll Road
Camberley, Surrey
GU15 3SY
United Kingdom
44 1276 705170
Fax 44 1276 705150

Asia-Pacific
Suites 3601-2, Citibank Tower
Citibank Plaza, 3 Garden Road
Hong Kong
852 2230 3333
Fax 852 2230 3200

UGS PLM SOLUTIONS
an EDS company