

# Briefing for Navy Open Architecture Working Session

Trevor S. DeLew



# A Leading Product Lifecycle Management Software Provider

**PTC helps manufacturing companies improve their product development processes and win with superior products**

**Employees 4,000**

**FY 2005 revenue \$725 Million**

**Cash and equivalents (as of 10/2/05) \$250 Million**



# 20 Years of Customer Success



<b>Raytheon</b>	<b>TOYOTA</b>	<b>HITACHI</b>	<b>ITT Industries</b> <i>Engineered for life</i>	<b>WelchAllyn</b>	<b>XYRON</b>
<b>AIRBUS</b>		<b>intel</b>	<b>ABB</b>	<b>stryker</b> Medical	<b>Reebok</b> <i>Product Lifecycle Management</i>
<b>BAE SYSTEMS</b>			<b>YORK</b> <i>Leading the World in Heating,</i>	<b>ABIOMED</b>	<b>Herman Miller</b>
	<b>Audi</b>	<b>MOTOROLA</b>	<b>Rexroth</b> Bosch Group		<b>LIZ claiborne</b>
			<b>MITSUBISHI</b> <small>HEAVY INDUSTRIES, LTD.</small>		
<b>GENERAL DYNAMICS</b> <i>Strength On Your Side</i>			<b>BABCOCK BORSIG POWER</b>		<b>MOEN</b> <i>Buy it for looks. Buy it for life.</i>
<b>United Defense</b>		<b>Lam</b> <small>RESEARCH</small> <i>Changing the Value Equation</i>			<b>patagonia</b>
	<b>HARMAN/BECKER</b> AUTOMOTIVE SYSTEMS	<b>TOSHIBA</b>	<b>CATERPILLAR</b>	<b>GUIDANT</b>	
				<b>Medtronic</b> <i>When Life Depends on Medical Technology</i>	
					Limited brands
					<b>ROLEX</b>

## Recent News for PTC in A&D

- ◉ **Boeing** deploys Windchill as enterprise PLM on largest program (FCS) – 10,000+ users
- ◉ **Boeing** deploys Windchill on MMA (Multi-Mission Maritime Aircraft) Program
- ◉ **Boeing** chooses Windchill as change management and collaboration system on 787
- ◉ **Lockheed Martin** deploys Windchill as enterprise PLM on all space programs and system integration programs (11,000 users)
- ◉ **Lockheed Martin** selected for Presidential helicopter (US101) and selects Windchill for PDM/ Collaboration
- ◉ **Lockheed Martin** selects Pro/E as EPI corporate CAD standard
- ◉ **Raytheon** selects Windchill as enterprise PLM standard – (Pro/E is CAD Standard)
- ◉ **NASA** identifies Windchill as the enterprise PLM environment for \$100B U.S. manned space program (Exploration Systems Mission Directorate – ESMD)
- ◉ **Airbus** begins to deploy Windchill throughout its supply chain
- ◉ **UK MOD** selects Windchill for enterprise PLM for new aircraft carrier

# Normalization

- PTC has been providing Computer Aided Design & Process Enabling Applications for 15 Years
- Our Technology has been on the forefront of distributed design & design outsourcing across many verticals, especially Federal, Aerospace, & Defense
- Our customers have demanded that we provide capabilities that support Open Architecture or Modular Product Architecture development processes
- Component Reuse (whether mechanical, electrical, or software) and interface management are key enablers to us supporting those activities
- During the last 4 years, marquee programs driven by DoD and other agencies have applied our capabilities to their respective open architecture initiatives

# Questions

→Please address the following questions when developing your Repository Presentation:

→What is the structure / architecture of your repository?

- ◉ Windchill is a J2EE based application framework that relies on oracle for persistence.

→What tools do you use and support?

- ◉ As a standards based application, we support integrations to a wide array of authoring tools. We have out-of-the-box integrations to DOORS, Clear Case, Clear Quest, CAD systems, and a number of other COTS products.

→What types of assets does your repository contain / handle?

- ◉ Windchill is capable of managing any kind of object. Its core object model can be extended using an easy to use graphical interface to create new objects on the fly.

→What is needed by the user to access the repository?

- ◉ Credentials managed in an LDAP that is part of existing infrastructure and bound to architecture.

# Questions

## →By what means are assets / components moved in and out of the repository?

- ⊙ Objects are access controlled and configuration managed. Users first need access to system and required to check-in / check-out for modification access.

## →How do developers / integrators gain access to assets / components within the repository?

- ⊙ If credentialed, access via a web browser.

## →How does your repository protect data rights / intellectual property?

- ⊙ IP is protected through a combination of Organization Containers and Access Control policies contained within those Containers.

## →How is classification of assets / components handled within the repository?

- ⊙ If classification is related to a specific system, its respective product / system tree provides the classification. If asset is to be used independent of a specific system, PartsLink can be used which is a component classification system.

## →How are changes to the assets / components managed?

- ⊙ Check-In / Check-Out, Lifecycle, & Workflow

## Questions

- **What metrics are collected related to the repository?**
  - ⊙ Every transaction logged in database which can be queried and formatted for any number of customer defined reports.
- **Do you have help desk / technical support to aid users with using the repository?**
  - ⊙ Technical Support 24X7 for maintenance paying customers.
- **What are there additional capabilities related to your repository that have not been mentioned in response to the above questions?**
  - ⊙ Application services for a comprehensive collaborative environment supporting proceses across various domains.
- **What are the outstanding issues related to your repository that have not been mentioned in response to the above questions?**

# Key definitions – Product Architecture terminology

## Platform

- A set of parts, sub systems, interfaces and manufacturing processes that are shared among a set of products and allow the development of derivative products with cost and time savings

## Product Architecture

- The scheme by which the functional elements of the product are arranged into physical chunks and by which the chunks interact

## Module

- A physical entity (chunk) with specified interfaces defined according to company-specific reasons
- “The key difference between modules and assemblies is the management and control of interfaces” – Client Platform Manager

## Interface

- A defined relationship between two modules, components or parts on the physical level within a product structure
- "Standard Contracts" with owners of key information in the scope of a Modular Product Architecture

# Key definitions – Product Architecture terminology

## Sub-assembly

- A devised product structure satisfying a minority of stakeholders e.g an assembly “sub assembly”
- Consequently: ”A sub-assembly is not always a module, but a module is a sub-assembly.”

## Modularization

- The decomposition of a product into building blocks (modules) with specified interfaces, driven by an entity-specific reason

## Product Family

- The products that share a common platform but have specific features and functionality required by different sets of customers are a product family (Meyer & Utterback, 1993)

## Product Variant

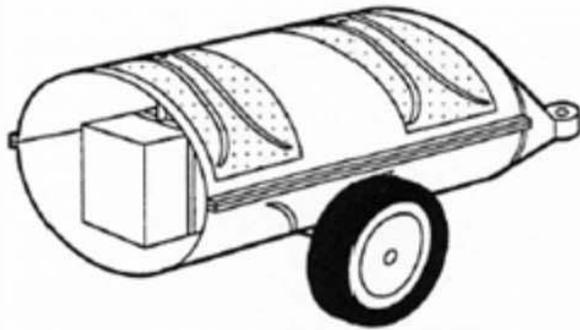
- A unique product configuration satisfying a set of specific customer needs

## Commonality

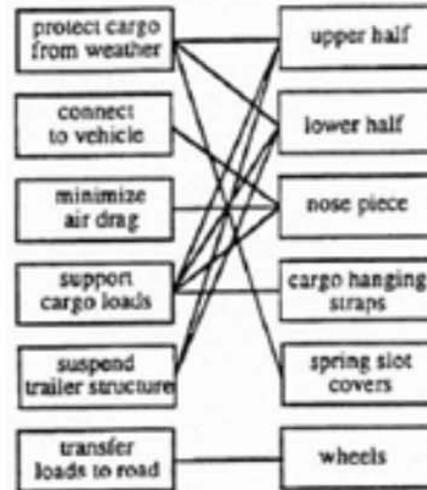
- The degree to which a part or module is used in the product variants of a product family.  
100% commonality = a common part/module

# There are Two Core Architectures our Customers Develop

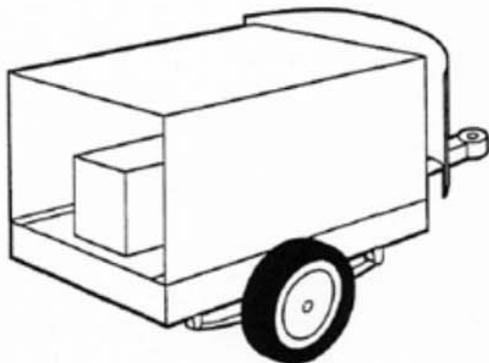
## Integral Architecture



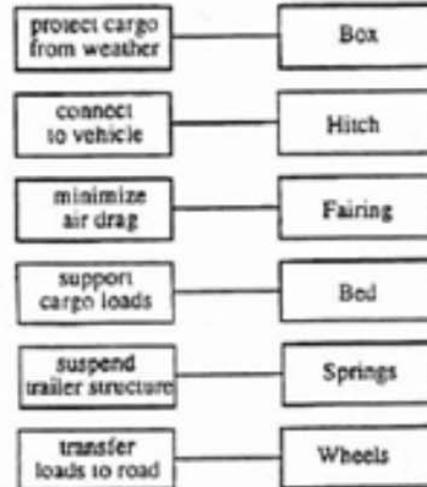
### Functions “Chunks”



## Modular Architecture



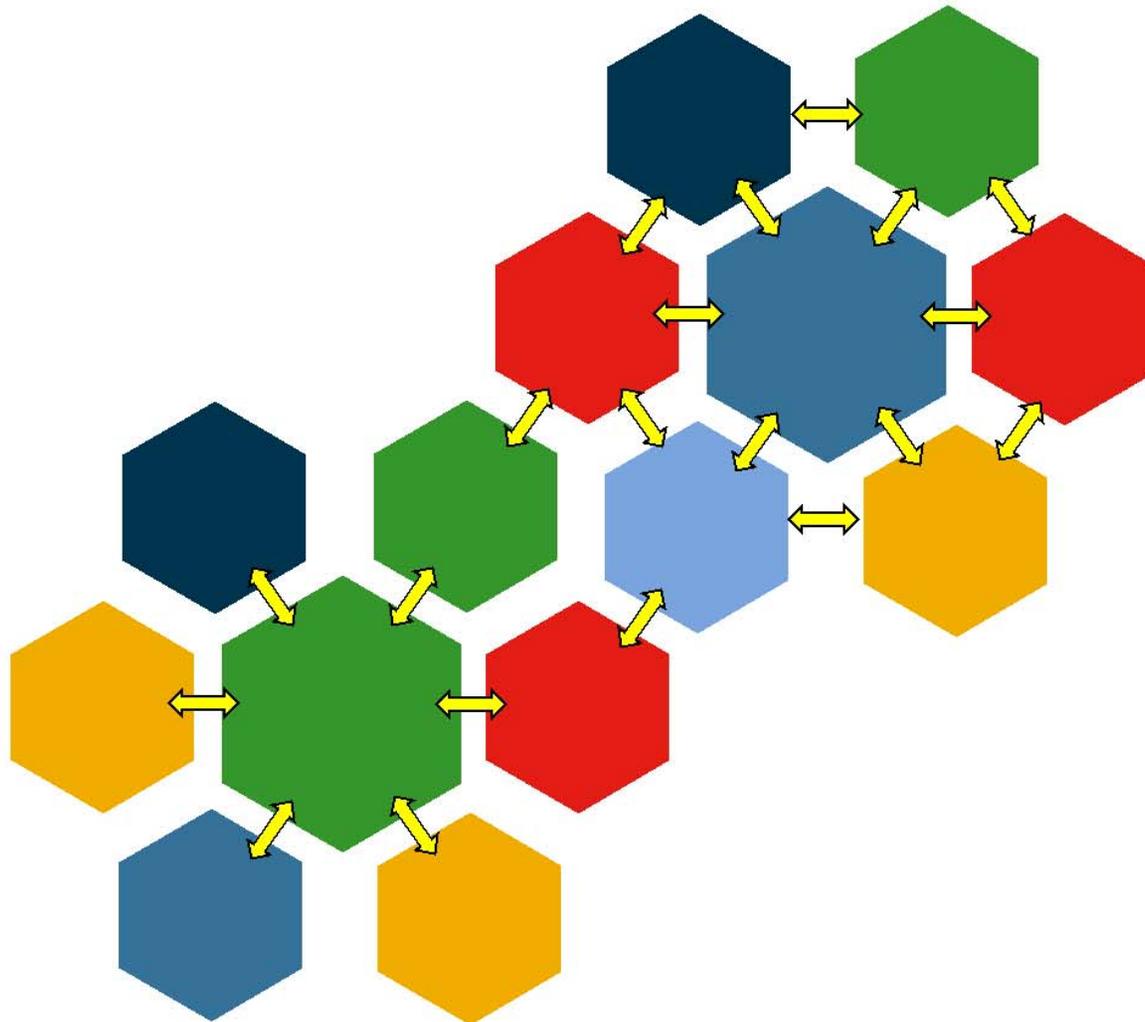
### Functions “Chunks”



**Note: Products are rarely strictly modular or integral.**

# What is a Modular Architecture?

## Basic Characteristics of Modular Product Architectures



# What is an Interface?

## 4 major classification types of interfaces:

- ⊙ **A – Mechanical attachment**

- Bolting, Clamping, etc...

- ⊙ **T – Transfer**

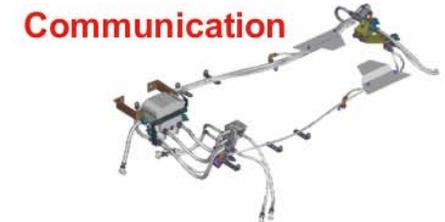
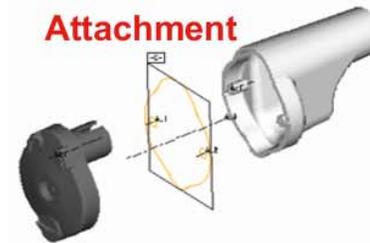
- Electrical power, fluid or gas flow

- ⊙ **C – Communication**

- Data or signals

- ⊙ **U – User interface**

- Man/machine interface – something a user must interact with



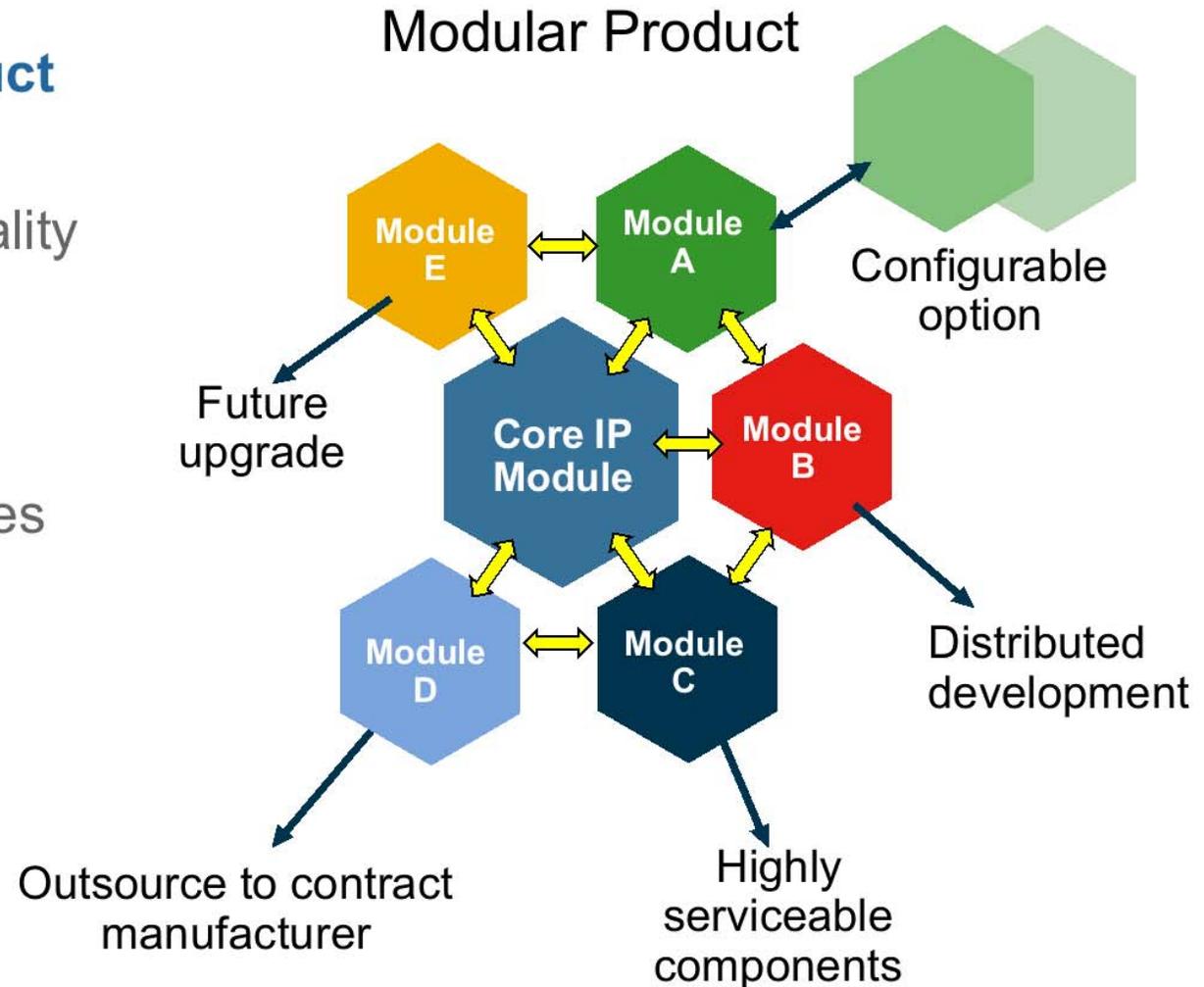
## Any single interface between two modules will likely contain a combination of types

- ⊙ *Example:* The interface between an engine and transmission will contain both mechanical attachment (the transmission bolts to the engine) and transfer (the torque and horsepower that is transferred)

# The goal: aligning system architecture to enable operational strategies

## A well-defined Modular Product Architecture (MPA) enables:

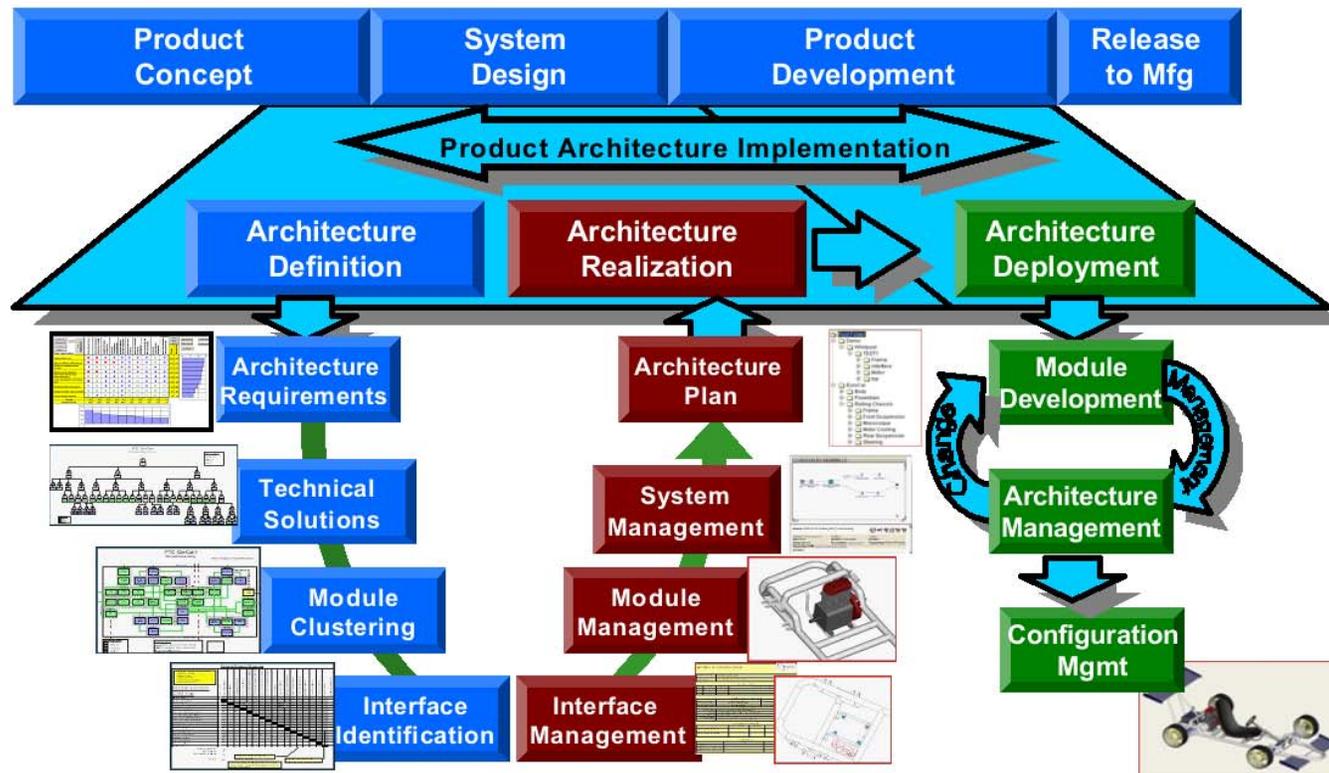
- ⊙ Increased design commonality
- ⊙ Increased product variants
- ⊙ Distributed development
- ⊙ Protected core competencies
- ⊙ Outsourcing
- ⊙ Improved serviceability
- ⊙ Simplified upgrading



# Modular Product Architecture Development Process

The PTC modular product architecture development process involves three deliverable-based phases:

- ⦿ Architecture Definition
- ⦿ Architecture Realization
- ⦿ Architecture Deployment



# Architecture Definition Phase

The Architecture Definition phase results in the documented description of a new product structure:

- ◉ Modules identified
- ◉ Interfaces identified

Deliverables are Office documents that describe the structure:

- ◉ Diagrams of product structure trees
- ◉ Excel spreadsheets of module and interface matrices
- ◉ PowerPoint presentations of the structure

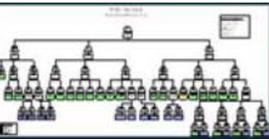
Architecture  
Definition



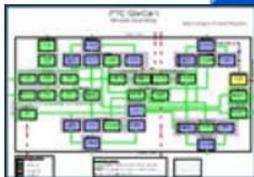
Architecture  
Requirements



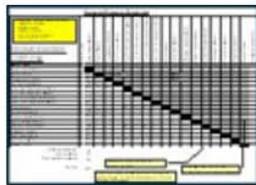
Technical  
Solutions



Module  
Clustering



Interface  
Identification



# Architecture Definition Phase

## Team:

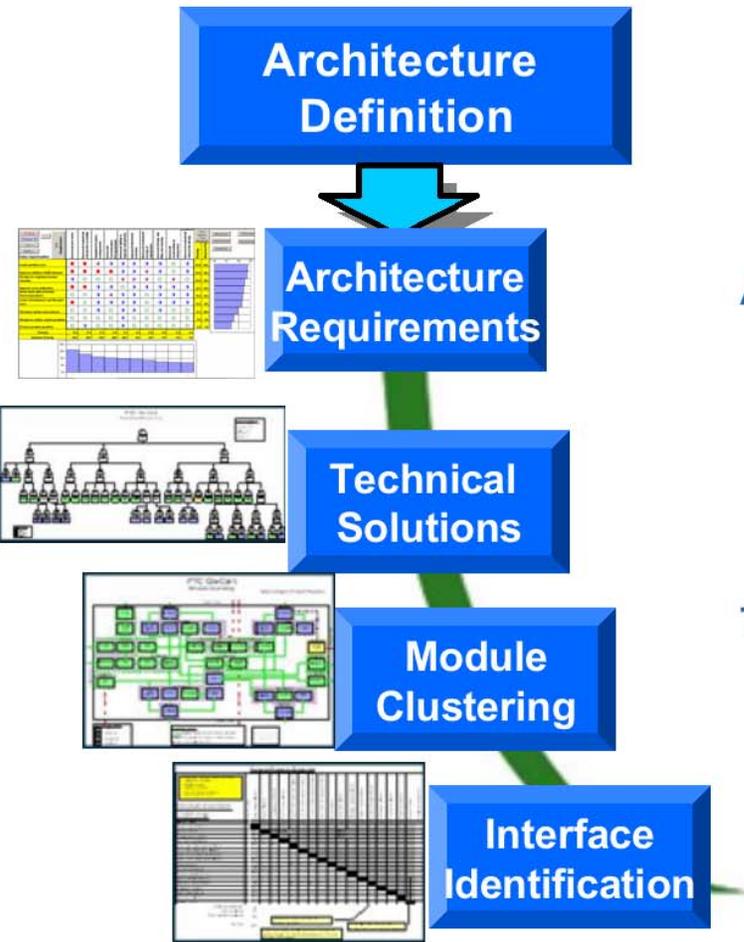
- Engineering/Platform Management
- Architecture Management, System Engineers, Module Managers
- Cross-functional stakeholders
  - Marketing, Sales, Manufacturing, Assembly, Service, etc...

## Activities:

- Initial event: Facilitated workshop with entire team for initial architecture definition
- On-going activities: Virtual collaboration to update/refine deliverables and get stakeholder approval for architecture changes

## Tools and Usage:

- Many System Design Tools – DOORS, Cradle, Rational
- ProjectLink:
  - To store architecture definition deliverables and enable virtual collaboration with the entire team, which may be widely distributed
- PDMLink:
  - Document vaulting of definition deliverables



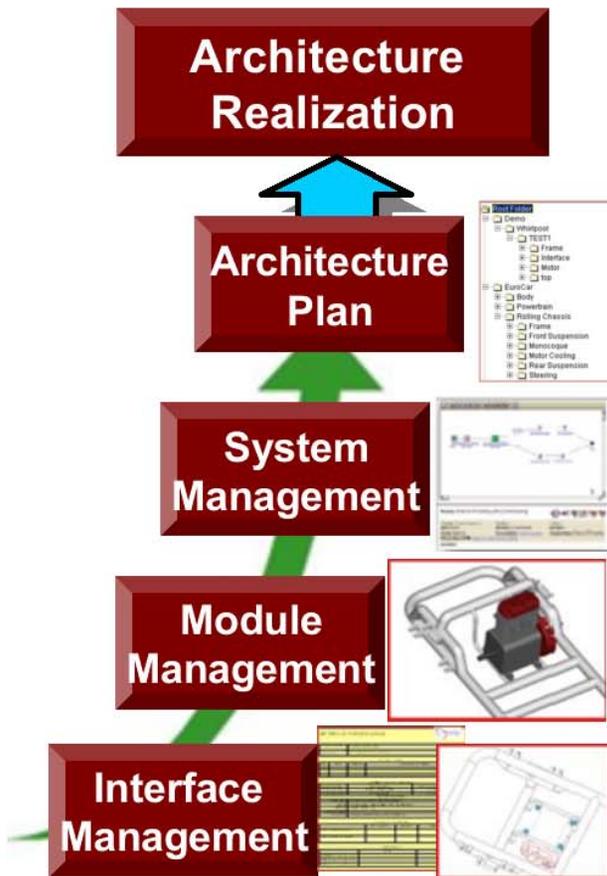
# Architecture Realization Phase

The Architecture Realization phase results in the embodiment of the new product structure:

- Upper-level product structure in PDM system
- Module specifications & system framework
- Interface specifications & system framework

Deliverables are actual business objects managed by PDM system:

- Product Structure
- System Objects
- Specification documents
- Not released at this time
- Under informal change control



# Architecture Realization Phase

## Team:

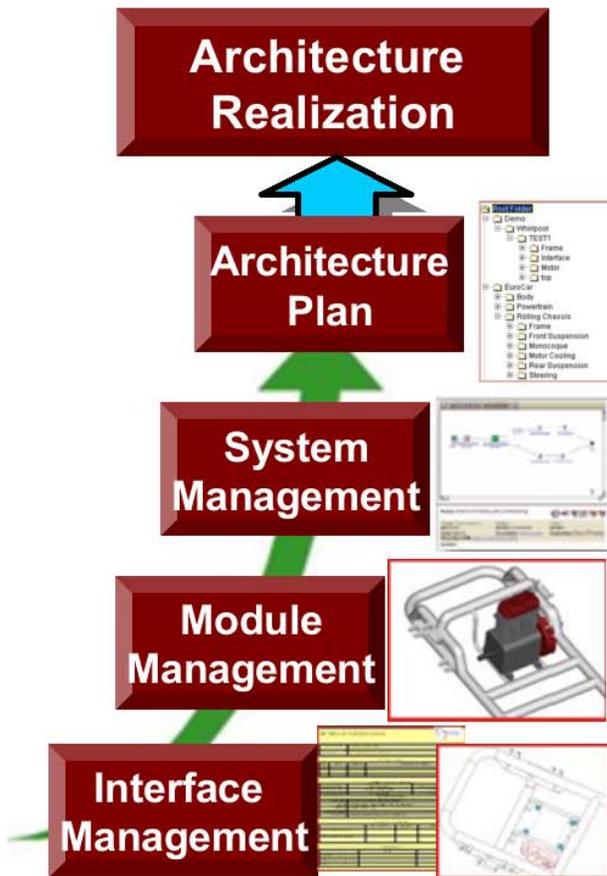
- System Engineers, Architecture Managers, Engineers/Designers
- Engineering/Platform Management
- Cross-functional stakeholders
  - Marketing, Sales, Manufacturing, Assembly, Service, etc...

## Activities:

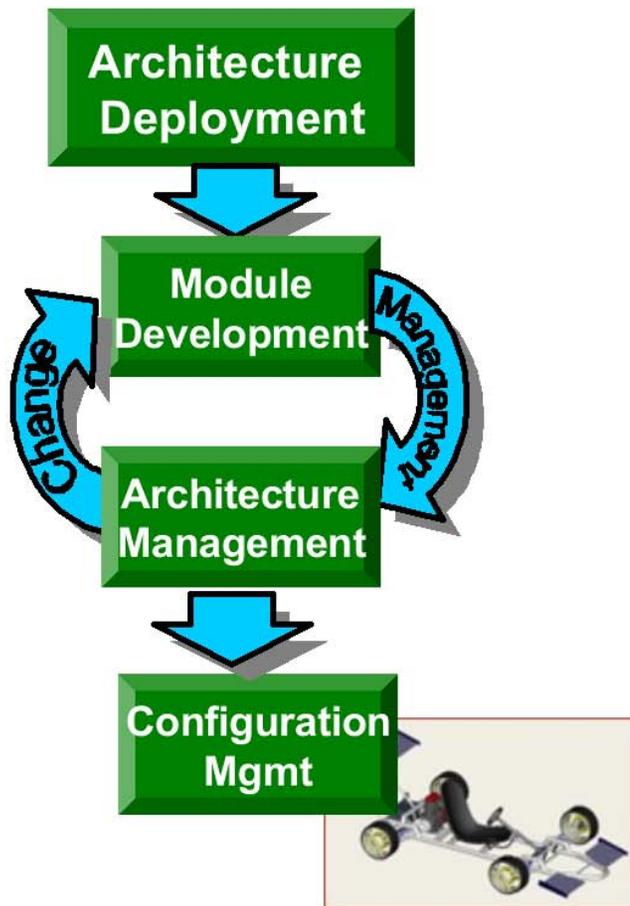
- Creation of Interface / Module structure PDMLink
- System-level functional simulation and analysis
- Design reviews with extended team

## Tools and Usage:

- Many System Design Tools – DOORS, Cradle, Rational
- ProjectLink:
  - Continued collaboration on the architecture definition deliverables
- PDMLink:
  - Vaulting and change control for the product structure, interface parts, module structure, interface and module specifications



# Architecture Deployment Phase



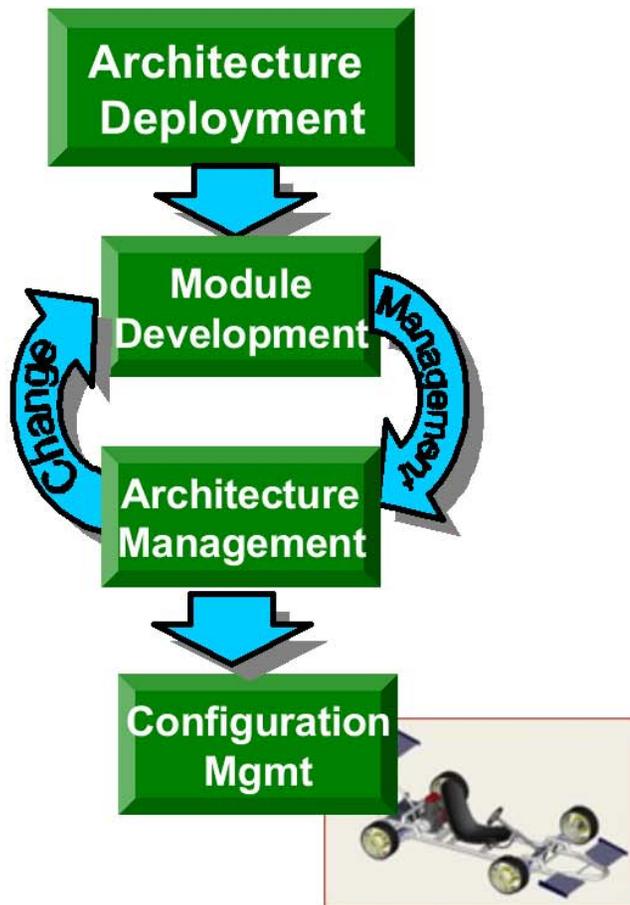
The Architecture Deployment Phase results in a fully-defined product design that meets its identified architecture requirements:

- ◉ Completed product structure in PDM system
- ◉ Completed system content and documentation for the product

The Deliverables are the actual business objects managed by the PDM system:

- ◉ System Artifacts, Assemblies, Drawings, Specifications
- ◉ Released for production
- ◉ Under formal change control

# Architecture Deployment Phase



## Team:

- System Engineers, Architecture Managers, Engineers/Designers
- Engineering/Platform Management
- Suppliers
- Engineering design partners
- Cross-functional stakeholders
  - Marketing, Sales, Manufacturing, Assembly, Service, etc...

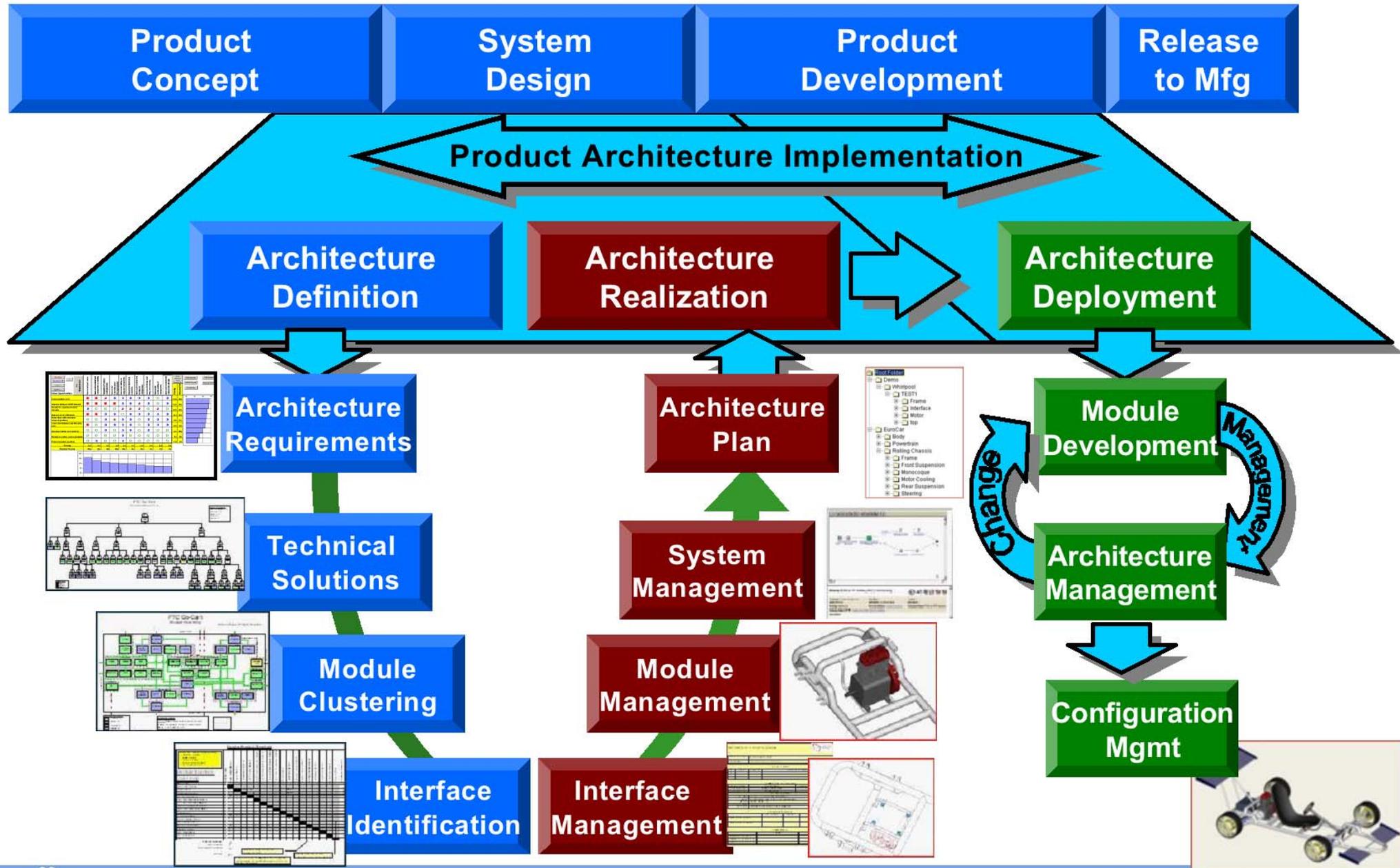
## Activities:

- Detailed design of modules
- Detailed functional simulation and analysis
- Design reviews of detailed design

## Tools and Usage:

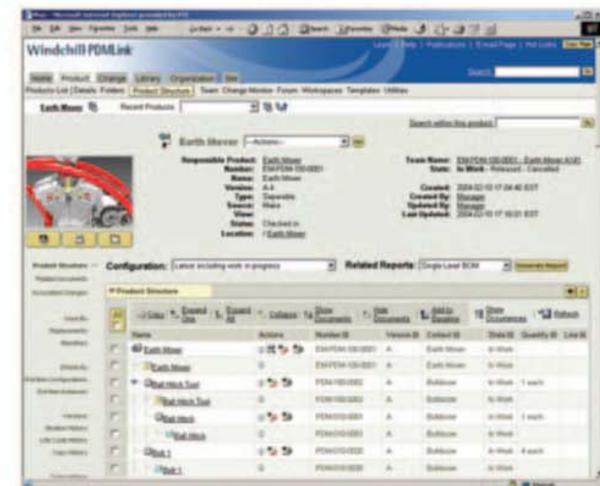
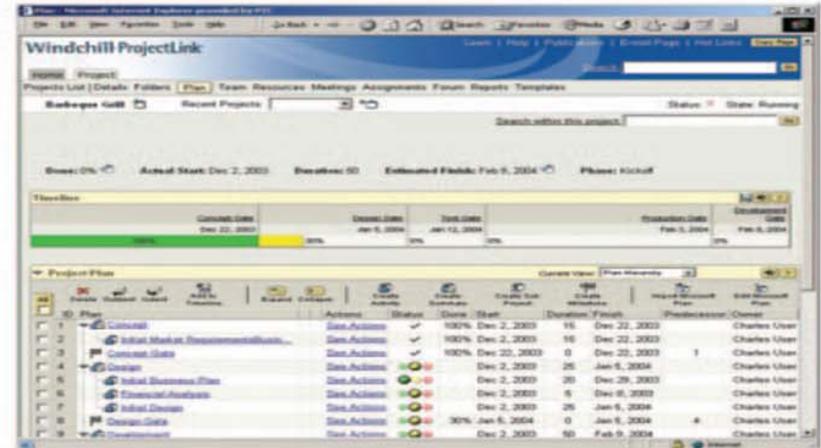
- Many System Design Tools – DOORS, Cradle, Rational
- ProjectLink:
  - Multiple projects for managing design collaboration on individual modules with outsourced design partners, suppliers and distributed engineering teams
- PDMLink:
  - Vaulting and formal change management for all parts, assemblies and drawings in the product structure, including interfaces and modules

# Modular Product Architecture Development Process



# PTC Open Architecture Enablement Tools:

- ProjectLink provides a team-centric dynamic collaboration environment, which will help effectively manage the team, the process and the deliverables associated with the architecture definition activities.
  
- PDMLink enables the project team to reference existing product data and also to share our own major deliverables with the extended enterprise.



# ProjectLink

- **ProjectLink provides a single repository for all of the deliverables and documentation generated during the Architecture Definition Phase.**
  - ⊙ Create links within the project to external documents and websites, or even share product data from PDMLink.
  - ⊙ Cross-functional representatives might be participating in the architecture definition activities for more than one product or platform.
  - ⊙ A single place where they can go to review a proposed architecture change and provide their input will make this process much more efficient.
  
- **ProjectLink provides a rich set of tools to enable the rapid iteration and collaboration of architecture documentation while still providing a structure for maintaining the history of changes.**
  - ⊙ Iteration history to maintain previous versions of the documents, as well as the discussion forums where team members can document their comments as the project evolves.
  - ⊙ The architecture will evolve over the course of the system design phase, which may last several months, one will need to be able to review what changes were made over the course of time and understand why they were made.
  - ⊙ The document routing and approval capabilities built into ProjectLink are ideal for managing the review and approval of documents within a diverse team – giving the team leader the ability to “push” the updated documents to the other team members for review.
  - ⊙ One can track the status of these approvals to make sure we don’t delay the process waiting for approval of architecture changes.
  - ⊙ ProjectLink also provides subscription capabilities where the team members can be automatically notified by the system whenever a particular document is updated.
  - ⊙ These capabilities provide the much needed structure for managing the architecture changes at this stage of the design process without burdening the team with a formalized release process.

# PDMLink

- **PDMLink serves as the ultimate repository for all product data available across the entire enterprise.**
- **During the architecture definition phase PDMLink can be used to search for existing product designs upon which to base the new design.**
- **The integration between PDMLink and ProjectLink will allow for the direct sharing of these artifacts with the architecture definition projects in ProjectLink.**
- **Gradually sharing of the results of the architecture realization efforts from PDMLink into this project as these artifacts become available.**
- **At certain milestones during the architecture definition process one will also be sharing some documents and deliverables that we created directly from ProjectLink into PDMLink.**
- **This will be important because visibility of these documents to the extended enterprise, while maintaining the ability to iterate and collaborate upon them in the team-centric project environment.**
- **Finally the official release all major deliverables into PDMLink will allow for management by a formalized change process.**

# A Representative Example

## FCS

Distributed Product Description

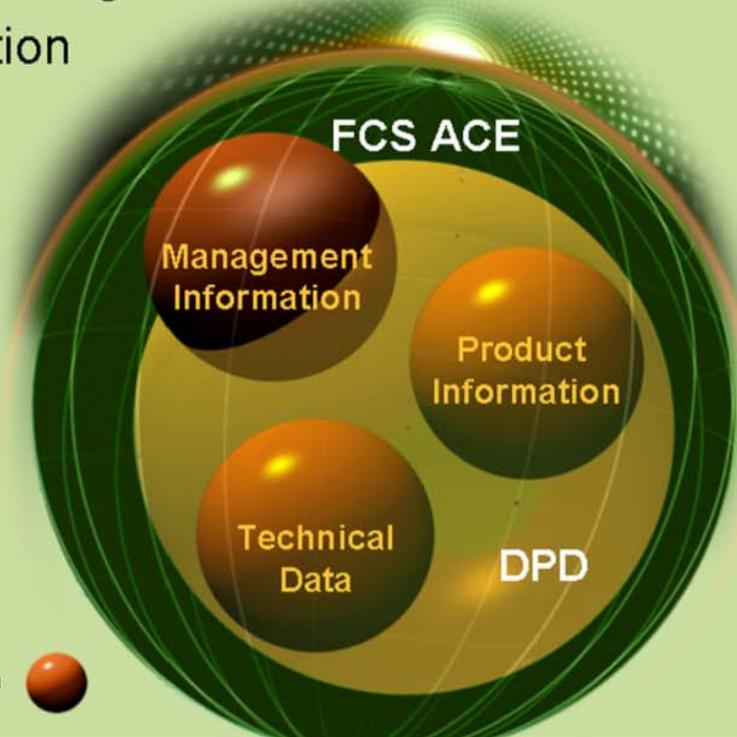
### ACE and the DPD

The FCS **Advanced Collaborative Environment (ACE)** is the single, authoritative, integrated source of FCS:

- Management Information
- Product Information
- Technical Data

The **Distributed Product Description (DPD)** is the subset of ACE providing a single access point to ALL authoritative FCS Product and Product Support Information.

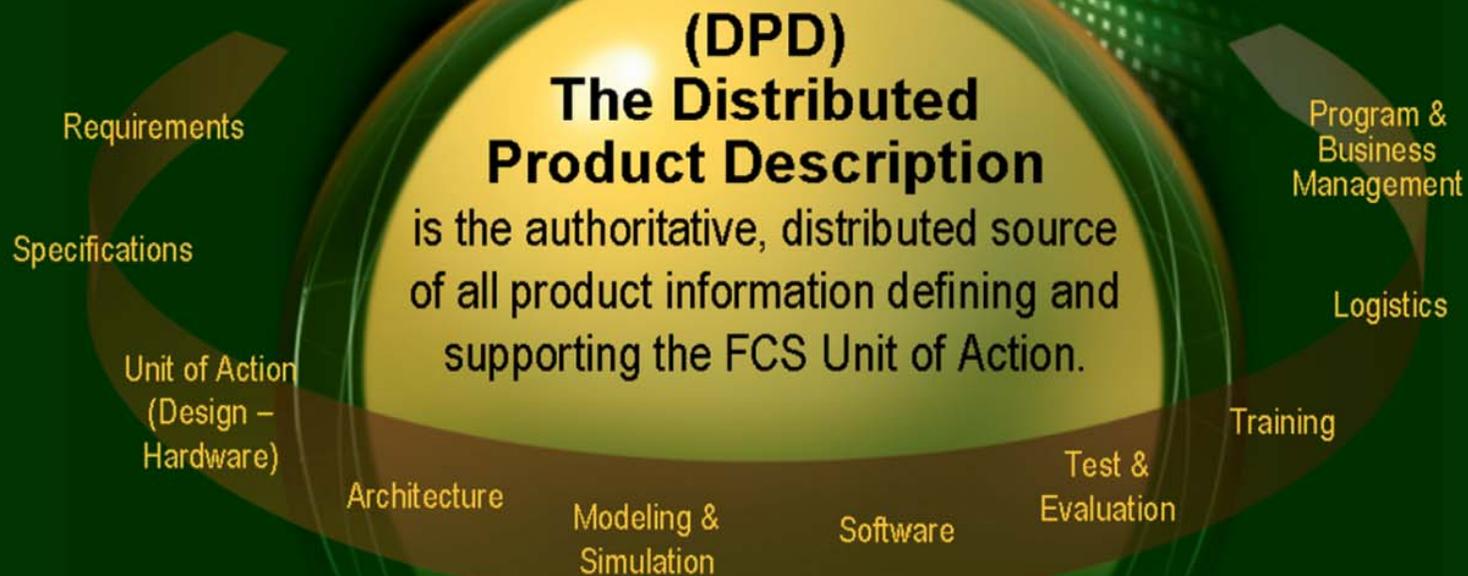
Authoritative Data



# FCS

Distributed Product Description

## What is the DPD?



**The DPD provides Access, Traceability and Configuration Management of all product information over the entire lifecycle of the FCS program.**

# FCS

## Distributed Product Description

### DPD Access

---

- Controlled access and integration to authoritative external One Team Partner and Government data (federation) (*Requirements, Specs, Design, Software, etc.*)
- Navigation of information structures (*Unit of Action, Specification Tree, Architecture views, etc.*)
- Focused search capabilities for direct location of specific product information.

# FCS

## Distributed Product Description

### DPD Traceability

The DPD links all stages of FCS product information to provide lifecycle traceability

- Customer Requirements (*ORD & O&O*)
- Specifications
- Unit of Action (*Design HW*)
- Architecture Views (*Operational, System, Technical*)
- Modeling & Simulation
- Software
- Test and Evaluation (*Validation & Verification*)
- Training
- Logistics



# FCS

## Distributed Product Description

### DPD Configuration Management

- Configuration and Change management processes for all FCS product information
- Controlled, authoritative Product Technical Baselines (*SoS Spec, Architecture, PIDS, Configuration Items, Design, TPMs, etc.*)
- Integrated configuration management of LSI and One Team Partner product information. (*Requirements, Specs, Hardware Design, Software, etc.*)



## DPD Partner Integration Commitments

IPT		DPD Structures				
		Committed to Federate Data	Connected to Federate Data	Requirements to Production	Specification Tree	Unit of Action Product Structure
SSEI	Process & Requirements Defined					
ACE	Demonstrate Capability					
ACE	ACE Interface Deployed (In Production)					
SSEI	Process Implementation					

WBS & Architecture Structures are not populated by One Team Partners

LEGEND	
Complete	
SFR - 8/05	
IPDR - 6/06	
2006 - Q3-Q4	
2007 - Q1-Q2	
2007 - Q3-Q4	
N/A	

## One Team Partner Commitments

Group 1						
MGV	General Dynamics Land Systems (Sterling Heights, MI)	X				
C4ISR	Raytheon Company (Ft. Wayne, IN)	X	X	X		
C4ISR	Raytheon Company (Plano, TX)	X	X	X	X	
MGV	BAE-ASD (Minneapolis, MN)	X	X			X
Group 2						
Army	ARDEC - Army Systems Picatinny	X				
C4ISR	General Dynamics AIS (Bloomington, MN)	X	X	X		
UGV	BAE-ARV (Santa Clara, CA)	X	X			X
MGV / ARV	BAE-GSD (Santa Clara, CA)	X	X			X
Group 3						
UGV	iRobot Corp (Burlington, MA)	X	X			X
C4ISR	Northrop Grumman - ESD (Linthicum, MA)	X				
Training	Northrop Grumman Info Tech (McLean, VA)	X				
C4ISR / Logistics	Northrop Grumman Mission Systems (Carson, CA)	X				
UAV	Northrop Grumman Systems Corp (San Diego, CA)	X				
Group 4 - Defining Federation Plan						
C4ISR	BAE Systems / CNIR (Wayne, NJ)	X				
C4ISR	Textron Systems (Wilmington, MA)	X				
UGV	Lockheed Martin Missiles and Fire Control (Grand Prairie, TX)	X				
Group 5 - Software Suppliers						
C4ISR	Austin Info Systems (Austin, TX)					
C4ISR	Boeing - McDonnell Douglas Helicopter Co. (Mesa, AZ)					
Training	Computer Sciences Corp (Hampton, VA)					
Training	Dynamic Research Corp (Andover, MA)					
C4ISR	General Dynamics Decision Systems (Scottsdale, AZ)					
UGV	General Dynamics Robotics Systems (Westminster, MD)					
Logistics	Honeywell Defense & Electronics Systems (Albuquerque, NM)					