



Collaborative Assistive Tool To Enable Novel Solutions: CATTENS

Stephen Waterbury, Code 585
February 17, 2016

Contents

- In a Nutshell
- Scope and Goals
- Stretch Goals
- CATTENS Architecture Diagram
- Use of Ontologies
- Demos

In a Nutshell

CATTENS will provide a Collaborative Design Environment utilizing the concept of model based systems engineering to enable the rapid development of a conceptual design from a clean sheet of paper to a proof of concept that can demonstrate first principle mission performance in the space of a week. In its final form, CATTENS is envisioned to enable design verification through Phase B and to incorporate performance verification tracking through launch and early operations.

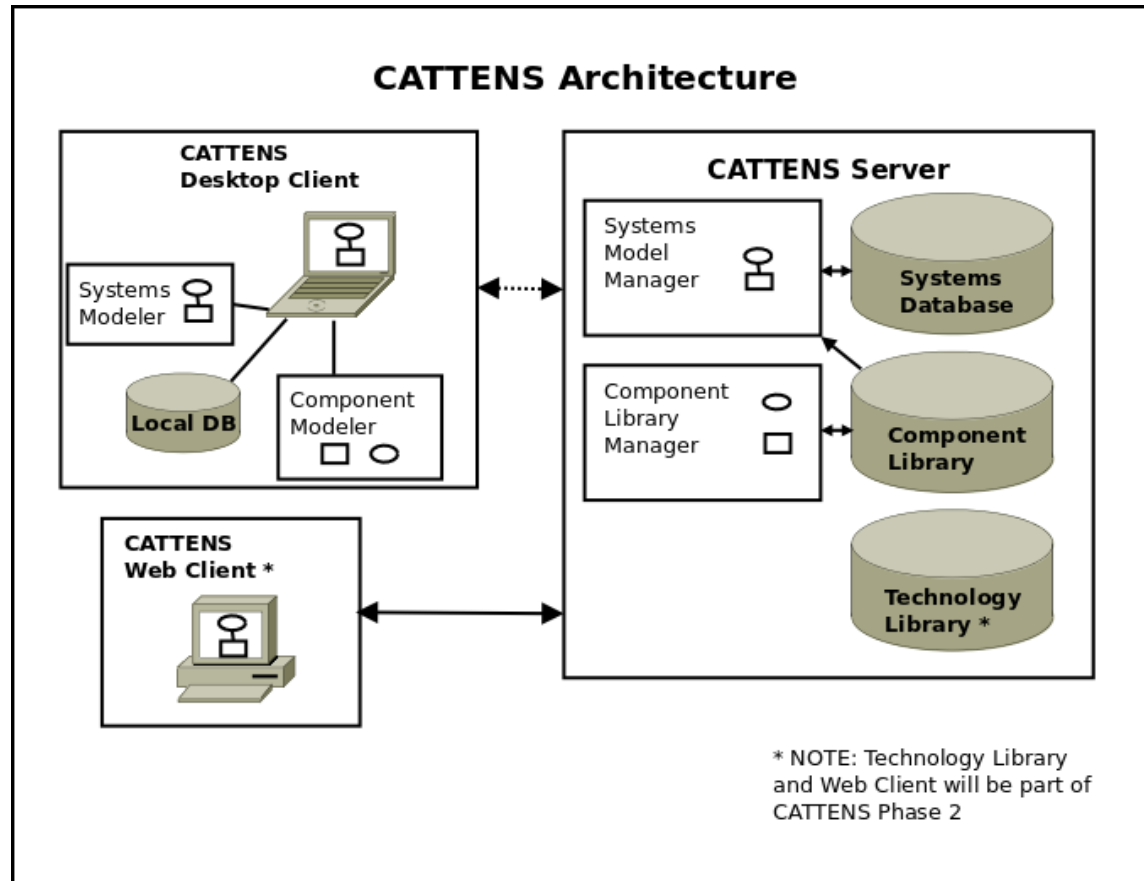
Scope and Goals

- Mission Design Lab [MDL] collaborative process support
- Mission Requirements capture, flowdown, and allocation
- Mission Con-Ops modeling
- Mission Elements and Components modeling
 - Mission Systems assembly structure
 - Master Equipment List (MEL)
- Parts Library (re-use of component models)
- Tracking of figures of merit
 - Performance budgets by subsystem and component
 - Resource budgets by subsystem and component
 - Customizable reporting / presentation

Stretch Goals

- Data exchange with other MBSE tools
- General Requirements Management Tool
 - As easy to use as Excel
 - But way better (an actual database)
 - And networked / collaborative
- Service Architecture for integrated MBSE / MBE
 - Engineering team status "dashboard" functions
 - Communication / notification via pub/sub channels
 - Design / Analysis collaboration services
- Model Repository Service (MBSE, CAD, FEA, ...)
- Engineering Parts / Models Library Service
- Engineering Knowledgebase / AI Service ["SkyNet"]
 - Searchable component histories / where used
 - System validation rules
 - Generic requirements database (e.g., GEVS)

CATTENS Architecture



Use of Ontologies

- An "Upper Ontology" (abstract domain model) is used
 - Based on core concepts of STEP (ISO 10303)
 - A unified conceptual model of engineering objects
- Provides the basis for Discipline / Application ontologies
 - MDL domain (Mission, Spacecraft, ...)
 - Mechanical Subsystems (Fairings, etc.)
 - Electro-Mechanical Subsystems (Attitude Control, etc.)
 - Electronic subsystems (Communications, etc.)
- DB tables and app classes are generated from ontologies
- Use of ontologies enables linkage to:
 - Inferencing and Rule Engines
 - Supervised structured machine learning
 - Intelligent optimizations

Demos

- Generation of DB and App Classes from Ontologies
- Client Modes: Data / DB / Systems
- Object Viewer / Editor
 - Generated at runtime from the ontology-defined class schemas
- Basic Requirements Management