COE Annual Conference & TechniFair
April 25-28, 2004 • Miami Beach, Florida

G-R: Getting the Most Gains Out of Knowledge-Based Engineering

Jeff Rogers and Brian Prasad
Parker Hannifin Corporation
Aerospace Group
Control Systems Division

Parker Hannifin Corporation

Automation	Climate & Industrial Control	Filtration	Aerospace	Fluid Connectors	Hydraulics	Instru-mentation	Seal

Abex NWL	Control Systems	Customer Support Military	Gas Turbine Fuel	Nichols
Parker Aerospace Japan	Control Systems Commercial	Customer Support Commercial	United Aircraft Products	Airborne
Servo Control			Air & Fuel	Aircraft Wheel & Brake

Strategic Charter

To be the premier provider of motion and control systems for our global customers

Mission

Our mission is to provide unequaled value through
superior performance
technical innovation
speed and responsiveness
premier customer service
financial strength
to our customers, company, team members and community

Vision

JustOne TEAM

...our lean vision

Vision

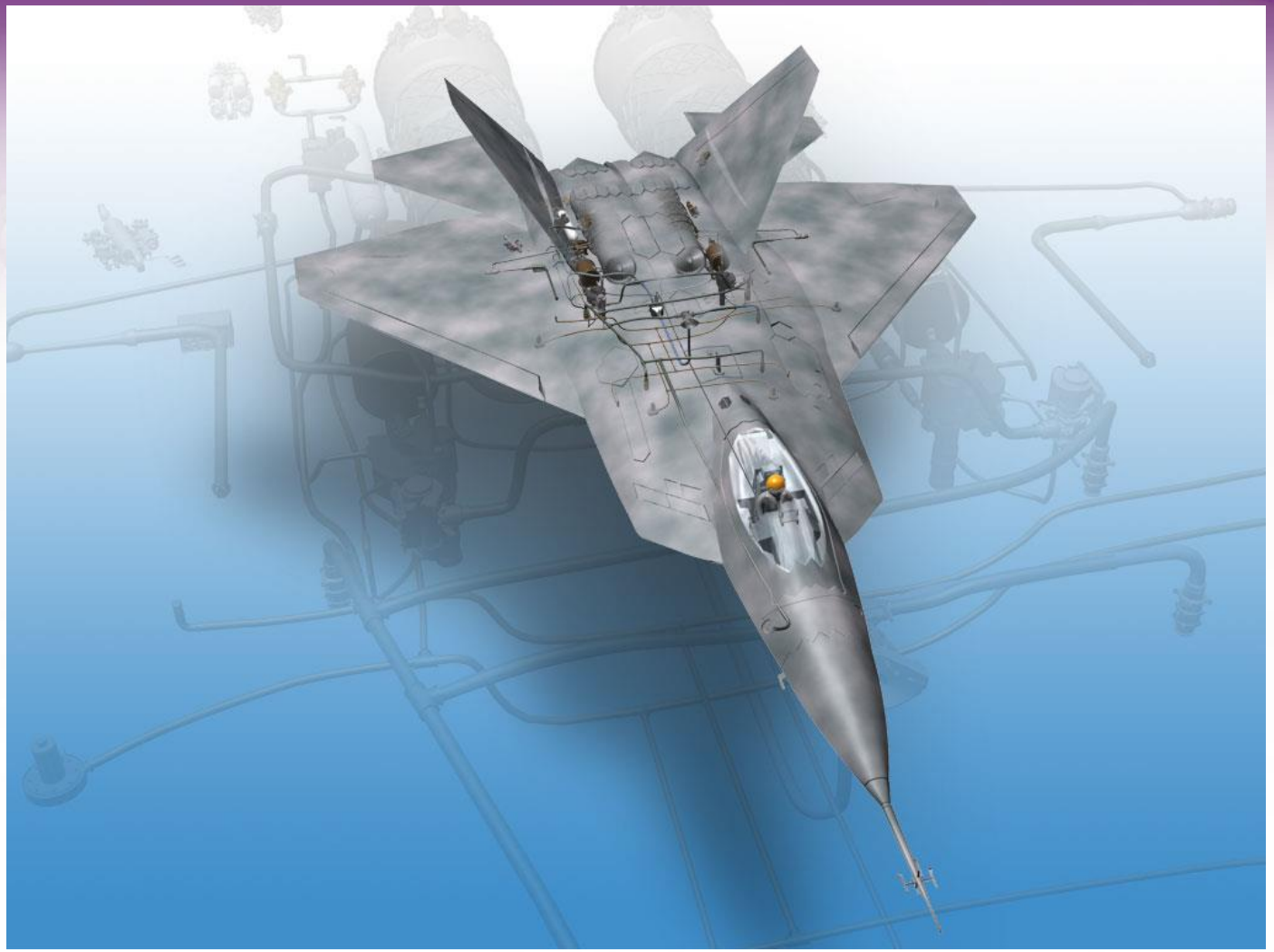
The Spirit of Innovation CSD

Continuous Improvement

Innovation

Values

Integrity
Teamwork
Leadership
Performance
Continuous Learning

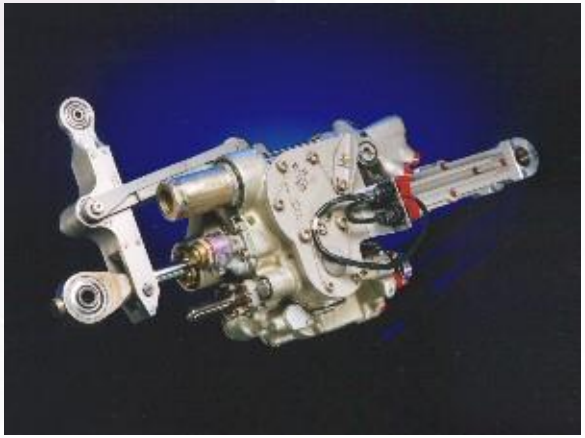


Parker

COE

Bringing Together the Users of CATIA, ENOVIA, DELMIA, and SMARTEAM

www.COE.org

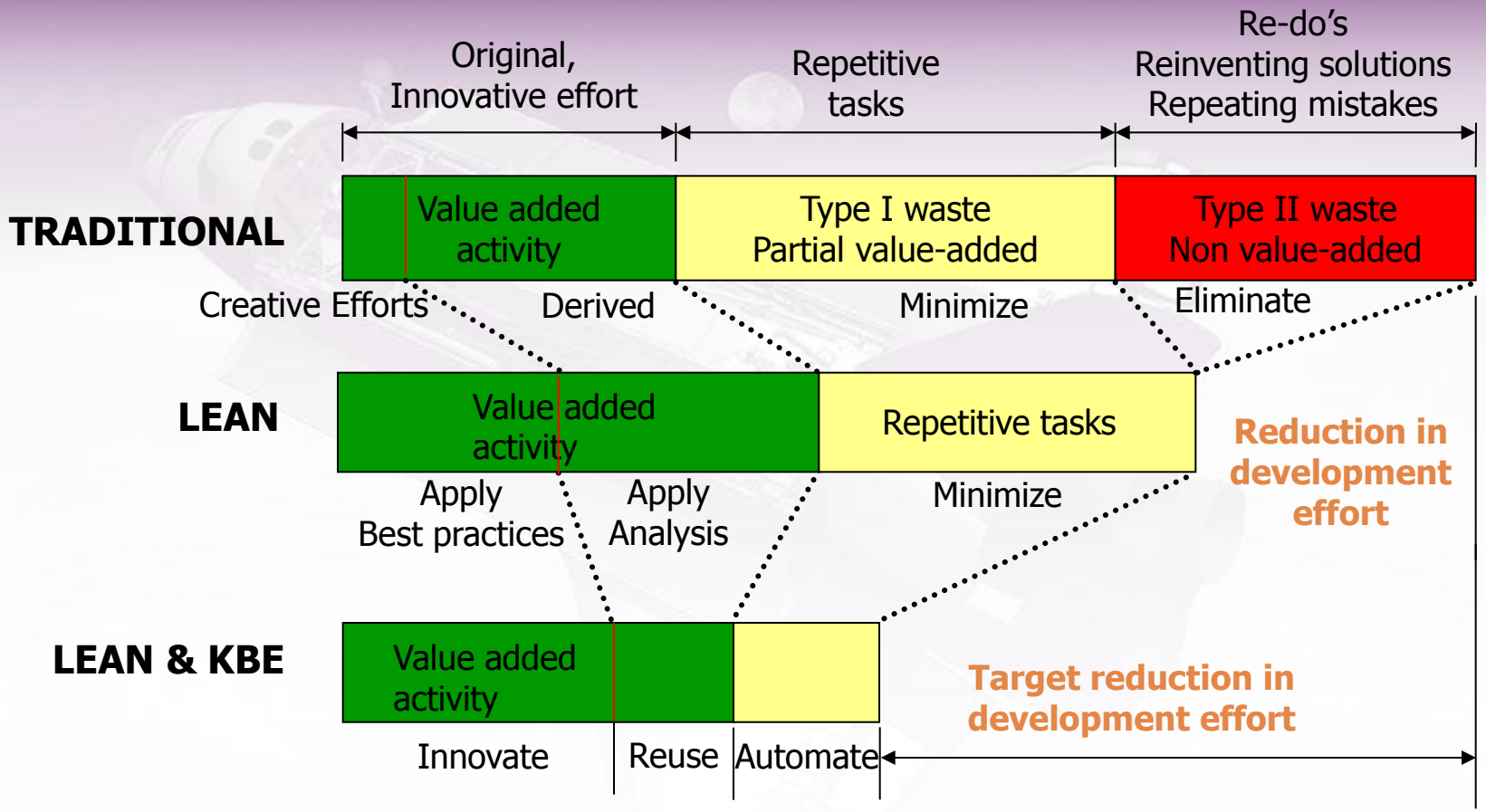


Functional Requirements

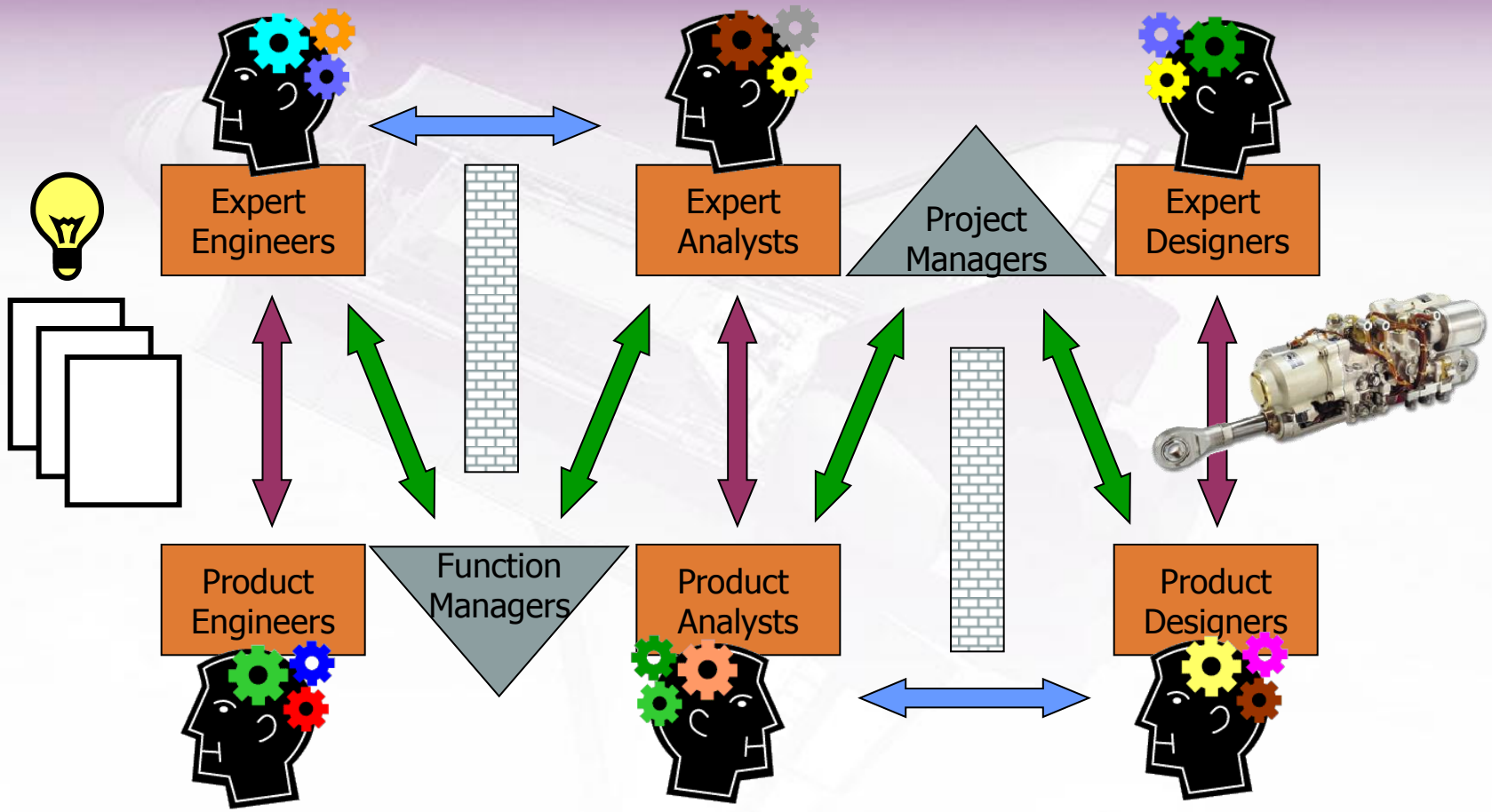
Product Development Process

Engineered Product

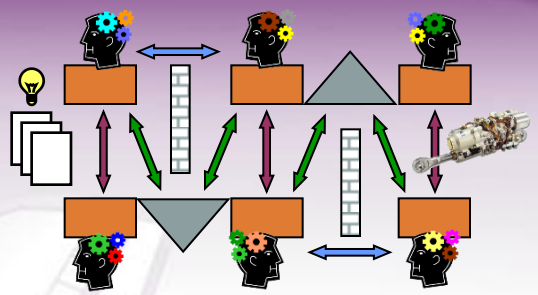
Product-development process



As-is process

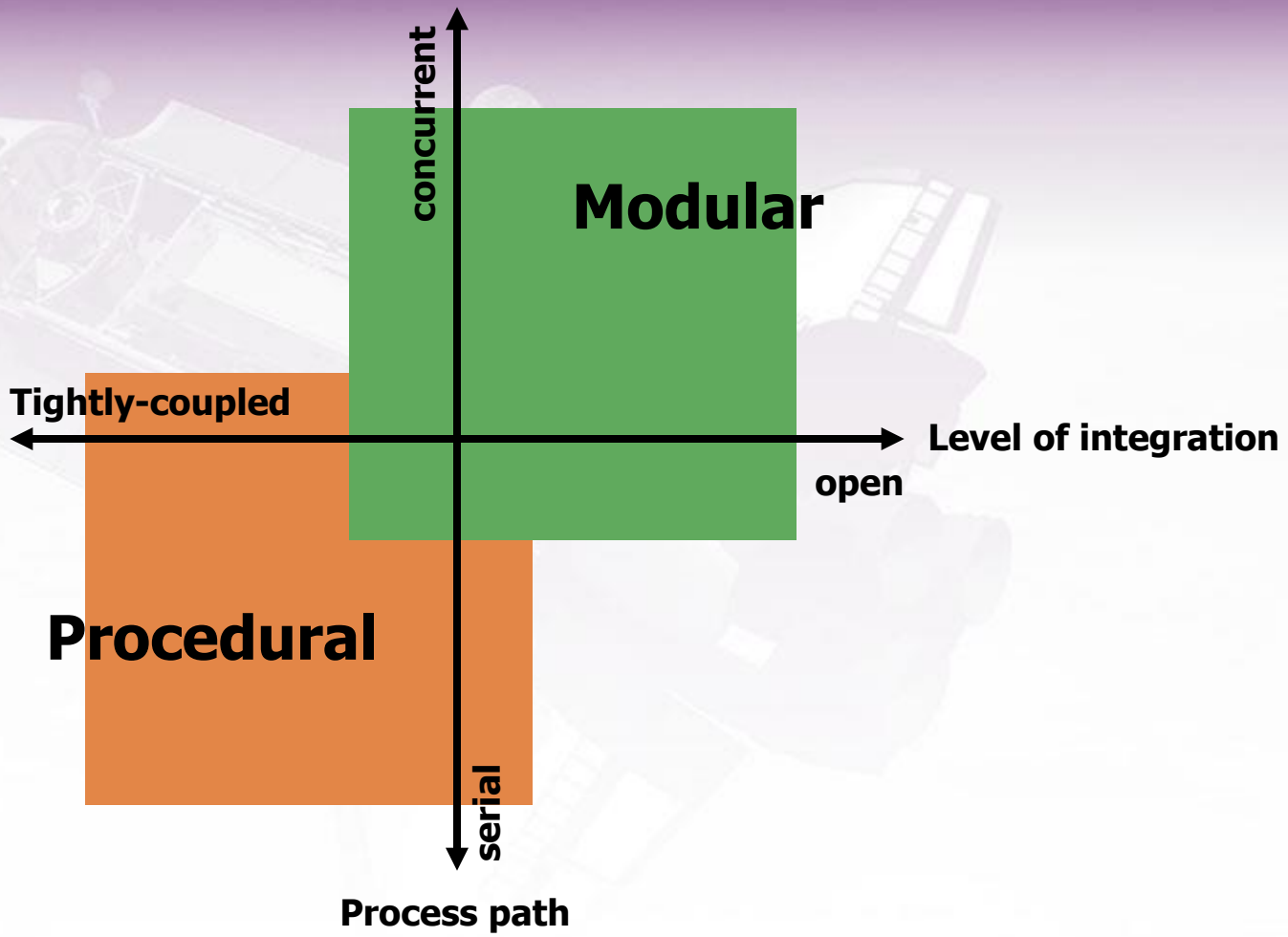


What's wrong with this?

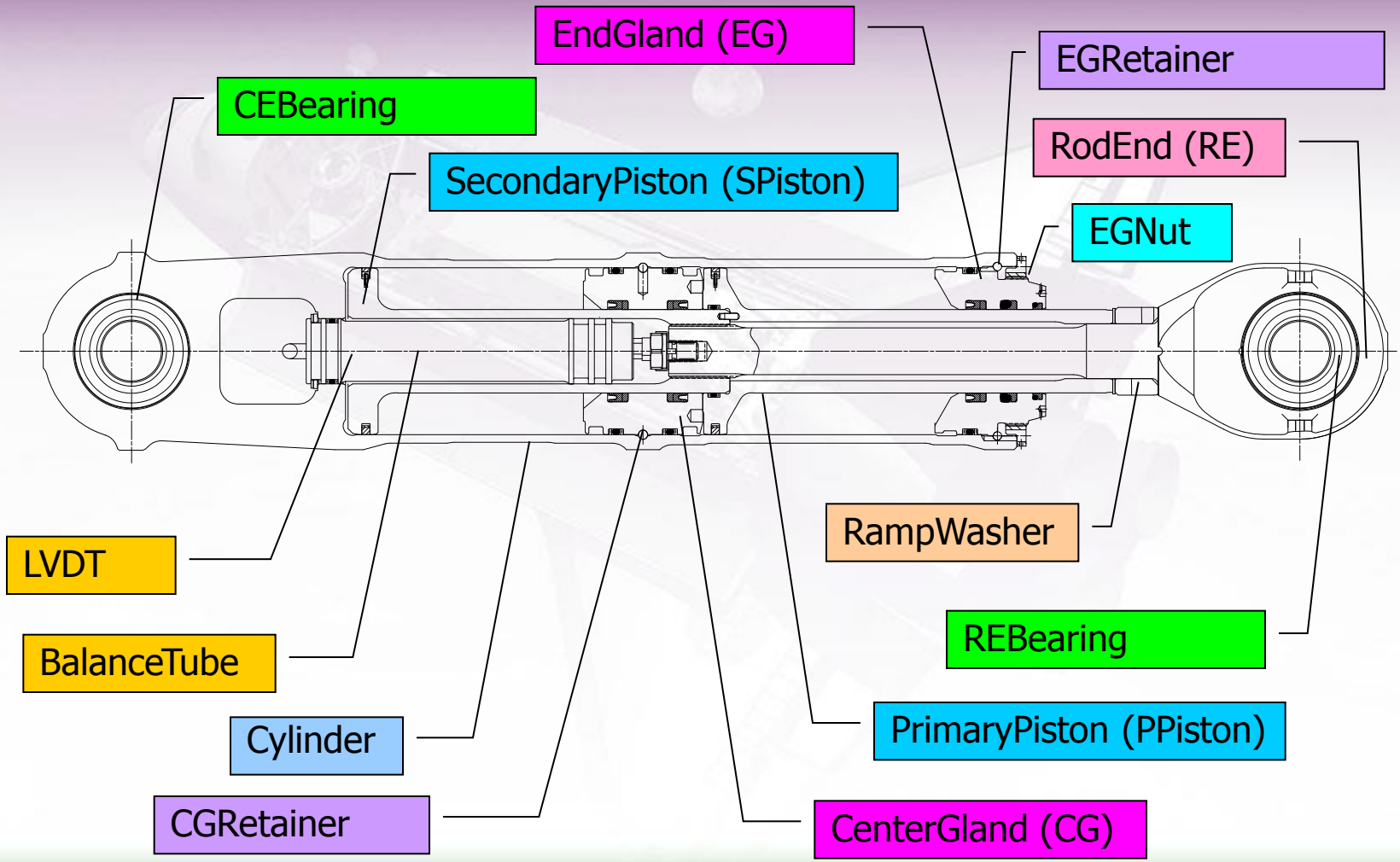


- Knowledge is fragmented
- Subject matter experts (SME) often scarce and busy
- When people retire, information is lost
- Less uniformity and consistency
- Time-intensive, manpower dependent
- Often design is done via trial and error—case-based reasoning

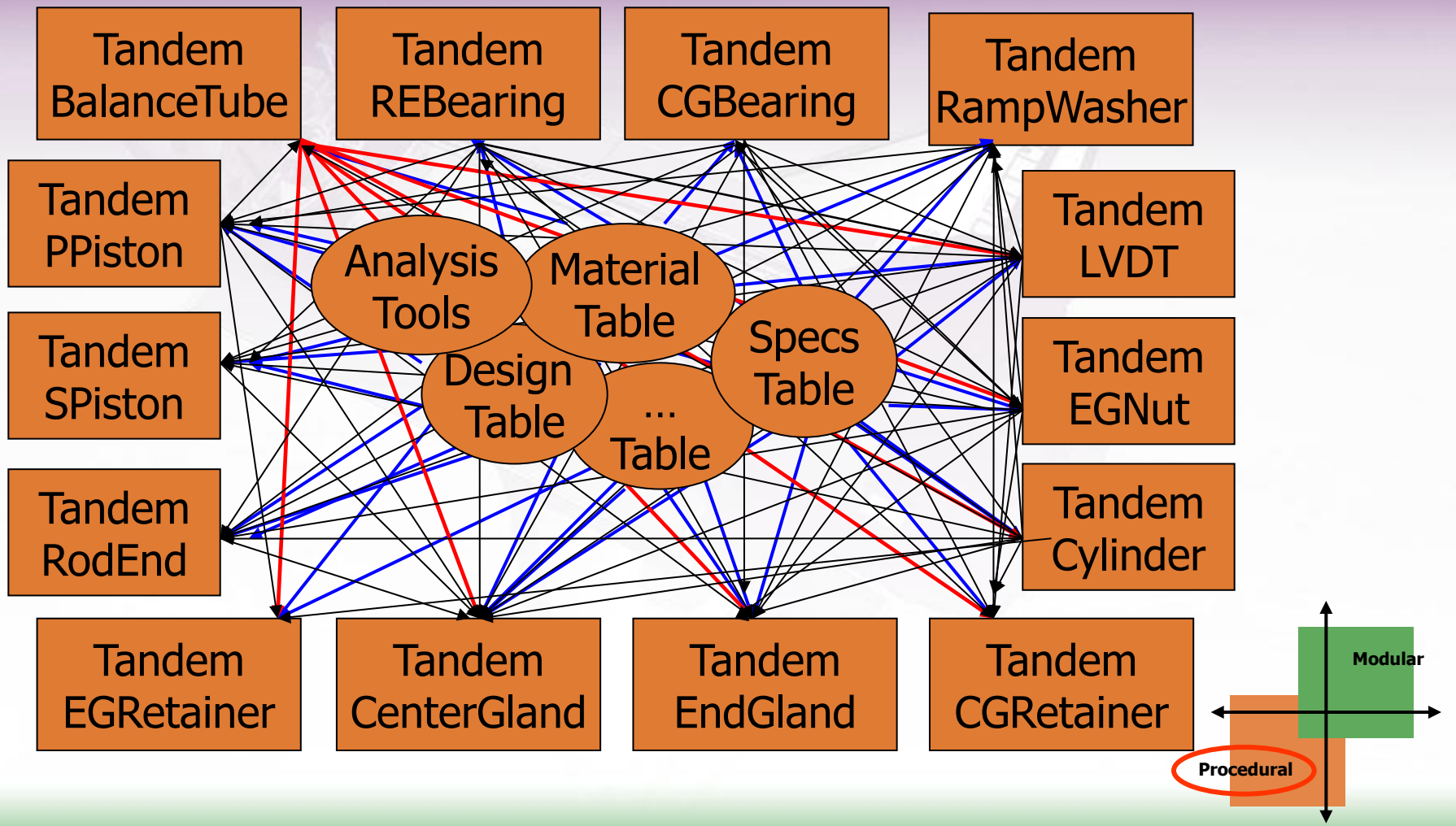
Better, new approach?



Let's consider an example

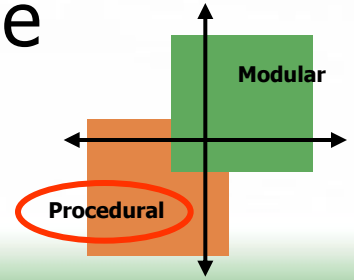


Serial, tightly-coupled KBE system

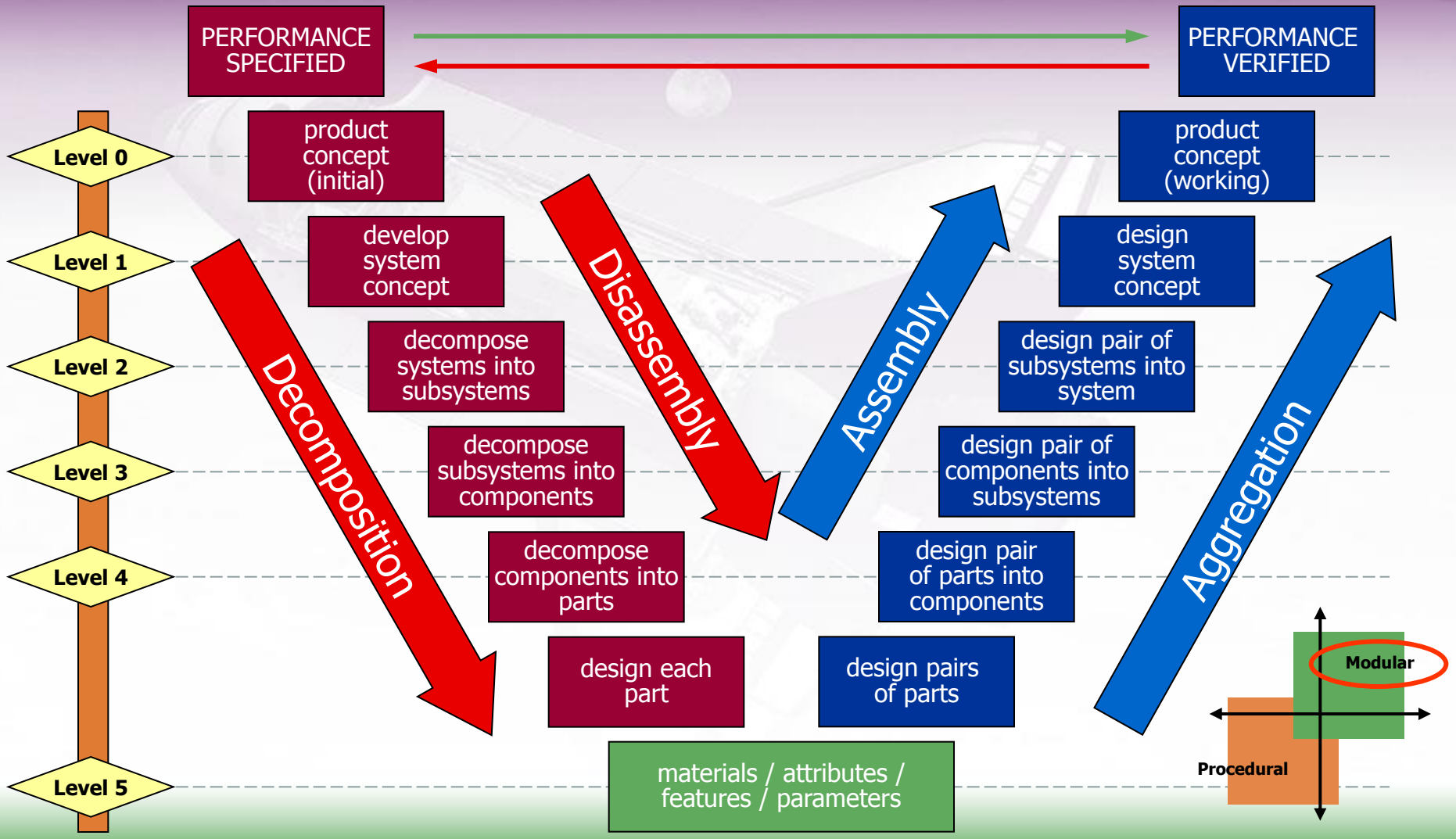


Drawbacks of procedural process

- Part and product specific
- Hard-coded interfaces
- Cumbersome to maintain
- Incompatible API's
- External parameter linking issues
- Very sensitive to interface changes (parameters, rules, features)
- Expansions are complex and error prone
- Inflexible



A system's approach



Open, concurrent KBE system

Product Configurator

Choose a System Configuration

Build System Specs

Choose a Sub-System Configuration

Build Intra-Part Rules & Relations

Choose a Component Configuration

Build Assembly Constraints

Build chosen Component Templates

Build Inter-Component Relations

SmartPart Configurator

Configure An Product

Apply System Specs

Build System Solution

Apply Intra-Part Relations

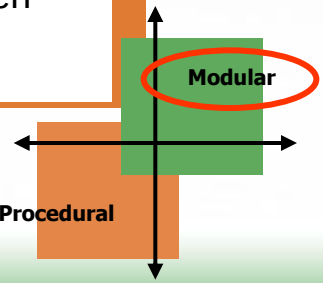
Build Sub-system Solution

Apply Assembly Constraints

Build Component Solution

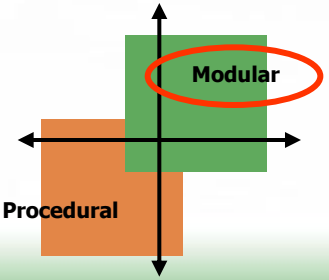
Apply Inter-Component Relations

Instantiate Chosen Component Templates



Merits of modular process

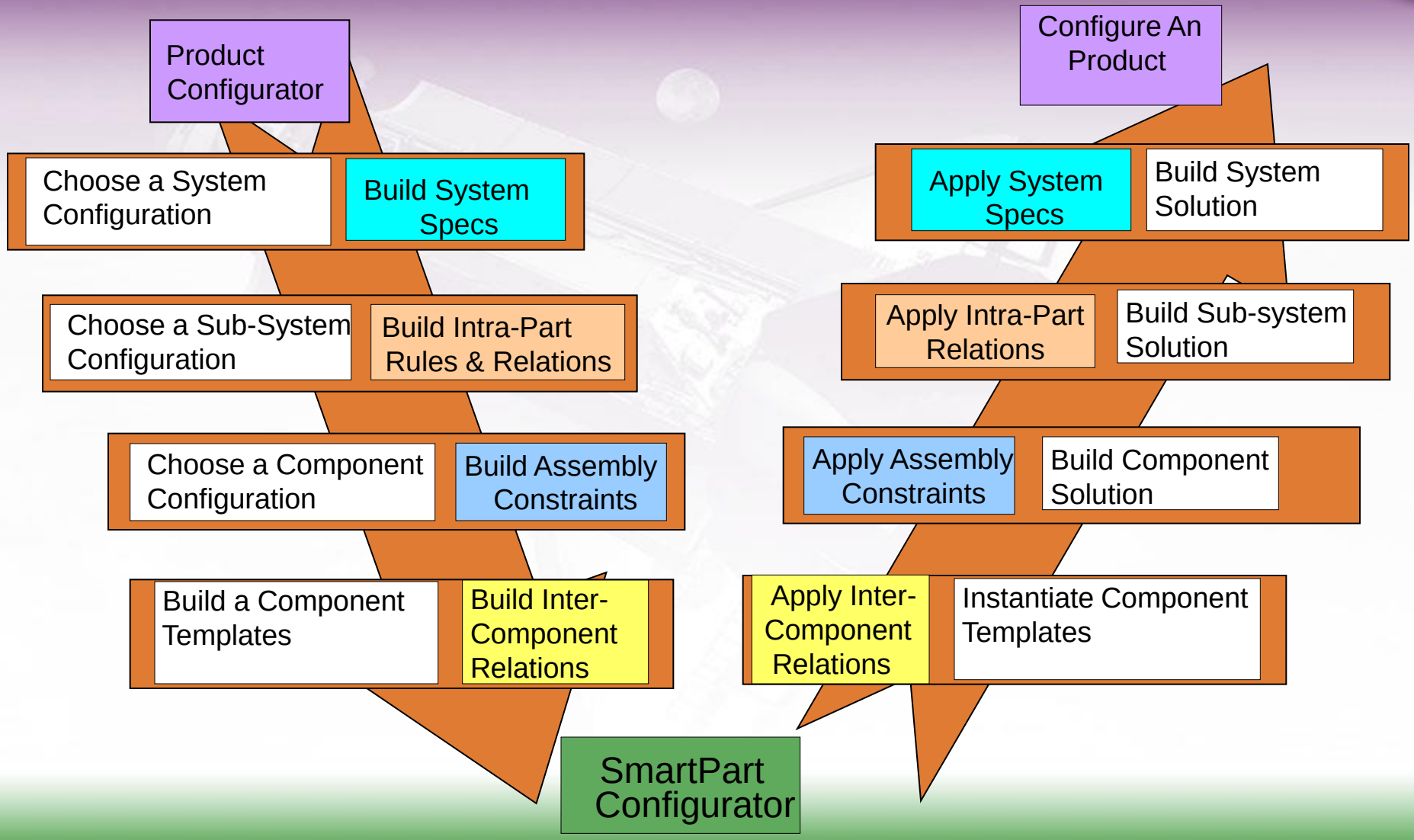
- Product-Independent
  - Architecture
- Part-Independent
  - Concept
- Tool-Independent
  - Method



A CATIA v5 implementation

- System Architecture
  - JustOne system model and common tree structure
- Generative Rule Bodies
  - Rule bodies create more rules dynamically on the tree; asleep until awaken
  - Retrieve templates; no generative geometry
- Internal Linking
  - Two generalized automation methods to pass/exchange information intrapart and interpart

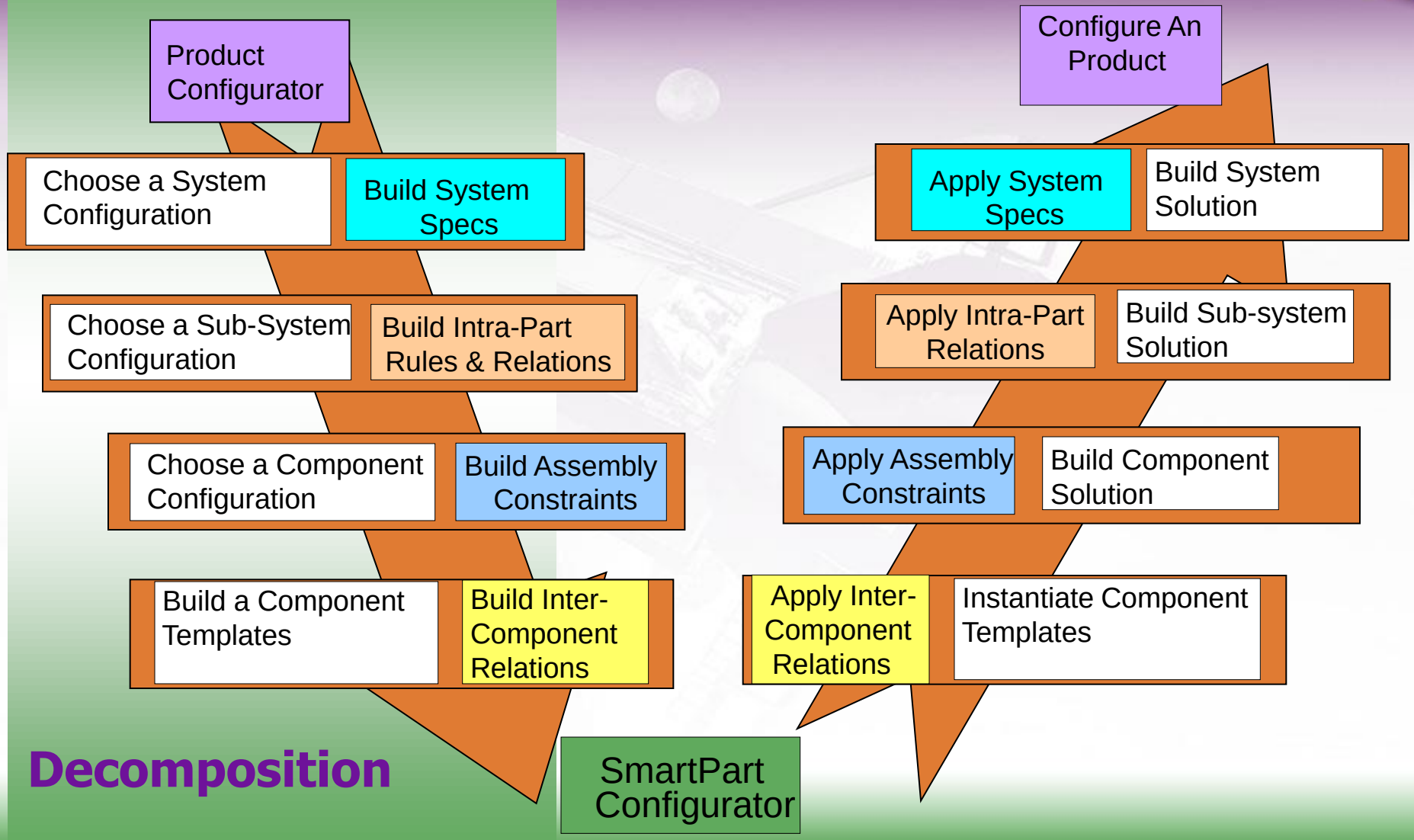
Demo–Applying the concept



Demo–What's in play?

- Two summary Excel spreadsheets (materials and functional requirements)
- Seed file (PKT, GenScript)
- A new CATProduct

Demo–Initialize the tree



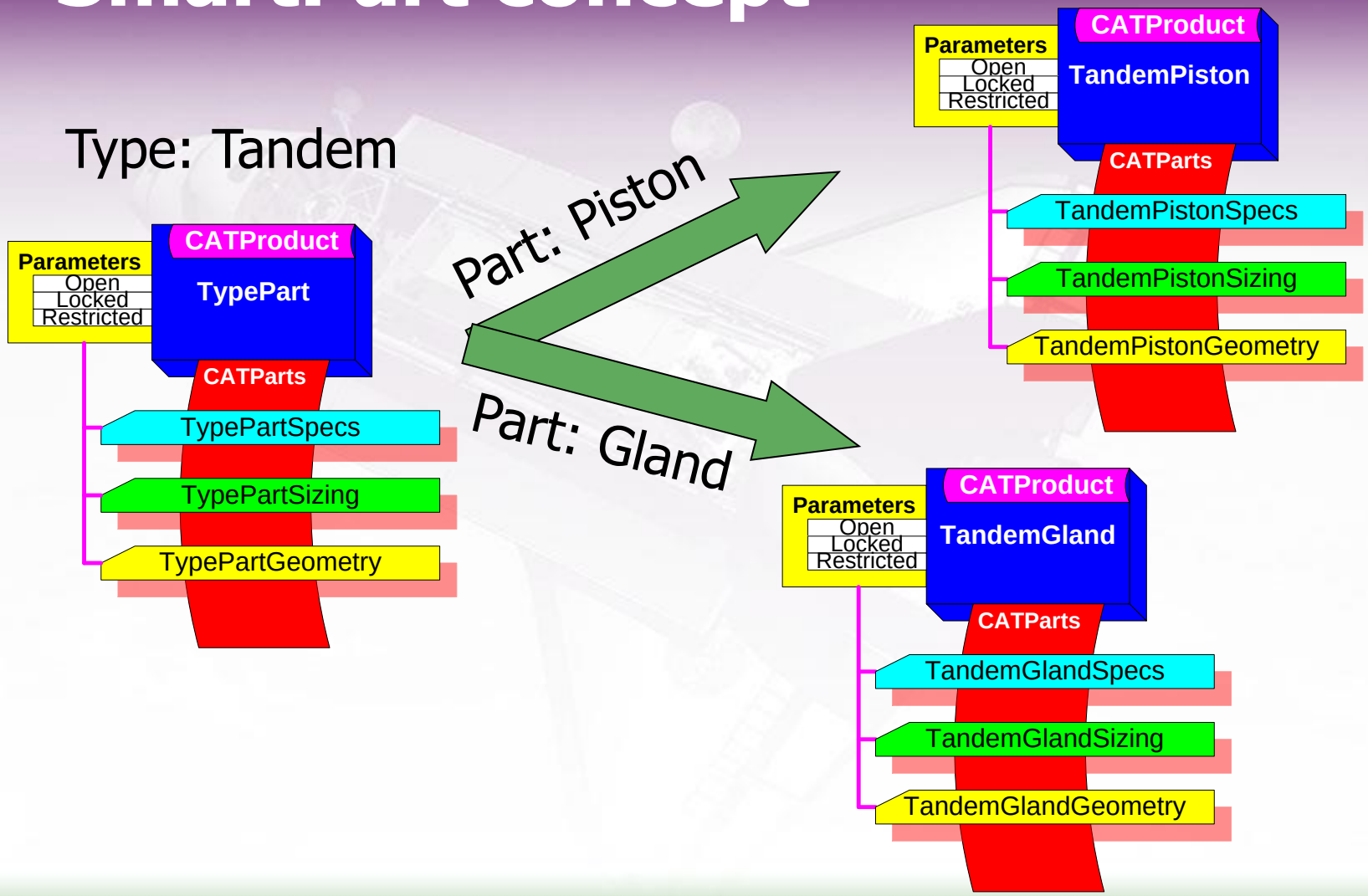
Decomposition

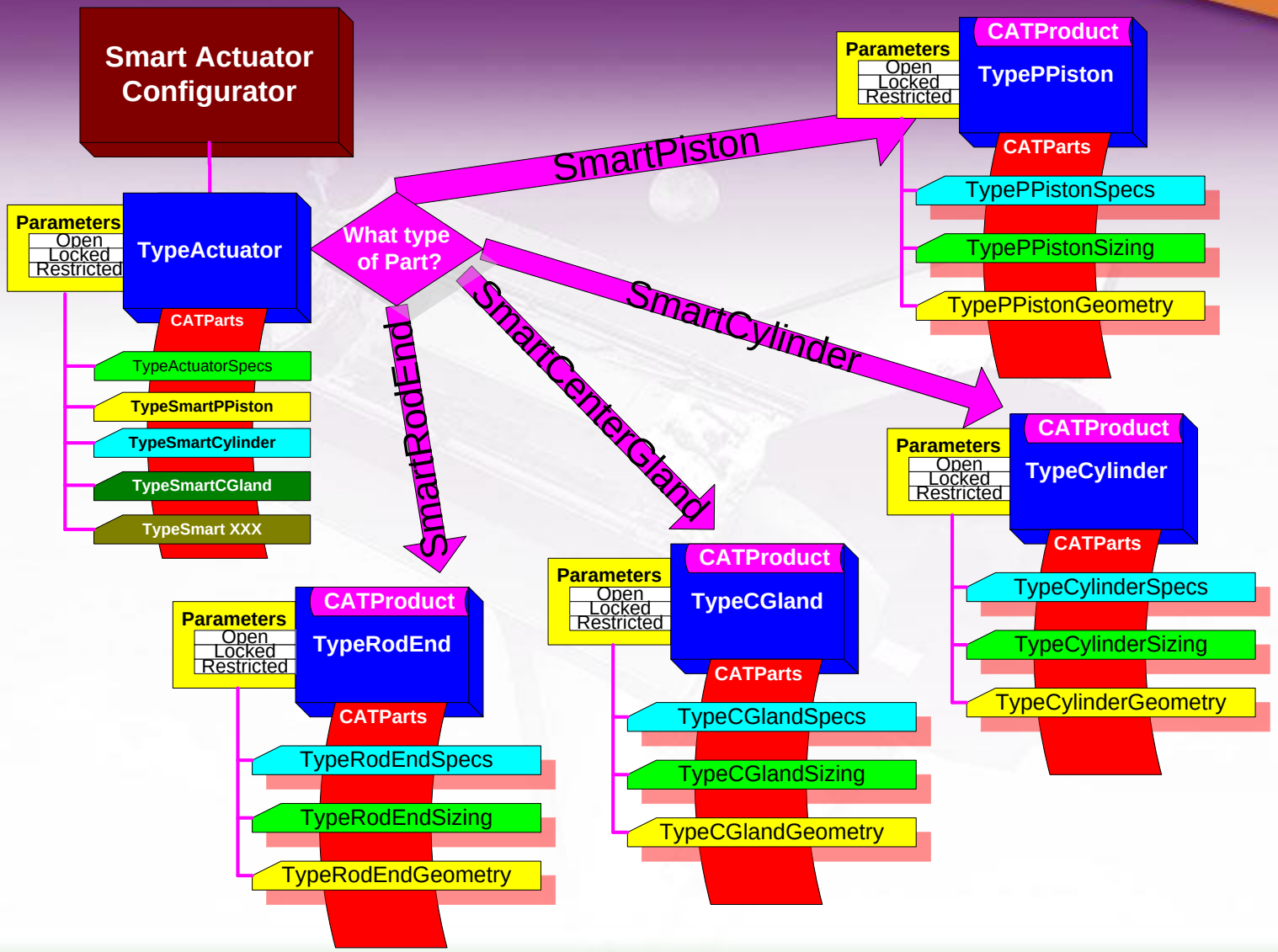

CATIA V5
Start
File
Edit
View
Insert
Tools
Window
Help
100%
None
Parker
CATIA V5 R12
Product1
CATIA V4,V3,V2 Workbench
start
DCM waiting...
CATIA V5R12
CATIA V5
V:\Pilots\Bria...
2 Microsoft...
1:51 PM

Demo–Salient points

- Initialized the system/customer specs
  - Automation
- Used KBE scripting language to construct a reconfigure-able and smart model of the product
  - Automation
- Defined a collection of ready-to-fire rule bodies for reconfiguring part
  - Reusability
- Built "generatively" a product tree
  - Extensibility

SmartPart concept

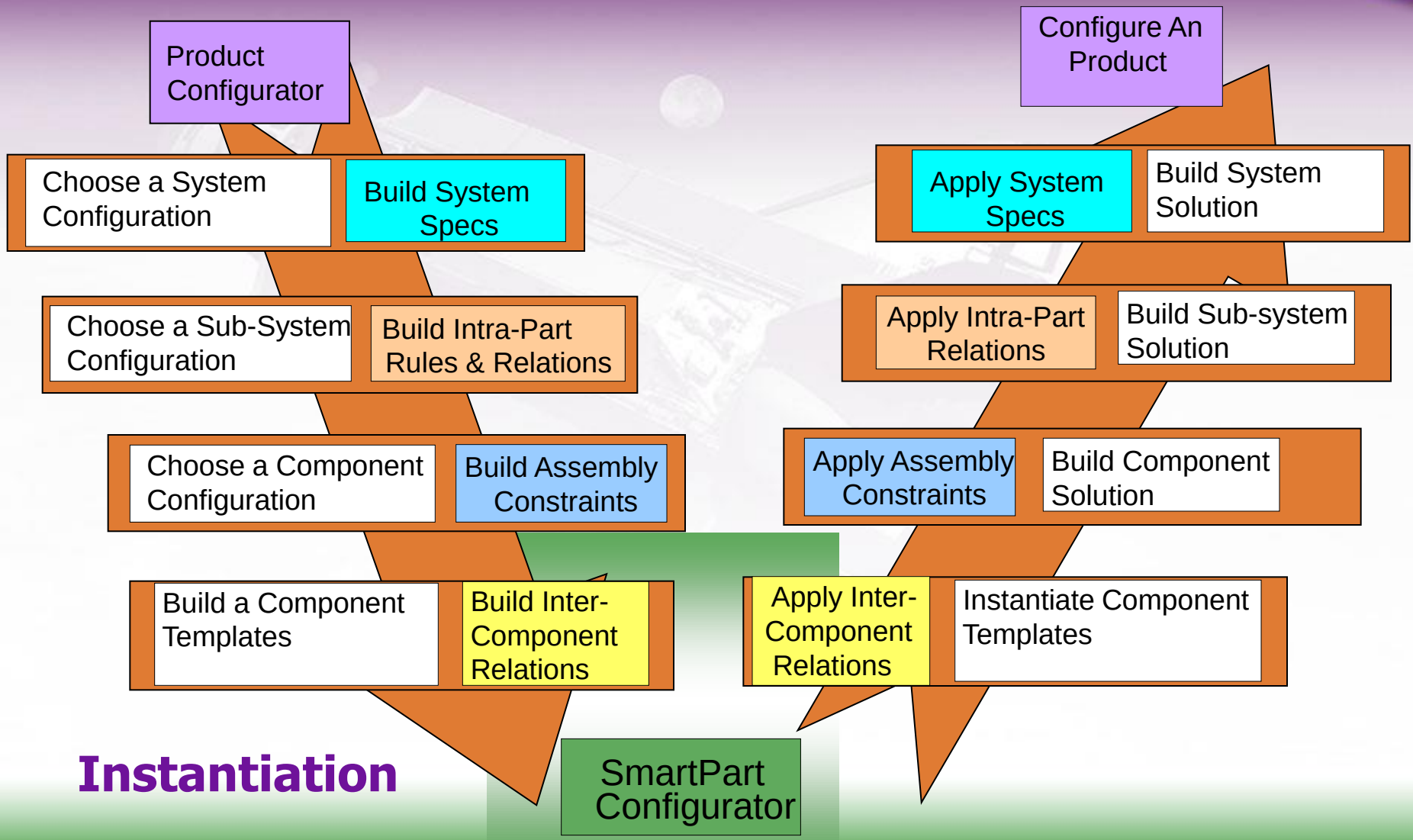


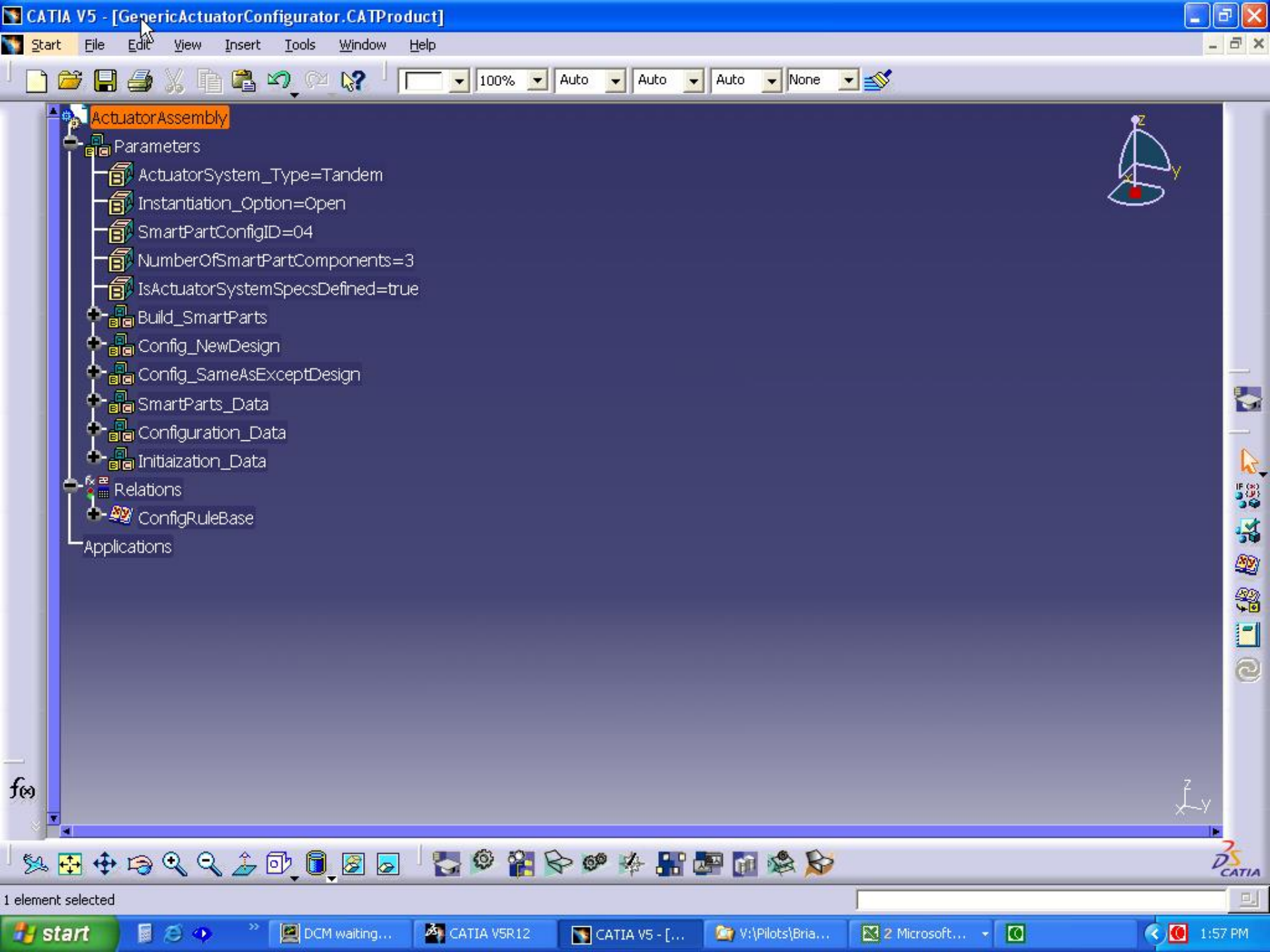

Smart Actuator Configurator
What type of Part?
SmartPiston
SmartCylinder
SmartCenterGland
SmartRodEnd
Parameters
Open
Locked
Restricted
TypeActuator
CATParts
TypeActuatorSpecs
TypeSmartPPiston
TypeSmartCylinder
TypeSmartCGland
TypeSmart XXX
CATProduct
TypePPiston
TypePPistonSpecs
TypePPistonSizing
TypePPistonGeometry
TypeCylinder
TypeCylinderSpecs
TypeCylinderSizing
TypeCylinderGeometry
TypeCGland
TypeCGlandSpecs
TypeCGlandSizing
TypeCGlandGeometry
TypeRodEnd
TypeRodEndSpecs
TypeRodEndSizing
TypeRodEndGeometry

Demo–What's in play?

- Three templates for each part
- Other design tables in background
- A new CATProduct for each part

Demo–SmartParts Creation

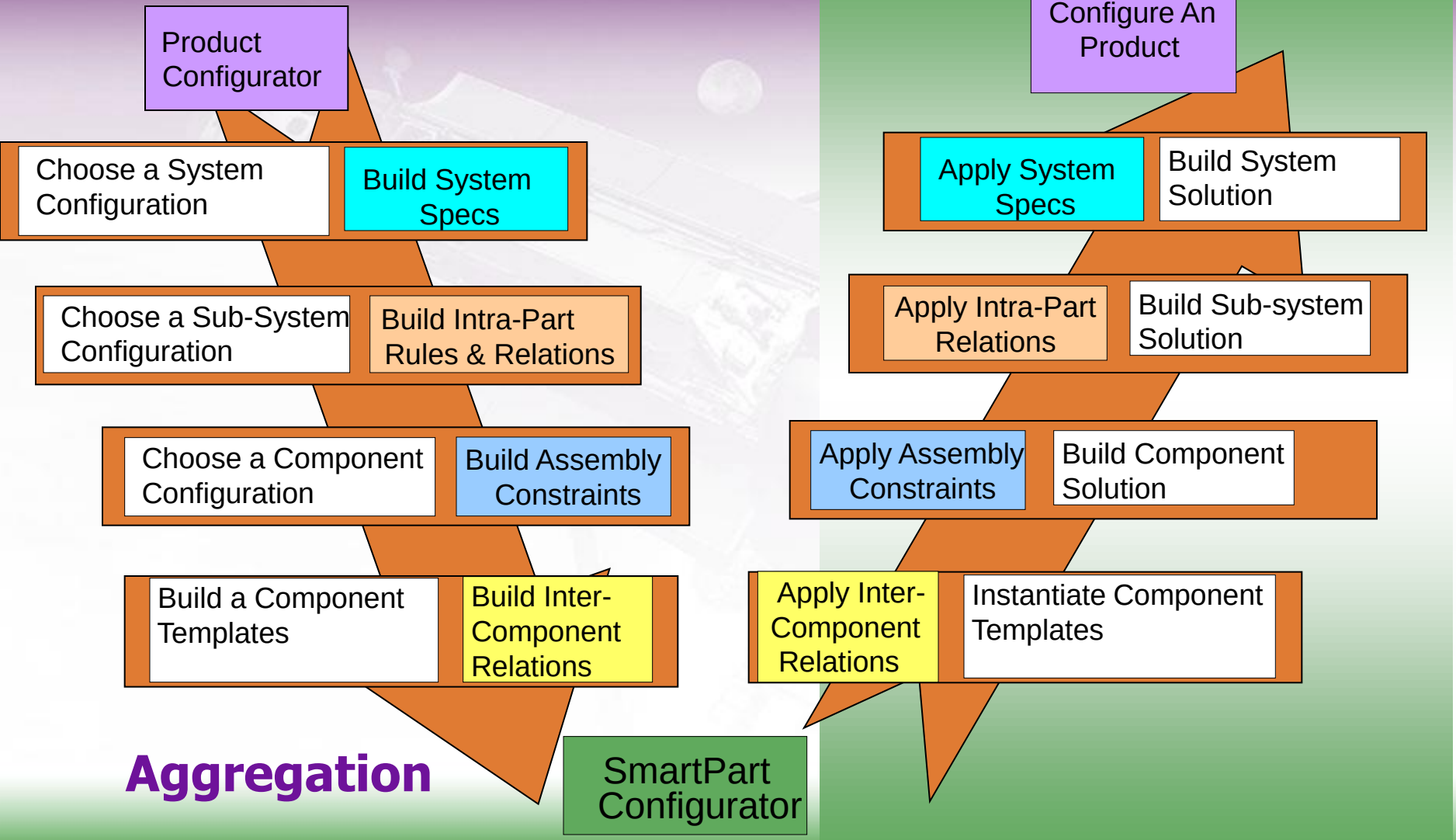


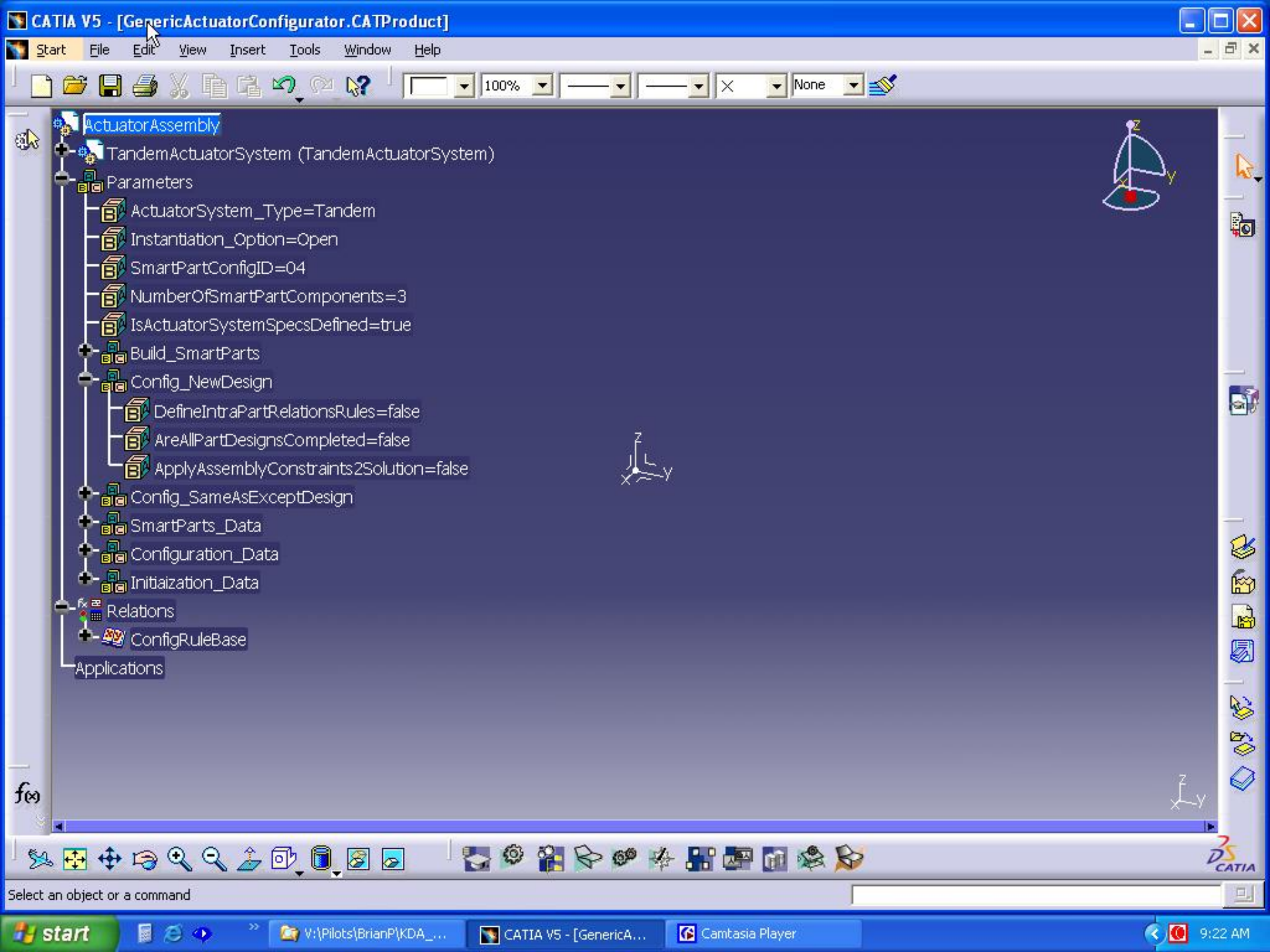

CATIA V5 - [GenericActuatorConfigurator.CATProduct]
Start
File
Edit
View
Insert
Tools
Window
Help
100%
Auto
Auto
Auto
None
ActuatorAssembly
Parameters
ActuatorSystem_Type=Tandem
Instantiation_Option=Open
SmartPartConfigID=04
NumberOfSmartPartComponents=3
IsActuatorSystemSpecsDefined=true
Build_SmartParts
Config_NewDesign
Config_SameAsExceptDesign
SmartParts_Data
Configuration_Data
Initiaization_Data
Relations
ConfigRuleBase
Applications
1 element selected
start
DCM waiting...
CATIA V5R12
CATIA V5 - [...
V:\Pilots\Bria...
2 Microsoft...
1:57 PM

Demo–Salient points

- Defined new rules for creating SmartParts
  - Reusability
- Rules fired to build new product tree
  - Extensibility
- Each product tree has three components of SmartPart
  - Systematization
- Interpart relations were established to bind components of the SmartPart
  - External links eliminated, maintainability

Demo–Configuring For Specs



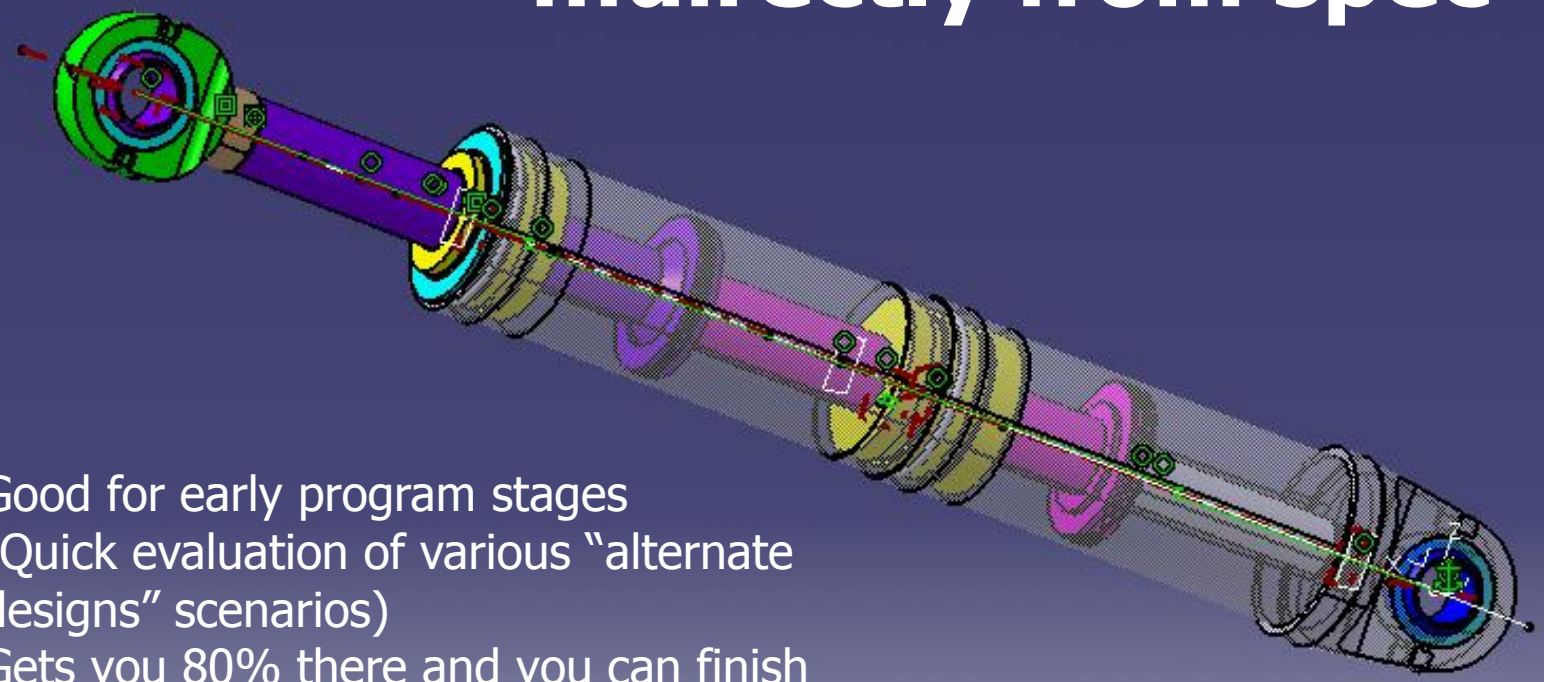

CATIA V5 - [GenericActuatorConfigurator.CATProduct]
Start
File
Edit
View
Insert
Tools
Window
Help
100%
None
ActuatorAssembly
TandemActuatorSystem (TandemActuatorSystem)
Parameters
ActuatorSystem_Type=Tandem
Instantiation_Option=Open
SmartPartConfigID=04
NumberOfSmartPartComponents=3
IsActuatorSystemSpecsDefined=true
Build_SmartParts
Config_NewDesign
DefineIntraPartRelationsRules=false
AreAllPartDesignsCompleted=false
ApplyAssemblyConstraints2Solution=false
Config_SameAsExceptDesign
SmartParts_Data
Configuration_Data
Initiaization_Data
Relations
ConfigRuleBase
Applications
Select an object or a command
start
V:\Pilots\BrianP\KDA_...
CATIA V5 - [GenericA...
Camtasia Player
9:22 AM

Demo–Salient points

- Specs parameters & constraints passed from "systems" to "subsystems", to "components," to "parts" during "decomposition" and vice versa during "aggregation"
- Smart Parts were "instantiated" and constraints satisfied
- Solution is reconfigurable for changing spec requirements

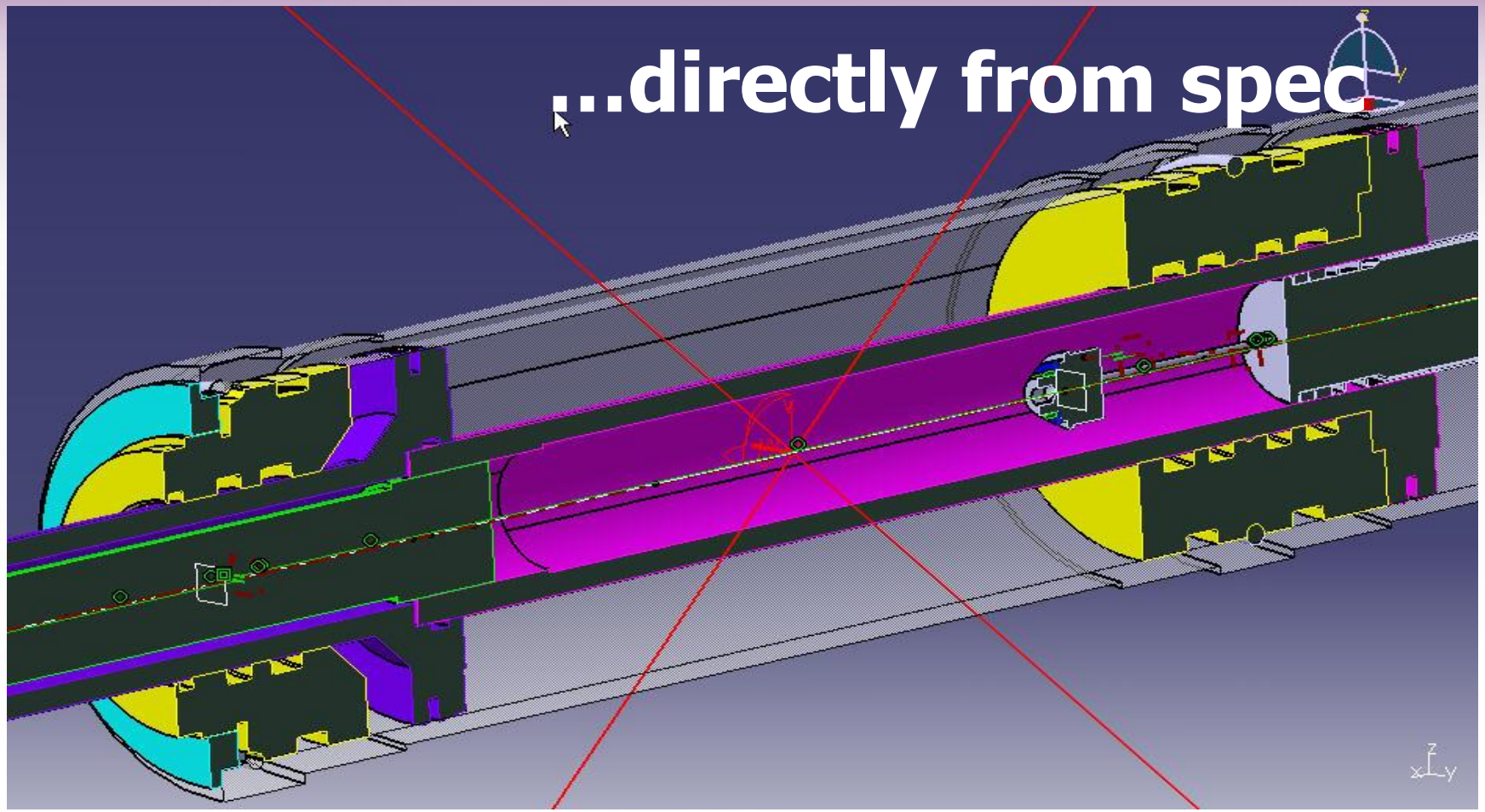
Engineered design…

…directly from spec



- Good for early program stages (Quick evaluation of various "alternate designs" scenarios)
- Gets you 80% there and you can finish the rest (20%) in native CATIA mode

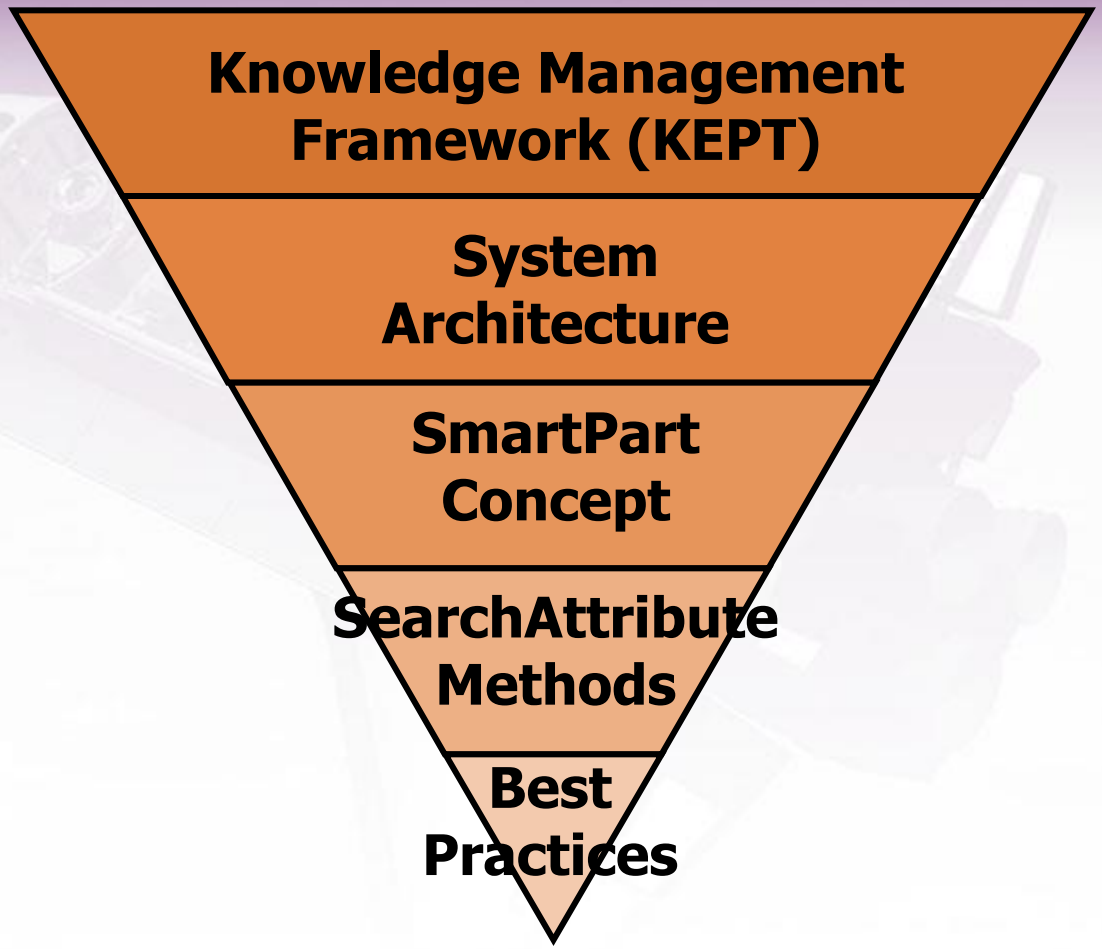
Engineered design…



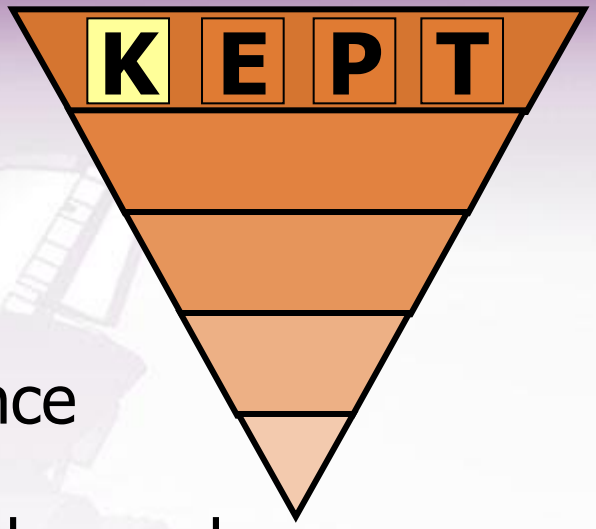
Gaining the most from KBE

- Take a holistic view of your product development needs
- KBE has its own life. Think about integration and interfaces. They are big deal for KBE.
- Employ a modular, open, and concurrent strategy for building KBE systems
- Think engineering centric versus geometry-centric; analysis driven, geometry is a by-product
- Follow a knowledge management framework for applying KBE

Gaining the most from KBE

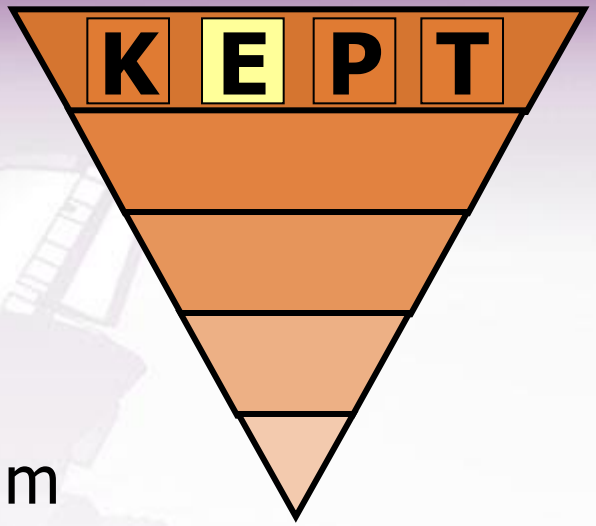


Gaining the most from KBE



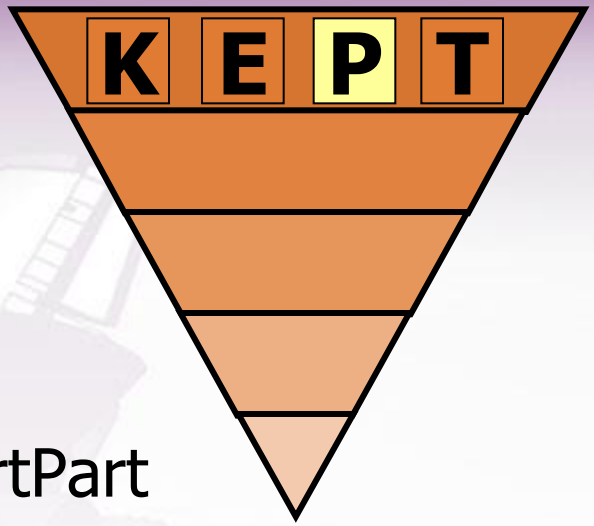
- Leverage Knowledge (K)
  - Capture and maintain intellectual capital
  - Use spreadsheets for inputting specs, material data and rules since interfaces are system-maintained
  - Try not to fragment your knowledge and rules into multiple systems / multiple interfaces
  - Great value in storing your rules & equations in your strategic PLM system

Gaining the most from KBE



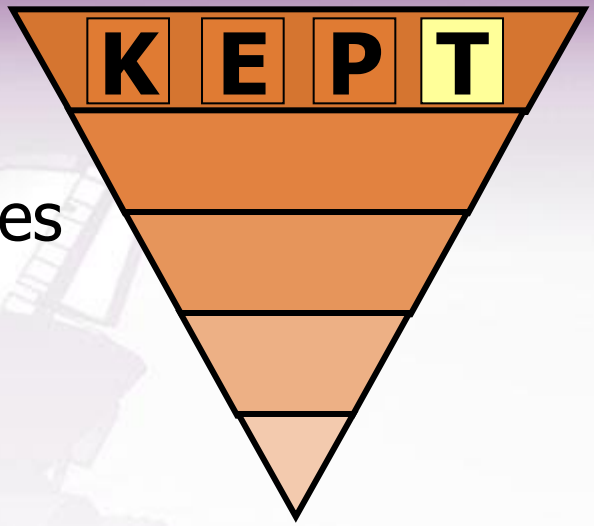
- Engage Enterprise (E)
  - Establish a knowledge sharing culture
  - Educate about KM, KBE, and its benefits
  - Create a cross-functional KBE team
  - Make it easy for SME's to contribute and maintain knowledge
  - Appoint Knowledge Keepers
  - Use proactive promotion for KBE thinking

Gaining the most from KBE



- Develop Automated Process (P)
  - Identify value-streams (e.g., streamline repetitive tasks)
  - Automate to design processes (e.g., Product Configurator, SmartPart Configurator, and others)
  - Develop strategies to minimize interfaces

Gaining the most from KBE



- Apply Advanced Tools (T)
  - Use system engineering techniques
  - Build inside CATIA V5 using Knowledge Advisor (KWA), Knowledge Expert (KWE), and Product Knowledge Template (PKT)
  - Minimize writing version dependent code

Special Credit

Dr. Brian Prasad
and the KDA Team at Parker CSD

Virtual Services
MSC.Software

bprasad@parker.com, jrogers@parker.com
www.parker.com

Questions?

Watch for us on the History Channel's Modern Marvels, May 12, 2004

bprasad@parker.com, jrogers@parker.com
www.parker.com