Step AP233 Systems Engineering

Experience from a Pilot Implementation

Jacques Spee spee@nlr.nl
NLR Flight division, Amsterdam, The Netherlands

STEP for Aerospace workshop
NASA/JPL, 25-27 January 2000
Presentation Contents

- Application - Vehicle Management System
- STEP AP233 Use & Experience
- Systems Engineering Environment
- Outlook

Questions? Afterwards…
Application
Vehicle Management System

Motion
Airframe
Fuel
Aerodynamics
Engine
Cooling
Actuation
Electrics
Hydraulics

VehSys
FUM
Engine
ECS
APU
Hyd
Fuel
Intake
Bleed
Inlet
RamAir
Avionics
Cabin
Outlet
Water
Fuel
Circuit

ECS
APU
Hyd
Fuel
Circuit

HYD:Red

HYD:Green

© CONVEX Consortium 1999
Application

Process Overview

Modeling

Synthesis

Analysis

Design

Visualization
Application

Engineering Process Issues

- Systems Integration
  Complex Interaction, Functional Diversity, Large Scale

- Consistent Product Model
  Multiple Notations/Views, Unified Models, Data Sharing

- Network Computing
  Distributed Resources, Dissimilar Operating Systems

- Process Management
  Traceability, Process State, Multi-disciplinary Teams
AP233

Dynamic System Models

---

**Engineering Definition**

Mixed Numerical/Algebraic Models
Performance Models
Transfer Functions
Linear and Nonlinear Frequency Responses, etc.

**Parametric Models**

```plaintext
function DCsystem_odes()
begin
  loop
    void DCsystem_ode()
    {
      "Automatic C-code generation"
    }
    Numerical integration
  end loop
end
```

**Real-Time Simulation Code**

**Non-Real-Time Simulation Code**

**DC Electrical System**

**DC Electrical System Model**

**Bond Graph**

Differential and Algebraic Equations
Automatic Generation of Symbolic Algebra
Model Consistency Checks

**Object Model**

© CONVEX Consortium 1999
AP233

Control Systems Engineering
**Main usage of AP233 - draft 1**

- **Units of Functionality applied**
  - functions: definition, instances, io_port
  - flows, bindings
  - data types: definition, instances
  - visual presentation

- **Schema adaptations**
  - data types: matrix, transfer function, state space, ...
  - parameters, parameter ports
  - domain-dependent units & scaling
  - a-causal modelling $\Rightarrow$ generic port and link
  - physical modelling
**AP233**

**Required Extensions**

- **Component definition language**
  - equations (formulas), tables, ...

- **Visual presentation**
  - system schematics, including system dynamics info
  - graph plots: multi-dimensional
  - user interfaces

- **Configuration management, PDM-harmonised**

- **Executable context (scenarios), re-usable**

- **Multi-level model details**
See
Systems Engineering Environment

--- User Interface Layer
--- Tool Layer
--- Communication Layer
--- Server Layer

© CONVEX Consortium 1999
**Engineering Workflow - Definition**

**Link:**
- data flow between activities
- defined type of data from AP233

**Assigned Tool: Executable Task**
- automatically started upon task execution
- inputs/outputs predefined data types

**Hierarchy: Task Decomposition**
**SEE Benefits**

- On-line process state, always up-to-date
- Traceability, process history
- No low level tool / data handling
- Knowledge capture, best-practice templates
- AP233 neutral format for dynamic system models
- Shared product model, managed configuration
- Distributed computing facilities as one computer
Outlook

Collaborative Engineering

- **Visual-iSE (Visual integrated Systems Engineering)**
  - computer-based systems engineering infrastructure
  - AP233, shared repository, engineering workflow

- **Platform for:**
  - Concurrent Engineering in Aircraft Development
  - Multi-disciplinary Optimisation, Blended Wing Body

© CONVEX Consortium 1999
Outlook

**AP233 Development**

- Systems engineering integrating all STEP APs???
- Involve all stakeholders

**PARTICIPATE!!!**
- Generate User Requirements
- Apply & Test AP233
- Feedback Experience