

April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

When & Where to Use Knowledgeware, Generative Scripts & VB Tools

Brian Prasad¹, Robert Garrison² Jeff Rogers¹ and Christian Isaacs²

¹Parker Hannifin / Parker Aerospace ²Rand Worldwide / Rand Professional Services







Presentation Topics

- Parker Actuator Configurator Redux:
 - Creation & implementation of a generic KBE application that supports *engineer2configure for all possible actuator families*.
 - Rationale for using Knowledgeware tools, scripting and advanced KBE concepts to derive parts from specifications.
- Project Linchpin The User Interface:
 - Leveraging the dynamic nature of the core system architecture.
 - Rationale of using Visual Studio .Net with Knowledgeware.
- Project Challenges:
 - A unified system architecture definition, case-based UI coding, development & deployment throughout.
- Lessons Learned





Founding Partner







A Typical New Product Design & Development Scenario





What's wrong with this?

- Knowledge is fragmented
- Subject matter experts (SME) often scarce and busy



- Less uniformity and consistency
- Time-intensive, manpower dependent
- When people retire, information is lost
- Often design is done via trial and error case-based reasoning





ounding Partner







Parker Configurator: Knowledege-centric approach





A CATIA V5 implementation

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

ounding Partner Procedural



COE 2008 Annual PLM Conference & TechniFair April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

Specs Definitions (Excel Inputs)

		В	С	D		E	F		G		H		I			J	
	1	Constraint Name	Туре	Value (ir	Const	raint Orienta	First Pro	duct Firs	st Publication		Second Pr	Second Pu	ublicat	tion	Co	mpute	
4	2	CY2PP_Axial_Coincidence	Coincidence		CatCst0	DrientUndefine	Cylinder	Cyli	nder_AxisLine		PPiston	PPiston_Axi	sLine		Y		
5	3	CY2REEG_Axial_Coincidence	Coincidence		CatCst0	DrientUndefine	Cylinder	Cyli	nder_AxisLine		REEndGland	REEG_AxisL	Line		Y		
6	4	CY2REEG_Transverse_Parallel	Parallel		CatCst0	DrientSame	Cylinder	Cyli	nder_TransverseLine		REEndGland	REEG_Trans	sverseLi	ine	Y		
7	5	CY2REEG_Contact	Coincidence		CatCst0	DrientOpposite	Cylinder	Cyli	nderNutBoreThread_R	EEndGland_C	ContacREEndGland	REEGMiddle	Ring_C	Cylinder_Conta	actPlaY		
·	6	CY2EGLockNut_Axial_Coincidence	Coincidence		CatCst	DrientUndefine	Cylinder	Cyli	nder_AxisLine		EGLockNut	EGLockNut	AxisLin T	ne	Y	i	
0	/	CY2EGLockNut_Transverse_Parallel	Parallel Contoct		CatUstu NZA	JrientSame	Cylinder	Cyli	nder_TransverseLine	taatEaaa	EGLockNut	EGLockNut	Culinda	erseLine ar ContactEor	Y N		9
9	u q	CV2CEEG Avial Coincidence	Coincidence		DatCet(Drientl Indefine	Cylinder	Cyli	nder_COLOCKINUL_CON	itatii ate	CEEndGland	CEEG Avial	ine (]	
10	10	CY2CEEG Transverse Parallel	Parallel		CatCst0	DrientSame	Cylinder	Cyli	nder TransverseLine		CEEndGland	CEEG Trans	sverseL	ine	Y		
11	N	om_Suppry_Pressure (psi)	4100	J.UUT	41	100.001		4057.001 . CH	160	0.000003	11		3			
12	N	om_PTank_Pressure (psi)		175	5.001	1	150.001		600.001	100	000000.0	24	-				_
13	N	om_STank_Pressure (psi)		100	0.001	1	150.001		600.001 <mark>- SH</mark>	160	600000.0	JI		J			
14	Li	imit_Load_Compression (lbf)	120060	0.001	1800	060.001	E	61600.001	160	600000.0	01	0	3			
15	Li	imit_Load_Tension (lbf)		112140	0.001	1121	140.001	6	61600.001 <mark>3</mark>	160	50000.0	01	0	3	5c4	4a0	
16	UI	ltimate_Load_Compression	on (ll	180090	0.001	1800	090.001	9	32401.001 <mark>(MS</mark>	160	600000.0	01	0	2			
17	U	ltimate_Load_Tension (lb	f)	168210	0.001	1682	210.001	9	32401.001	160	600000 0	11 O	25	3			
18	U	ltimate_Load_Side (lbf)			800		800		400	160	50000.0	11 0	25	25			-
19	Eı	ndurance_Load_Fatigue((lbf)	104400	0.001	1044	400.001	5	51334.001 <u></u>	400	50000.0		.20	2.5			
20	Pı	roof_Supply_Pressure (ps	;i)	6000	0.001	60	000.001		6500.001 2	160	50000.0	JI U	.25	2.5			
21	Bı	urst_Supply_Pressure (ps	i)	10000	0.001	100	000.001	1	10820.001	160	50000.0	0 0	.25	2.5			
22	In	npulse_Supply_Pressure	(psi)	6000	0.001	60	000.001		6100.001 <mark>G</mark>	160	50000.0	D1 0	.25	2.5			
23	Pı	roof_Tank_Pressure (psi)		4000	0.001	40	000.001		1275.001 <mark>G</mark>	160	50000.0	D1 0	.25	2.5			
24	B	urst_Tank_Pressure (psi)		6000	0.001	60	000.001		2125.001 _G	160	50000.0	D1 0	25	2.5			
25	In	npulse_Tank_Pressure (ps	si)	2000	0.001	20	000.001		700.001	160	50000.0	11 0	25	2.0		+	-
26	In	npulse_Load_Cycles_Fati	gue	1000	0.001	10	000.001		5000.001	100	30000.0		.25	2.0			_
27	St	troke_Nominal (in)		9	9.487		9.487		8.592								
28	R	etract_Length (in)			- 24		34.957		33.394								
29	B	earing Friction Coeff Pr	oof		0.15		0.15		0.15 <mark>9</mark>	160	50000.0	01 0	.25	3			
30	B	earing Friction Coeff Bu	rst		0.2		0.2		0.2								
21	D.	aaring Eriction Cooff Ex	41.000		0.1		0.1		0.1	_	-	Found	dion Part	ner	Foundi	an Partner	











April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

Achieving a Generic Product Configurator





April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

Merits of modular process



Product-Independent

- Generative Architecture
- Generic Systematization
- Part-Independent
 - Reusable Templates
 - Pattern Decomposition
 - Tool-Independent
 - General-purpose parameters Interchange **Methods**



Product Solution

Configurator

Modula

r Partner



April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

Shifting Focus to the User...

- CATIA Knowledgeware:
 - Has a wealth of tools.
 - Spartan user interface.
 - Executing the Actuator Configurator:
 - Modular & flexible, but required numerous operations to be executed manually and in a specific sequence.
- Requirements for a good user experience:
 - Make it easy to use effectively.
 - Make the experience of using the application enjoyable.
 - Failsafe the architecture to eliminate user mistakes.
- Parker Configurator--UI Needs
 - JustOne Interface and seamless dynamic interaction with the tree, Knowledgeware parameters and generative rules (no hard coding).











April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

...While Leveraging the Original Investment





Development of a User Experience (UX)





UX Development Tools for use with CATIA KBE

- Visual Basic for Applications (VBA):
 - Good for smaller internal deployments.
 - Does not allow development of processes that run outside of CATIA.
- Visual Studio 6:
 - Allows development of application processes outside of CATIA.
 - Architecture is old.
 - Code is unmanaged.
 - UX design tools are not modern.





Founding Partner







UX Development Tools for use with CATIA KBE

- Visual Studio .NET:
 - VS2003 1.0 Framework
 - Suitable, but development tools were already being deprecated.
 - VS2005 2.0 Framework
 - Targeted development environment for this project:
 - Out-process capable.
 - Exhaustive component libraries for form design.
 - Robust development environment.
 - VS2008 3.0 / 3.5 Framework
 - Not on the market at the time of this project.













- Leverage a modular architecture and case-based coding methodology (~8,000 lines) for UI in order to give user a better experience.
 - Provide hooks to the existing triggers:



• Facilitate template access & table use:





• Flow with Run Time operations & hook to Reporting:



- UX coding approach:
 - Modular architecture and a case-based coding methodology. Frequent dynamic Interactions with tree.
 - Tools and forms reusable for future projects.
 - Limited to manipulating already extant KBE features in CATIA.













User Help

Feedback on Progress

COE 2008 Annual PLM Conference & TechniFair April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

Actuator Configurator UX Development

STEP 1: Type	STEP 1: Type	
New Select this Option if you are starting from the seed models (generic CATPart templates). Existing Select this option if the SmartParts have already been created and you wish to continue inputs (i.e. you wish to continue inputs (i.e. you wish to continue inputs (i.e. you wish to continue of the configuration parameters) or if you wish to run more iterations.	 New ○ Existing With minimal effort, this form & its 	
STEP 1: TYPE STEP 2: BASIC PARAMETERS STEP 3: ENGINEERING SPECS STEP 4: MATERIAL SPECS	underlying code can be adapted to any KBE-based equipment (SmartParts) deployment built in CATIA V5.	User Interaction of Program Output
STEP 5: SMARTPART WIZARD		
STEP 6: BUILD ACTUATOR		
STEP 7: CONFIGURE ACTUATOR STEP 8: OUTPUT REPORT		
STEP 9: COMPUTE WEIGHT		
STEP 10: OUTPUT FINAL SOLUTION	Accept	





- Enterprise challenges:
 - CATIA Knowledgeware is built on a foundation of simple syntax that most engineers can master.
 - Facilitates development of small or large KBE tools.
 - Many companies have very few or no software engineers conversant with CATIA Automation methods.
 - Simple VB Scripts built with the assistance of macrorecording might be the limit of Automation capability.





ounding Partner







- Modular and still ~ 8,000 lines of code?
- Code development breakdown for this project:















• Code reuse breakdown for this project:







Founding Partner





April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

Actuator Configurator UX Development







Founding Partner







- Modular code by modular developers:
 - Content was already present.
 - Ran independent of UI.
 - Parker designed the KBE tools used.
 - Tools were embedded in CATIA V5.
 - Form was created to support access of Content.
 - Behavior was created to interface Form with Content.
 - Developer: Christian Isaacs (Rand Worldwide)
 - ~ 6 month hiatus ensued between first release and commencement of final release.





Founding Partner







Form Behavior Content



- Modular code by modular developers:
 - Project handed off to second developer.
 - Minimal dialog between developers.
 - If done properly, code says it all.



- Behavior was extended to meet project changes.
 - Developer: Robert Garrison (Rand Worldwide)
- Code hand-off to Parker for further reuse on this and other projects.
 - Internal Software Engineer received a half-day code review from second developer.





Founding Partner

Content



Form

Behavior





- In retrospect:
 - Content is within reach of any user.
 - Developed & deployed solely in CATIA.
 - Exploits rich Knowledgeware toolset.
 - Form & Behavior are also easily within reach.
 - Many .NET Express Edition tools are free from Microsoft, so even small companies have access.
 - Skilled partners can develop the modular and extendable architecture that you then carry forward.
 - Training in Automation of CATIA is also readily available through a variety of sources.
 - Good option for projects of smaller scope & for proving out the benefits.







Form

Content

Behavior

April 27-30, 2008 • Walt Disney World Swan and Dolphin • Lake Buena Vista, Florida

C 0

Actuator Configurator UX Development

ASPC Beta MACTUATOR CONFIGURATOR 🔊 20 E ASPC Beta STEP 2: BASIC PARAMETERS STEP 2: Basic Parameters ACTUATOR CONFIGURATOR 🙋 Working Directory Working Directory: C:\Temp 00 A directory on your local Actuator Type: FFamily × workstation, where all CATParts, templates, design tables, & Excel Instantiation Option: Open ~ STEP 5: SMARTPART WIZARD STEP 5: Confirm Entries files will be copied to and then saved. Configuration ID: 15 **Confirm Entries** Actuator Type Engineering Specifications Return to Engineering Specs In this step, you have the Defines a specific design opportunity to review the inputs Config ID Type 15 configuration of an Actuator that you provided during Steps 2-Family (e.g. Hydraulic Simplex or 4 and to modify those inputs, if Actuator Name Eclipse_PitchTrimActuator_Fake Tandem solutions). you wish Actuator PID 358XXX Instantiation Options: To make modifications, click on Open either the Return to Engineering ActuatorSystem_Type FFamily All IP knowledge, formulas, Specs text area or the Return to equations, sizing and design Material Specs text area, as ActuatorConfig Type SingleOutput rules are exposed. desired. Once you click on one of BSCatalog Selection NookTable MM these text areas, you will be STEP 1: TYPE returned to that step, where you LMDeviceType BallScrew may make changes and then proceed forward to the current STEP 2: BASIC PARAMETERS RMDeviceStyleHeading PlanetaryGear Train step again. STEP 3: ENGINEERING SPECS If you are satisfied with your Material Specifications Return to Material Specs STEP 1: TYPE STEP 4: MATERIAL SPECS Temperature_Operating STEP 2: BASIC PARAMETERS Part Name Material_Type All (Fdea) STEP 5: SMARTPART WIZARD Nook Ball Screw LOW-ALLOY STEEL, 125 KSI HT, ALL WROUGH FORMS 160 10 STEP 3: ENGINEERING SPECS STEP 6: BUILD ACTUATOR NookBallNut LOW-ALLOY STEEL, 125 KSI HT, ALL WROUGH FORMS 160 10 STEP 4: MATERIAL SPECS 10 STEP 7: CONFIGURE ACTUATOR Retract Stop LOW-ALLOY STEEL, 125 KSI HT ALL WROUGH FORMS 160 **STEP 5: SMARTPART WIZARD** ActuatorSystem NOT APPLICABLE 160 10 STEP 8: OUTPUT REPORT Empty NOT APPLICABLE 160 10 **STEP 6: BUILD ACTUATOR** STEP 9: COMPUTE WEIGHT ACTUATOR STE The benefits become clear, when the ORT > project rolls out to production. Create SmartParts AL SOLUTION Cancel Founding Partner Founding Partner



Lessons Learned

- Maintaining a generative structure and dynamic creation of knowledge at run-time helps reusability and promotes generality of applications.
- Consider a similar structure for the User Interface when deploying Knowledge Systems.
 - CATIA's KBE interface is not intuitive for users.
 - Too many interactions required by the user can result in non-use by those who would benefit most.
- When deploying complex systems, commit to an agile architecture for UI.
 - Use modern Development Environments & schemas:
 - .NET, XML, etc.
 - Design the User Experience in tandem with the system.







Lessons Learned

- Make the system flexible and portable.
 - Strive to develop projects that minimize or eliminate redundant coding.
 - Strive for generic classes that can be reused.
 - Resist the temptation to build KBE features with your UX.
 - Keep Contents separate from Form and Behavior.
- A well structured project presents few problems in expansion or adaptation.





Founding Partner







Questions?

• For more information, contact:

Brian Prasad

Leader, Knowledge Engineering Team Parker Aerospace, Control Systems Division Irvine, California bprasad@parker.com

Robert Garrison

Solutions Architect Rand Worldwide, Rand Professional Services Seattle, Washington rgarrison@rand.com









