

AIP SmallTalk to C#.NET Conversion

Issue Inquiry [X]

Search Criteria

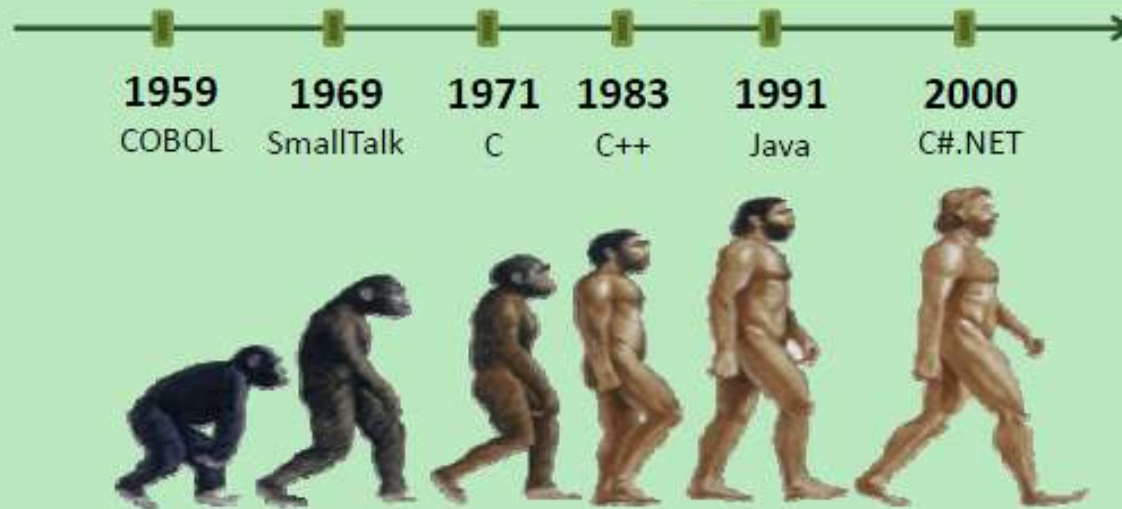
Issue ID: ID Context: Issue Name:

Issues

Issue ID	Issue ID Context	Status	Issue Name

Legacy AIP application in SmallTalk

- Current environment hinders creation of new features
- MicroFocus COBOL and CICS components present licensing issues
- Current modal form branching design makes navigation difficult
- Excellent asset model mapping can be extended and reused
- Replace legacy system with a new architecture in C#.NET



Data



View



Controller



Data

Data Access Layer

- ☐ Create, Read, Update, and Delete AIP table rows
- ☐ Return additional metadata from tables and configuration
- ☐ Use test-driven development to let these methods evolve *without model classes*



View



Thin Client



- ☐ Client requests page templates for a workflow
- ☐ Client receives form views with data-bound controls
- ☐ Search requests populate existing data and lists
- ☐ Simple validation rules are enforced on the client
- ☐ Other rules trigger additional server requests
- ☐ Navigation is driven by standard client interface rules
- ☐ The client maintains current state information
- ☐ Coded units test can replace and mimic client data entry

Controller

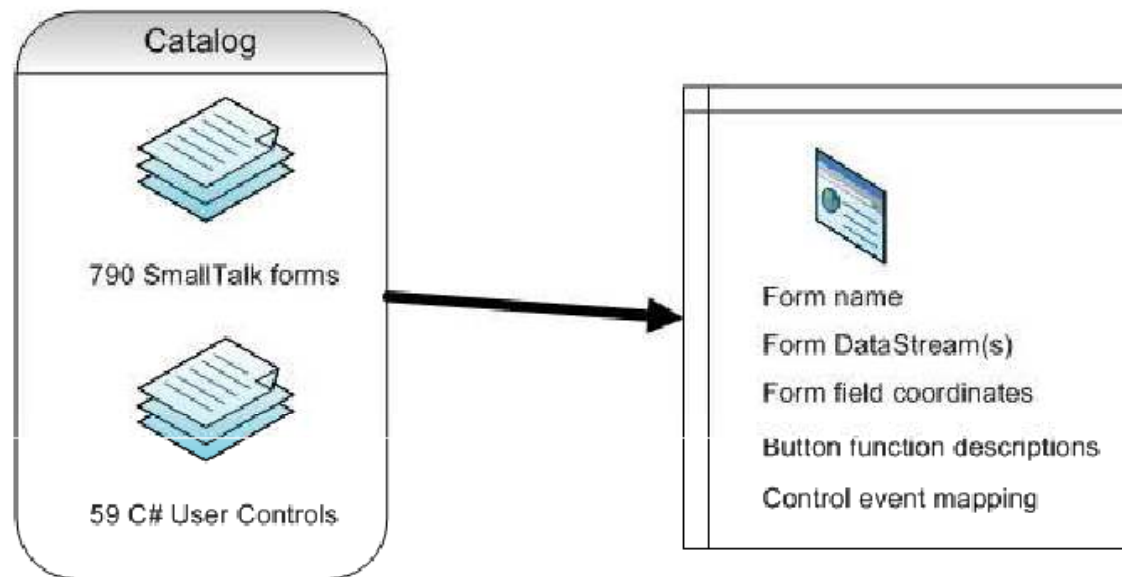


Data and Presentation Marshalling Engine

- ☐ Controller receives both full page and local event requests via WCF
- ☐ Form template information exists as view layout metadata linked to form event methods
- ☐ Generic template control types and coordinates are sent in an ADO.NET table
- ☐ Requested data stream fields are provided for template controls in ADO.NET tables
- ☐ A WPF platform adapter renders ADO.NET form layout and data stream field values as XAML
- ☐ Data manipulation, validation and navigation rule requests trigger server methods
- ☐ Controller returns new rendering or rule responses
- ☐ Test-driven development can ensure general data contract and layout component results
- ☐ Event rules can also be unit tested through code.



Creating the Form Catalog



- Use coded methods to create catalogs
- Discover form control coordinates
- Show field layout in editable documents
- Map each control's DataStream field source
- Group forms into workflow categories
- Prioritize workflows

Control Mapping

Win1446862			-263	-32				
+								
ID	Type	Text	X	Y	Width	Height	TabOrder	Parms
1446862	Panel	Model Definition	0	0	587	442	-1	
332744	GroupBox	Template	141	28	430	61		
398262	GroupBox	Start Date/Time -----	242	142	330	47		
398270	GroupBox	Selected Tasks	20	211	234	185		
725952	GroupBox	Available Tasks	343	211	229	185		
398274	Label	Type:	148	43	42	15		
463794	Label	Template Name:	261	43	185	15		
725982	Label	Model Id:	20	44	65	15		
332742	TextBox	Model Id:	20	59	104	23		mapTo=MODL_ID
398266	TextBox	Type:	149	59	106	23		mapTo=MODL_TYP
398172	TextBox	TemplateName:	261	59	272	23		mapTo=TMPLT_MNEM
922310	Label	Asset Name:	20	91	345	14		
332702	Label	Use for all types	441	104	140	18		
123	CheckBox	Use for all types	421	104	16	16		
398260	TextBox	Asset Name:	20	105	373	23		mapTo=MODL_NME
398256	Label	Description:	20	133	83	12		
332684	TextBox	Description:	20	147	214	53		mapTo=MODL_DESC,Multiline=true
398168	TextBox	Start Time:	250	159	155	22		mapTo=START_TMS
529234	TextBox	End Time:	410	159	155	22		mapTo=END_TMS
332704	DataGridView	Selected Tasks	26	226	221	134		
398178	DataGridView	Available Tasks	349	226	216	161		
332688	Button	< Add	262	242	74	25		
332644	Button	<< Add All	262	271	74	25		
463722	Button	Remove >	262	304	74	25		
463720	Button	Remove All >>	262	338	74	25		
332696	Button	C&hange...	26	365	65	25		

Model Definition - NEW

Model Id: Template Type: Name:

Name: ☐ Use for all types

Description:

Start Date/Time: 09/26/2012 12:00 AM End Date/Time:

Selected Tasks:

Id	Req	Name

Change... Req Task Occurrences...

Available Tasks:

DS ID	Datastream name

<Selected <<All Selected> All> Copy

Select... Send New Associated Types... Last Changed... Exit

Rendering Engine Proof-of-Concept

Model Definition - NEW

Model Id: Template Type: Name:

Name: ☐ Use for all types

Description: Start Date/Time: End Date/Time:

Selected Tasks

Id	Req	Name
----	-----	------



ATP 3.5 v1.0 (System)

File View Help

Model TP | File Export | File Upload | Issue | Request | Specialty Updates | Asset Extractor | FX Tasks

Model Id: Template Type: Template Name:

Asset Name: ☐ Use for all types

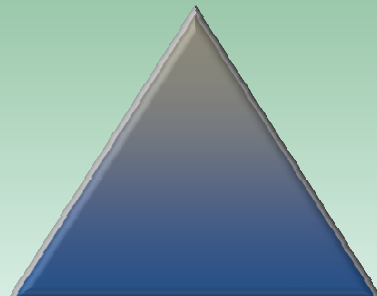
Description: Start Date/Time: End Date/Time:

Selected Tasks

DS_ID	REQ	DS_NME
TNT00003	Y	NORTHERN TRUST
TNT00008	N	VENDOR INHIBIT
TNT00014	Y	FUNDMASTER ONLY
80000123	N	FINS ROLE/ISSUE P
80000203	Y	ISSUE - UPDATE

Milestones

- 1) Create a preliminary catalog of control coordinates by form
- 2) Add field mapping information with domains for lists and combo boxes
- 3) Code read-only SQL data request methods in the model data layer using TDD
- 4) **Create a dynamic rendering engine to convert view data into functional screens**
- 5) Add event handling delegates for data manipulation, validation and navigation
- 6) Once data can be queried, work on submitting updates to that data
- 7) Code validation and update methods in the model data layer using TDD
- 8) Add exception display processing to the controller and rendering engine
- 9) Code add new and delete record methods in the model data layer using TDD
- 10) Write boundary test cases to ensure exception handling works
- 11) Convert SQL methods to optimized stored procedures and run the test suite
- 12) Work on additions to the catalog and refinements to the data and presentation components iteratively until the SmallTalk AIP application can be replaced.



Future Concerns

- **Multiple Document Interface host application**
- **Some form of editor for new template mappings**
- **Optimized search filtering with type-ahead and paging**
- **Support for multiple platforms including:**
 - ❖ **Command-line interface**
 - ❖ **Web applications**
 - ❖ **Other web service requests**

