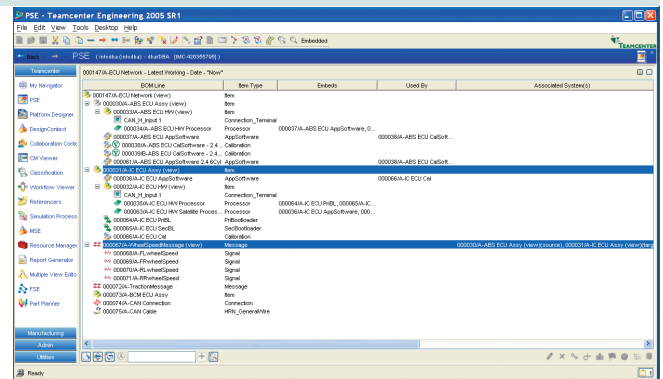


Teamcenter

www.ugs.com

Teamcenter® software helps companies improve the integrated software development lifecycle and the mechatronics design process by enabling product developers to manage released binary software in the same single source of product knowledge they use to manage their mechanical and electronics design information. Users can create and manage dependencies between a product's electronic control units (ECUs), ECU-to-ECU, ECU-to-software, and software-to-software elements. Teamcenter manages the signals and messages for any ECU communications architecture, while enabling users to trace service problems to the product's ECU, binary software and source code.

These functional interdependencies are not known to the product configurations that generate a specific product part list. As a result, mistakes in tracking them lie at the root of many of the final product's software errors.



Mechatronics-related challenges

Increased product complexity characterized by feature-rich modular product designs

Growing number of ECUs and the increasing complexity of embedded software

Decreasing cost of re-programmable memory

Outsourced design and development of embedded software for specific ECUs with the OEM engineering group still responsible for system integrity

Products increasingly differentiated on the basis of embedded software changes

Increased legal regulation of software releases

Teamcenter's embedded software management capabilities solve this dilemma by modeling these interdependencies directly in the software model and validating that these interdependencies are satisfied in any product-level configuration of a binary flash package.

Once a binary solution has been configured into a product computer, the engineer can create dependencies between the binary solution and binaries on other computers. These dependencies mean that a given binary solution will appear in a product-specific flash package only if the second binary solution has been correctly configured into the product under those configuration conditions.

The engineer also can create dependencies between a given executable on one computer and a second computer hardware part in the configuration independent of binaries that execute on the second computer. This approach ensures that the second computer is present in the final product configuration while making certain the requirement that software on the second computer functions correctly with the original software binary.

Integrating the software development process

The processes enabled by Teamcenter's embedded software management capabilities comprise one piece in the complete embedded software development lifecycle. Teamcenter also facilitates integrated source code management by supporting detailed software design and documentation processes that link requirements to the product's source code and binaries. Consistent with its open systems commitment, UGS integrates Teamcenter with key third-party solutions, including IBM Relational Clearcase, to manage and control software development assets.

Teamcenter enables product teams to extend the engineering bill of materials (BOM) by treating software as a separate part number and managing its sourcing, distribution, legal compliance and re-use throughout the product lifecycle.

Product development, assembly plant and dealer service teams can leverage Teamcenter to track as-defined, as-manufactured, and as-serviced software BOMs. Teams can use embedded software management in conjunction with Teamcenter's software configuration management capabilities to quickly trace in-service software problems to their ECU, binary, and source code origin – thereby improving the team's ability to rapidly and efficiently resolve problem-related events.

Teamcenter also enables product developers to improve software traceability so that software-driven electronics can be re-used across multiple investment programs. Teamcenter is especially adept at enabling product makers to control their mechatronics management processes and lower the expenses associated with regulatory compliance and the needs for costly after-sales product-related software service updates.

Business issues

Product complexity is increasing in many industries. One of this trend's major drivers is the growing importance of software-driven electronics in products traditionally comprised of mechanical components.

A wide range of today's products use software-driven electronics, including automobiles, aircraft, defense systems, machine tools and home appliances. For example, many automobiles are becoming "computers on wheels" with advanced safety features, diagnostics, engine control and other functionality that was virtually unknown only a few years ago.

Feature checklist

Management of released software binaries

Configuration of families of binaries, boot loaders, parameter configuration files and calibration files into binary packages

Configuration of binary packages into product configurations (to align with configuration rules that define the product's complete content)

Creation and management of dependencies between ECUs through ECU-to-software and software-to-software links

Interface to external Flash loader program

Signal management capability (to manage signals and messages for any ECU communications architecture)

Change history management (to facilitate software problem resolution)

In some cases, product makers use software to drive innovations that capture the imagination of today's consumers. In other instances, software-driven electronics enables common mechanical product configurations to deliver better performance or different operating characteristics that are highly valued by specific market segments.

Considered together, these business opportunities spur today's growth in mechatronics design and the need for highly integrated mechanical, electronics and embedded software components. However, mechatronics design involves a complex product development process that requires precise control over multiple components created using different design applications. This process raises new challenges for development teams, including the ability to better manage design and requirement changes, as well as to minimize take-to-market cycle times.

As cycle times decrease and pressure for functional and performance improvements rise, many design and engineering companies find they can meet customer expectations by increasing their products' software and electronics content while reducing their tooling and assembly costs.

However, while electronic and software content often holds down manufacturing cost, this approach increases the complexity of product design and configuration (e.g., instances where a single software error can trigger unacceptable product failure). In this environment, the software on every product on the product line must be proven to function correctly, as well as to work with the computers, wiring, sensors, actuators and other electro-mechanical content in every product configuration.

Teamcenter's embedded software management capabilities meets these needs by capturing the complex interdependencies between software, computers, product configurations and other hardware that come together in each product configuration. By configuring and tracing these interdependencies, Teamcenter creates validated binary flash packages to be placed on the product in production.

A variety of benefits accrue from this improved process including significant error reduction, increased error traceability and more precise software test and configuration verification. In turn, these benefits translate into faster and more reliable new feature introductions that incur less risk and lower long-term warranty liability.

Use cases for today's challenges

Traditionally, only ECU hardware (with software already embedded) is included in the engineering BOM. However, today's mechatronics designs require product teams to manage software components as parts in dependency-driven product configurations so that:

- Part numbers can be assigned and the software's sourcing, distribution, legal compliance and re-use requirements can be more effectively managed
- Interim software releases can be aligned with specific product development milestones
- Developers can quickly determine the correct software BOM for a specific product configuration
- Software-to-software dependencies can be accurately tracked to ensure effective change management
- In-service software problems can be rapidly diagnosed after release – including during “software campaigns” – and initial fixes can be performed without the need for rework

► Contact UGS

Americas 800 498 5351
Europe +44 (0) 1276 702000
Asia-Pacific 852 2230 3333

www.ugs.com



UGS, NX, Teamcenter, Parasolid, Solid Edge, Femap, I-deas, Tecnomatix, UGS Velocity Series, Geolus and Transforming the process of innovation are registered marks or trademarks of UGS Corp. or its subsidiaries in the US and in other countries. All other logos, trademarks or service marks used herein are the property of their respective owners. ©2006 UGS Corp. All rights reserved. 9/06