Unisys IT EAI Modeling Strategy with the UML

eIntegration Team
Unisys IT
UIT EAI Modeling Strategy

- One modeling language - Unified Modeling Language
- One modeling tool - Rational Rose
- Three levels of abstraction
  - Enterprise – Road Atlas
  - Value Chain – Route Plan
  - Application Interface – Street Map
- OMG UML Profile and Interchange Models for EAI
  - Complementary to our approach
  - Package software support issue
  - Evolving towards
Enterprise Model  

Road Atlas  

Vision – a high level logical model that describes the relationships and dependencies among systems.

- Purpose of the Enterprise Model
  - Catalog applications - suppliers / consumers
  - Catalog “business objects” - information documents
  - Show dependencies among producer/consumer
  - Abstracts interface and message format

- Usage/benefits
  - High level analysis, “ice breaker”
  - Portfolio review
  - Change impact assessment
  - Business objects where used
  - Integration planning
  - Reuse opportunity identification
Enterprise Model

Issues

- Key challenges
  - This uncharted desert isle
  - Where’s the doc?
  - But then they sent me away to teach me how to be logical

- Lessons
  - Just do it
  - ABC, easy as 123
  - Existing enterprise documentation can be helpful

- Best practices
  - Keep it simple
  - Make them an offer they can’t refuse
  - Brought to you by your local EAI competency center
  - Use it or lose it
Value Chain Model

Route Plans

Vision - physical realization of logical Enterprise Model for a set of related entities

- Purpose of value chain models
  - Identify the physical collaboration among actual information producers and consumers
  - Visualize information paths
  - Expose transformation points
  - Define contextual usage of data

- Usage/benefits
  - Enables business process view
  - Exposes data integrity, information latency and data fidelity issues
  - Supports process reuse, optimization, re-engineering
Value Chain Model

Issues

- Key challenges
  - Knowing why you’re taking the trip
  - Mission creep
- Lessons learned
  - Little bit country, little bit rock and roll
  - The “vision thing”
- Best practices
  - Don’t “just do it”
  - Model early, model often
  - Eat your own dog food
Application Interface Models

Street maps

Vision – a model to design and document an applications interface implementation.

- Purpose of application interface models
  - Specify application interfaces
  - Specify business service request

- Usage/benefits
  - Communicates
    - Intuitive yet rigorous
    - Common language (the UML)
    - Standard (complete documentation)
    - Through design, construction, testing, support
  - Get reuse to it
    - Plugging the brain drain
    - Reduce risk single point of knowledge
    - Reduces relearning time (learning curve)
    - Eliminates code wandering
Application Interface Model

Issues

- Key challenges
  - Level of detail to capture
  - Capturing data mapping specifications
- Lessons learned
  - Reading models: easy
  - Writing models: not so easy
  - How do I support thee, let me count the ways
- Best practices
  - Everything old is new again
  - Get a little help from your friends
  - Do some basic training
Summary

- One
  - Modeling language - Universal Modeling Language
  - Modeling tool - Rational Rose

- Three levels of abstraction
  - Enterprise – Road Atlas
  - Value chain – Route Plan
  - Application Interface – Street Map

- Conclusions
  - It works
  - IT vision and leadership is key
  - Competency center a critical enabler
  - Find what works for you to ensure wide participation
  - Balance breadth versus depth

- See November eAIJournal.com for detailed article
Thank You.
EAI Project Staffing Strategy

- Some full-time permanent staff
  - Core EAI team (aka Competency Center)
  - Shared data models and reusable work products
  - Specialized EAI technology and tool skills
- Some committed temporary staff for project durations
  - Application team members
  - Application data models, interfaces and internal processing
  - Application-specific technology and tool skills
- Some part-time temporary staff for task durations
  - Application team members, architecture staff, contract staff
  - Consulting or task delivery requiring deep knowledge of infrequently used technologies
  - Specialty technology and tool skills

Return