Command Management Prototypes

Theresa Brandt

24 August 1995

Command Management Prototype

Command Level Constraints

Ground Script Changeover

Reuse Maximization

Command Constraints

Purpose

- **Identify Constraint Types**
- **Evaluate Performance Considerations**

Approach

- **Studied Constraint Checking on Existing Missions**
  - **WIND/POLAR CMS**
  - **PARR**
- **Requested Information on Constraints from LMC**
- **Discussed Constraint Rule Types with FOT**
- **Examined Need for Interactive Constraint Checking**

Command Constraint Results

Wind/Polar Disadvantages

- **C Code, Not C++**
- **Searches Entire Command List for Each Constraint**

PARR Disadvantages

- **Focuses on Activity Level Constraint Checking**

Customized Solution

- **Lessons Learned from Wind/Polar and PARR**
  - **Rule Types**
  - **Integrated FOT Suggestions**
  - **Database Defined Constraints**
  - **Associating Constraints with Command Facilitates Access**
  - **Command List Implemented as Doubly Linked List to Allow Forward and Backward Search for Constraint Satisfiers**

Command Constraint Results

- Command Model will Spawn New Process to Perform Constraint Checks
  - Allows for Simultaneous Checking of DAS Command List, RTS Definitions, Activity Definitions, and Procedure Definitions

- Interactive Command-level Constraint Checking Can be Eliminated
  - Command Sequences Within an Activity Definition are Constraint Free

    Reduces Probability of Command-Level Constraint Violations in Scheduling or Load Generation
  - Activity Constraints are Interactively Checked at Scheduling
  - Reduces Amount of Data to be Passed between PAS and CMS

CMS Ground Script Changeover

Purpose

- **Streamline Transition from One Ground Script to Next**
- **Eliminate Need for Configuration Management of Ground Scripts**

Approach

- **Discussed Operations Concept with FOT**
  - **Present Current and Proposed Concepts**
  - **Get Feedback**
- **Analyzed PDR Design of CMS and FUI**

CMS GS Changeover Results

CMS Will Maintain Continuous Ground Schedule

- **New Directives Merged into Ground Schedule When DAS Received**

User Will Request Portion of Ground Schedule Based on Time Span

- **Previous User Edits of Ground Script Will Be Retained When New Portion of Ground Schedule Is Loaded**
- **User Will Be Notified If Request Is Not Contiguous**

CMS Reuse Maximization

Purpose

- **Reuse Software Across FOS Subsystems**

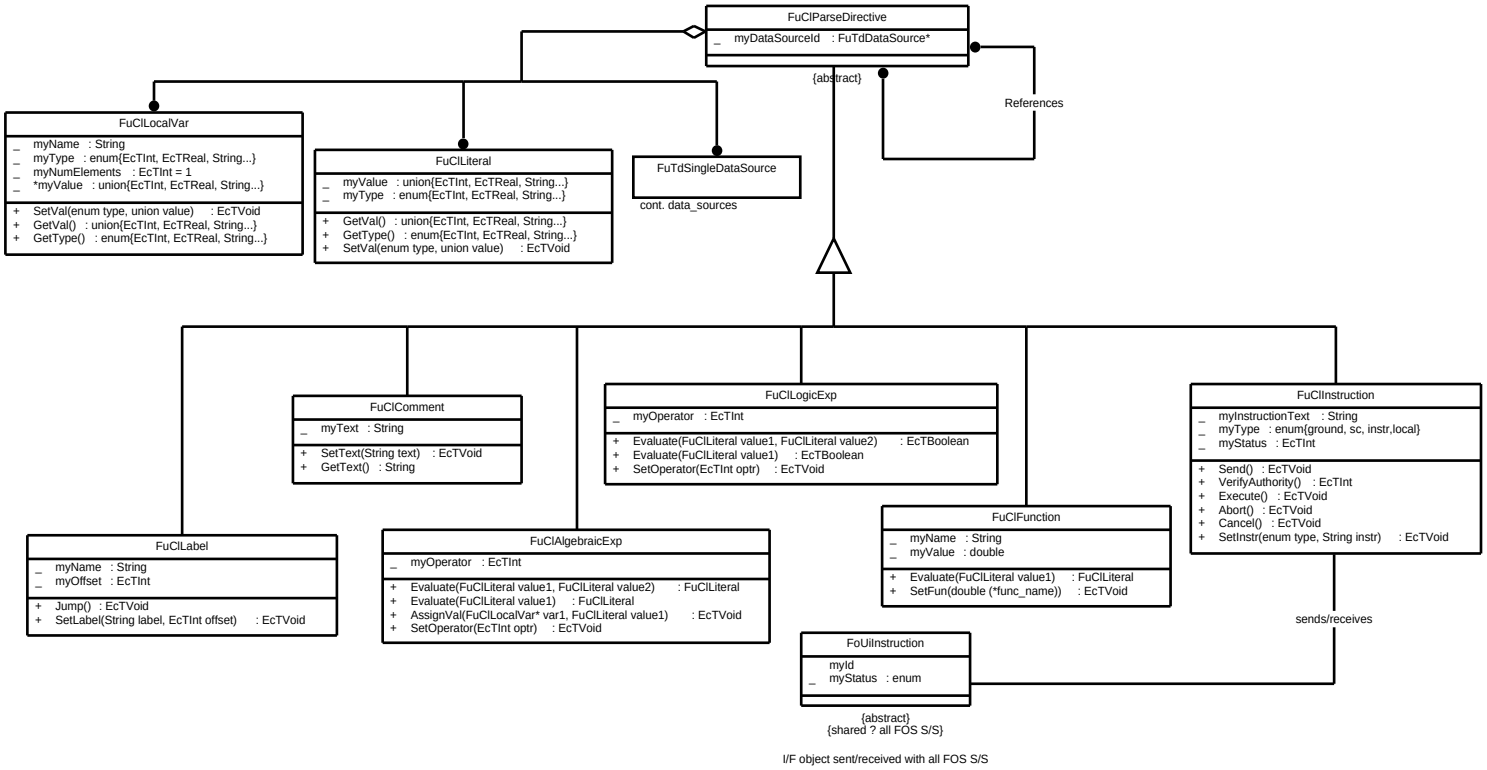
Approach

- **Analyzed Prototype Design of PAS Interface and CMS Processes**
- **Analyzed PDR Design of CMS and other FOS Subsystems**

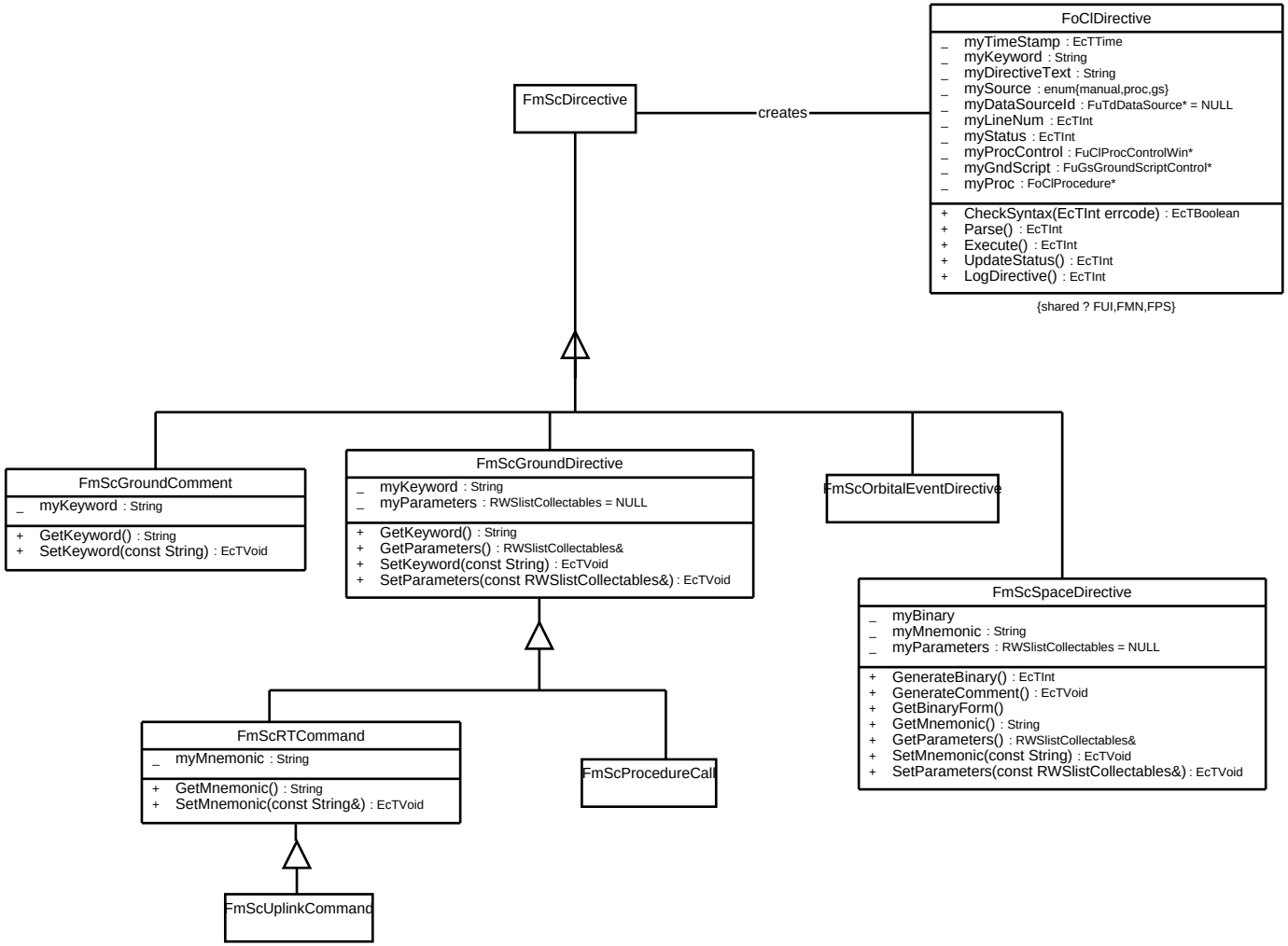
Results

- **Merged Command Directive Class with FUI and DMS**

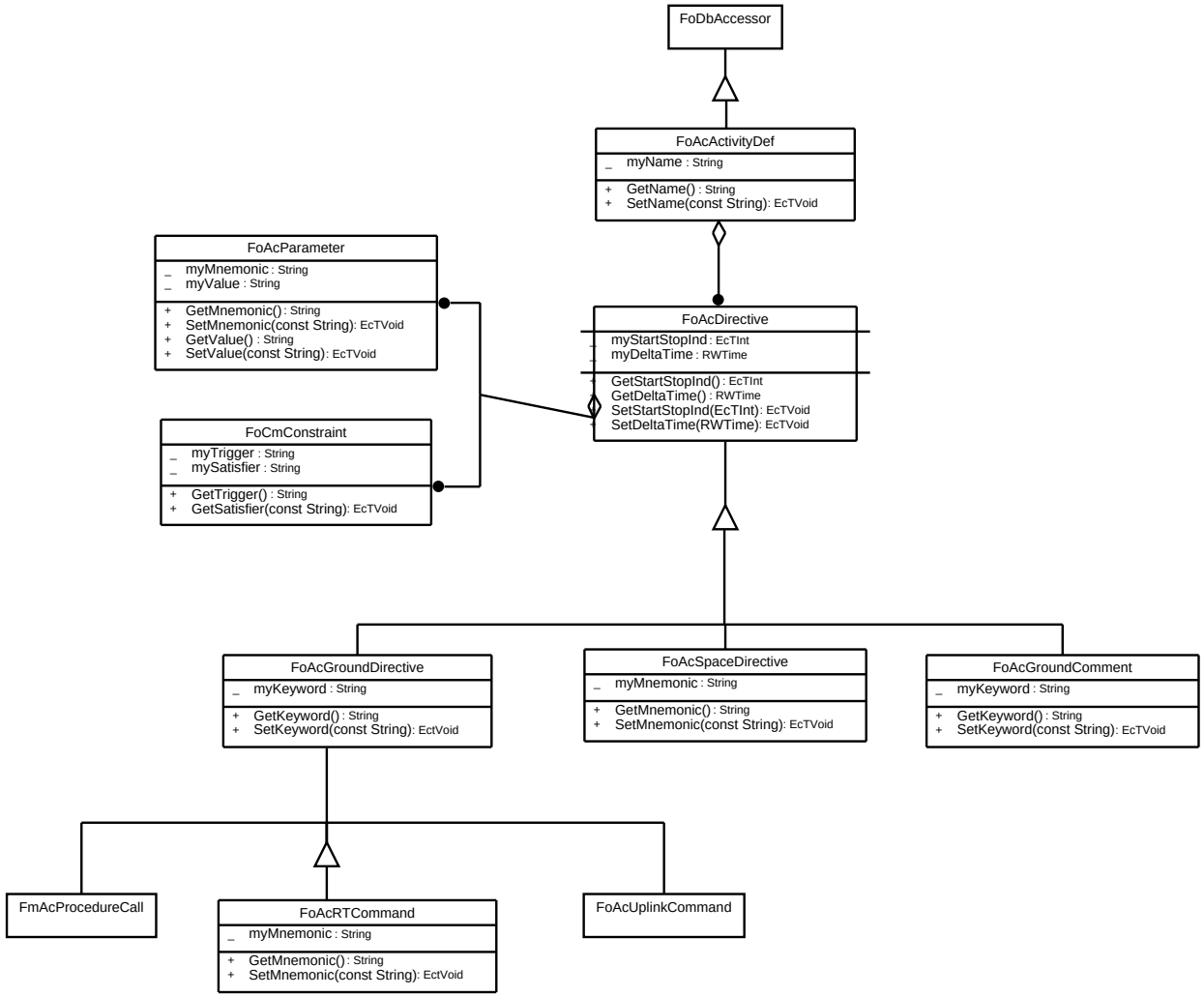
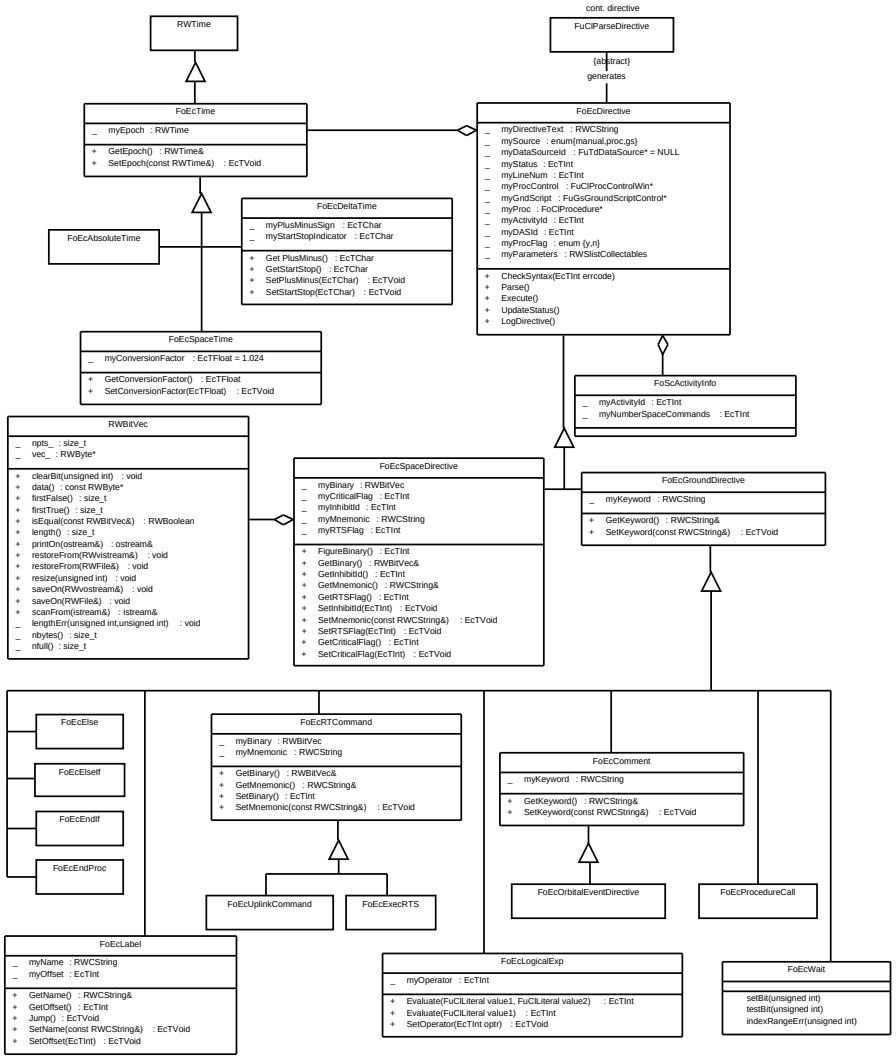
FUI Directive Object Model



CMS Schedule Directive



DMS Activity Definition Directive



FOS Merged Directive