KnowledgeBytes

for the best ideas in information technology

www.KnowledgeBytes.net

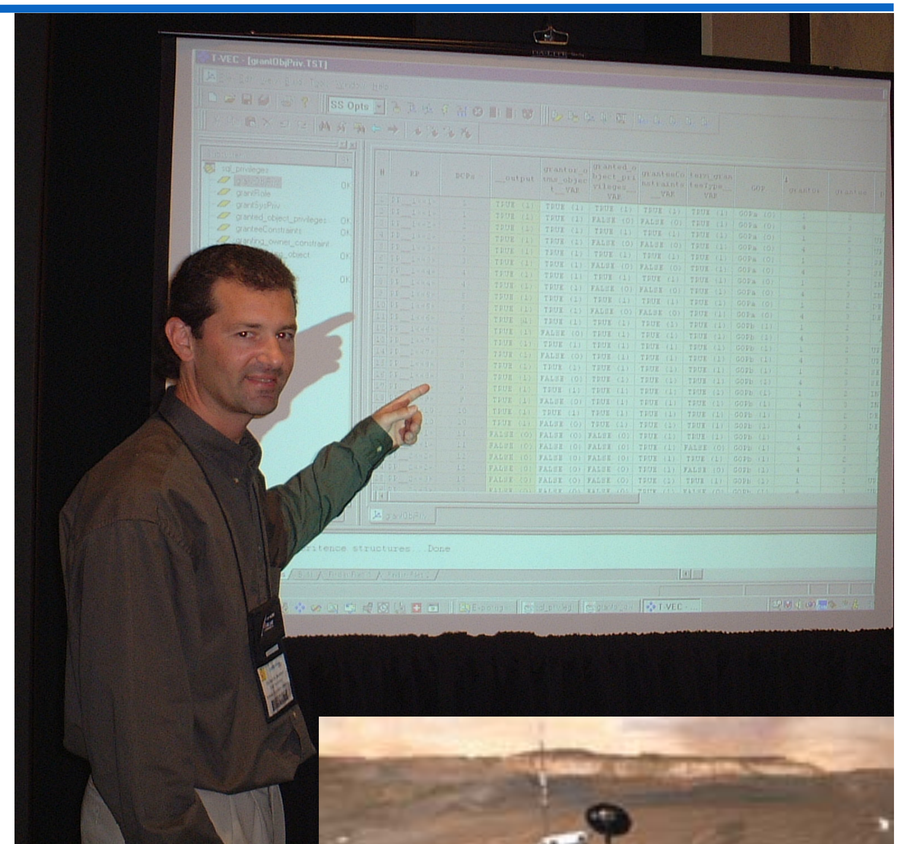
Collaborative Requirements Engineering for Smart Manufacturing System Verification and Validation

Prepared for NASA IV&V Workshop – August 19, 2013

Mark R. Blackburn, Ph.D.
Peter Denno (NIST)

About the Presenter: Mark R. Blackburn, Ph.D.

- 11 years building flight-critical avionics software and applying model-based software tools to automate test generation
- 19+ years experience in building modeling and analysis tools, commercial consulting, & entrepreneurial endeavors
- Researching Domain-Specific Modeling and model-based V&V approaches for autonomous adapting systems (e.g., robots)
- Notable Accomplishment: Discovered Mars Polar Lander Error in Fewer Than 24 Hours



Outline

- Part 1: Overview
  - Context and problem
  - NIST Collaborative Requirement Engineering (CRE) objective
  - Project objective
- Part 2: Approach
  - Leveraging Domain Specific Modeling (DSM)
  - Rationale
- Part 3: Design and usage details
  - Analysis and test generation
  - Operational scenarios
  - Demonstration
- Summary
  - Broader impacts and relevance to NASA

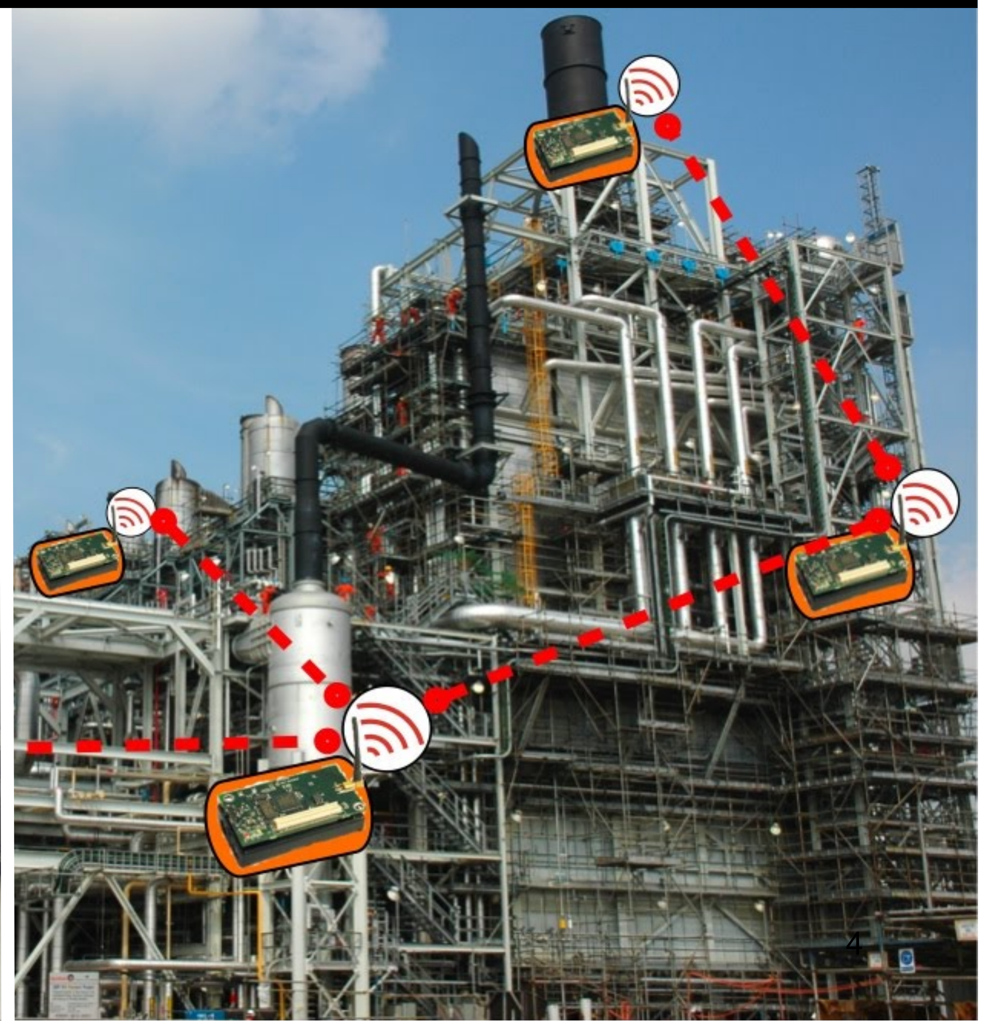
Part 1: Overview

Context and Problem

NIST Objectives

Project Objectives

Smart Manufacturing will enable development and V&V of Cyber-Physical Systems with increasing computational capabilities that will manage critical infrastructure



Context (1/2) – Example CHL System Breakdown Specification


NIST CRE PROJECT

CHL System Breakdown

System Performance and Components Specification

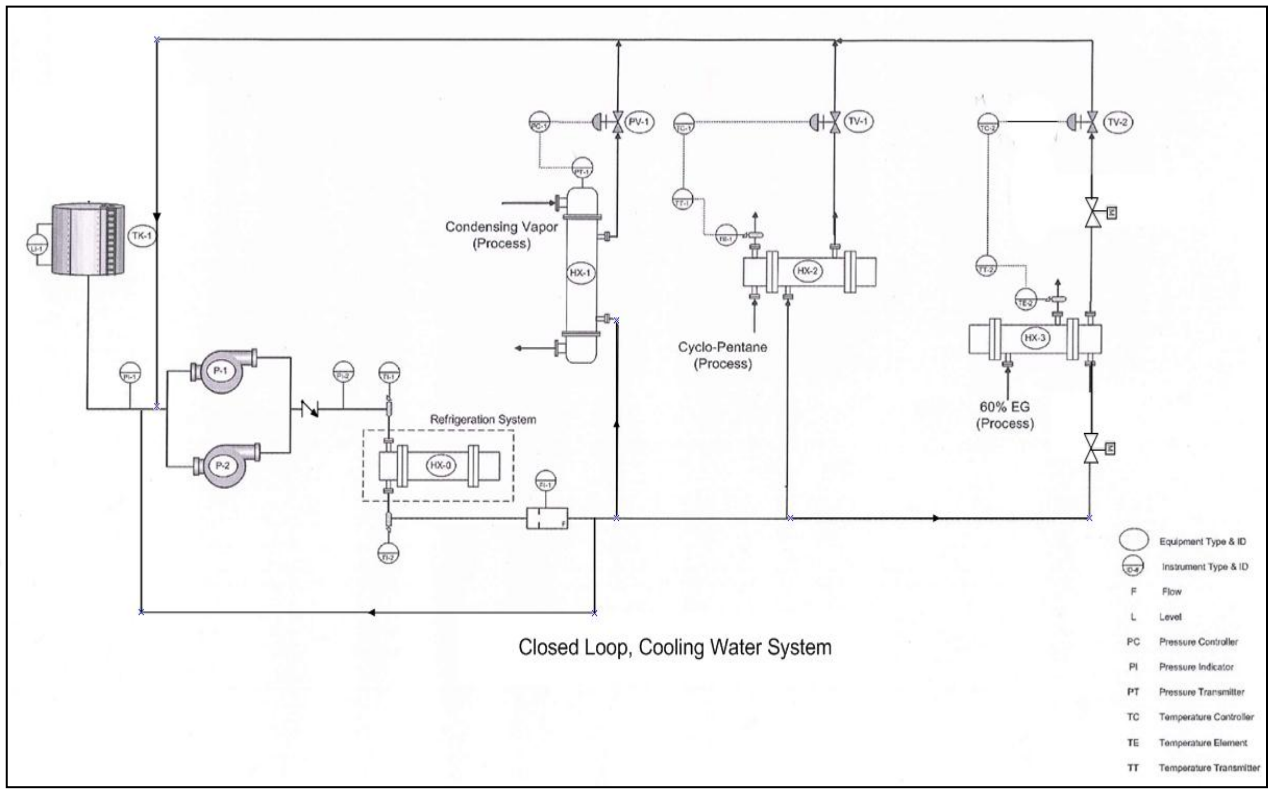
NIST CRE

10/4/2012

A description of the CHL system composition and high/low level requirements in terms of performance and interfaces. This document should aid in further defining the CHL system for adapting to other interfacing systems.

Includes

Closed loop, Heat transfer, Liquid circulating (CHL)



May also include:

- System Requirements Tree
- System Requirements
- Component Requirements
- Interface Requirements

Context (2/2) - Current Process Relies Heavily on Paper-based Specifications

- Product data sheets provide information that specifies functional and physical characteristics of a component of a system, plant or facility

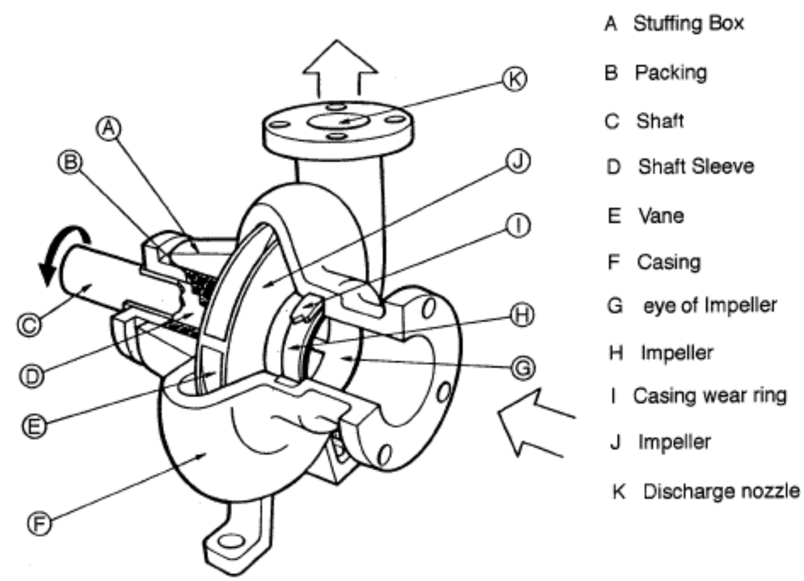
Product Data Sheet

CENTRIFUGAL PUMP API-610

DATA SHEET
MKS UNITS

JOB NO. ______ ITEM NO. ______
REQ / SPEC No. ______ / ______
PURCH ORDER No. ______ DATE 29-Oct-09
INQUIRY No. ______ BY DAA

1 APPLICABLE TO: ○ PROPOSAL ○ PURCHASE ☐ AS BUILT
2 FOR: ______ UNIT: ______
3 SITE: ______ SERVICE: ______
4 NO. REQ ______ PUMP SIZE ______ TYPE: ______ NO. STAGES ______
5 MANUFACTURER ______ MODEL: ______ SERIAL No. ----------
6 NOTES: INFORMATION BELOW TO BE COMPLETED ○ BY PURCHASER ☐ BY MANUFACTURER ○ BY MANUFACTURER OR PURCHASER
7 ○ **GENERAL**
8 PUMPS TO OPERATE IN (PARALLEL) No. MOTOR DRIVEN ______ No. TURBINE DRIVEN ______
9 (SERIES) WITH ______ PUMP ITEM No. ______ PUMP ITEM No. ______
10 GEAR ITEM No. ______ MOTOR ITEM No. ______ TURBINE ITEM No. ______
11 GEAR PROVIDED BY ______ MOTOR PROVIDED BY ______ TURBINE PROVIDED BY ______
12 GEAR MOUNTED BY ______ MOTOR MOUNTED BY ______ TURBINE MOUNTED BY ______
13 GEAR DATA SHT. No. ______ MOTOR DATA SHT. No. ______ TURBINE DATA SHT. No. ______
14 **OPERATING CONDITIONS** | **SITE AND UTILITY DATA (COT´D)**
15 ○ CAPACITY NORMAL ______ (m^3/h) RATED ______ (m^3/h) | WATER SOURCE ______
16 OTHER ______ | CHLORIDE CONCENTRATION (PPMW) ______
17 ○ SUCTION PRESSURE MAX/RATED ______ / ______ (Kg/cm^2g) | INSTRUMENT AIR: MAX / MIN PRESS ______ / ______ (Kg/cm^2)
18 ○ DISCHARGE PRESSURE ______ (Kg/cm^2g) | **LIQUID**
19 ○ DIFERENTIAL PRESSURE ______ (Kg/cm^2g) | ○ TYPE OR NAME OF LIQUID ______
20 ○ DIFF. HEAD ______ (m) NPSHA ______ (m) | ○ PUMPING TEMPERATURE:
21 ○ PROCESS VARIATIONS ______ | NORMAL ______ (°C) MAX ______ (°C) MIN ______ (°C)
22 ○ STARTING CONDITIONS ______ | ○ VAPOR PRESSURE ______ (Kg/cm^2) ______ (°C)
23 SERVICE: ○ CONT. ○ INTERMITTENT (STARTS/DAY) ______ | ○ RELATIVE DENSITY (SPECIFIC GRAVITY):
24 ○ PARALLEL OPERATION REQ'D | NORMAL ______ MAX ______ MIN ______



Problem

- As components (e.g., pumps, values, heat exchanges, piping, etc.) in facilities wear out, new components are substituted
  - Common for original requirements or design to not exist
  - May not to know how current facility implementation deviates from original design or requirements
- Concern that newly substituted component can create potential operational or safety issues of overall facility (e.g., produce too much heat, incorrect input/output pressure)

NIST CRE Objective

- Develop, extend and apply **formal methods** of requirements representation and demonstrate framework, protocols, data models, and tools for **collaborative requirements engineering** to support process facility design (aka “smart” manufacturing)
- A key focus of the research area is to **apply formal methods to requirements engineering**
  - A formal method is a method on which the rules of inference can be shown mathematically to be valid
- Benefits
  - Improved system validation
  - Ability to better trace rationale
  - Improved systems engineering

Project Objective

- Develop formal method representation of systems and requirements that facilitates management of requirements and generation of tests from requirements specifications
  - Support automating generation of system tests
  - Describe logical structure of envisaged system (its components and their interconnection)
  - Describe how interaction of components achieves system goals
- Apply formal method of representation to a closed-loop liquid circulating heat transfer (CHL) system
- Use resulting model and tooling to:
  - Analyze requirement/design specifications/models
  - Generate tests
  - Demonstrate traceability of tests to corresponding requirements

Part 2: Approach

Rationale

Design Theme:
Leveraging Emerging and Enabling Technology

Domain Specific Modeling Languages and Tools

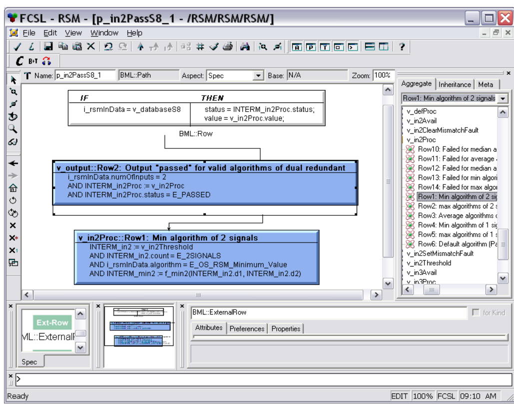
Approach: Key Themes

- Leverage Domain-Specific Modeling (DSM) Language
  - Provides relevant and intuitive graphical abstraction for the specific domain or subdomain
  - Allows for rich semantics required for formal analysis and test generation necessary for V&V effectiveness and efficiency
  - Addresses deficiencies in general purpose modeling approaches
- DSM tooling allows multiple views to be integrated
  - Model transformation built into the tools
  - Model languages are evolvable
- Integrates with formal analysis and test generation tools
  - Test generation provides effective test method
  - Formal methods hidden behind the scenes

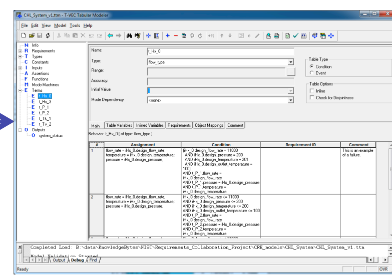
Based on Demonstrated DSM Pattern from DARPA Sponsored Research

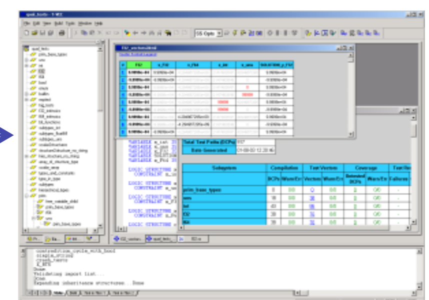
- DSM are easily evolvable
- Integrated analysis and test generation help in validating evolution of DSM modeling environment

Flight Control Domain-Specific Language



T-VEC Vector Generation System (VGS)



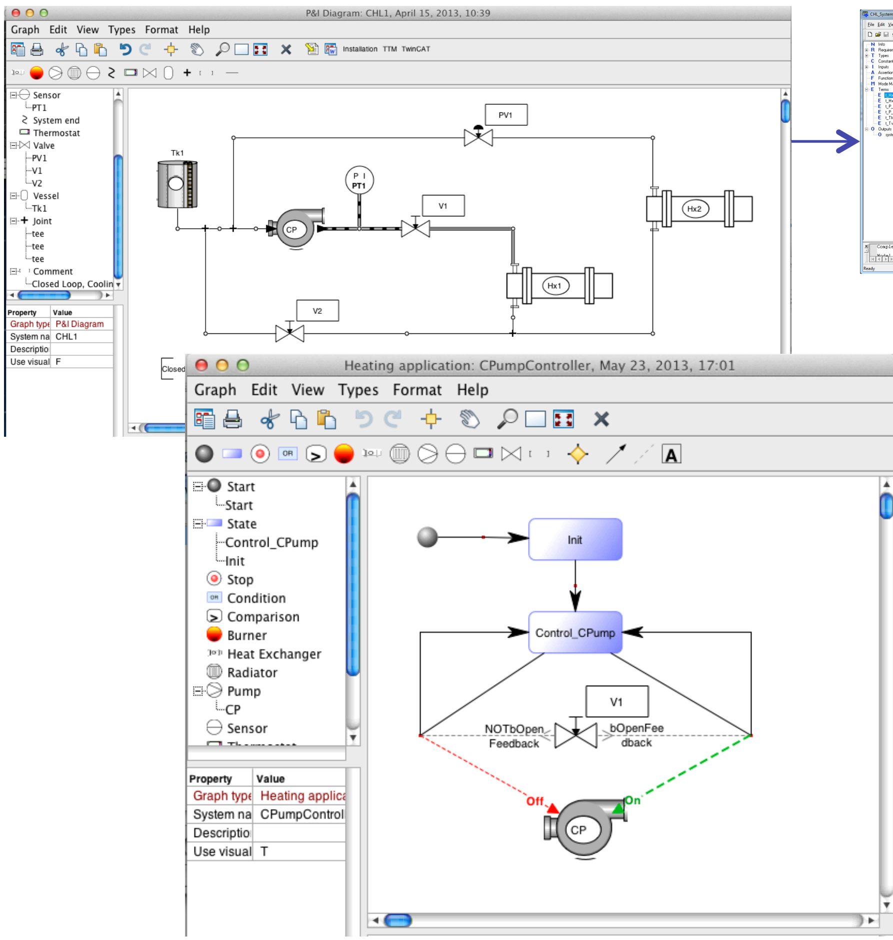


Producible Adaptive Model-based Software (PAMS) technology to the development of safety critical flight control software. PAMS has been developed under the Defense Advanced Research Projects Agency (DARPA) Disruptive Manufacturing Technologies program. Contract # N00178-07-C-2011.

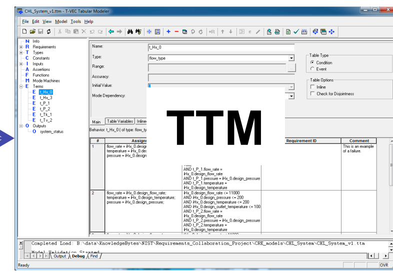
MODEL-BASED ADAPTATION OF FLIGHT-CRITICAL SYSTEMS, *Sumit Ray, BAE Systems, Johnson City, New York, Gabor Karsai, Vanderbilt University, Nashville, Tennessee, Kevin M. McNeill, BAE Systems, Arlington, Virginia, Digital Avionics Systems Conference, 2009*

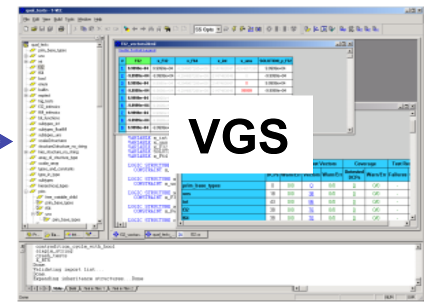
Prototype Design integrates DSM with Formal Methods Tools

Domain Specific Modeling (DSM) for Process Facility Design



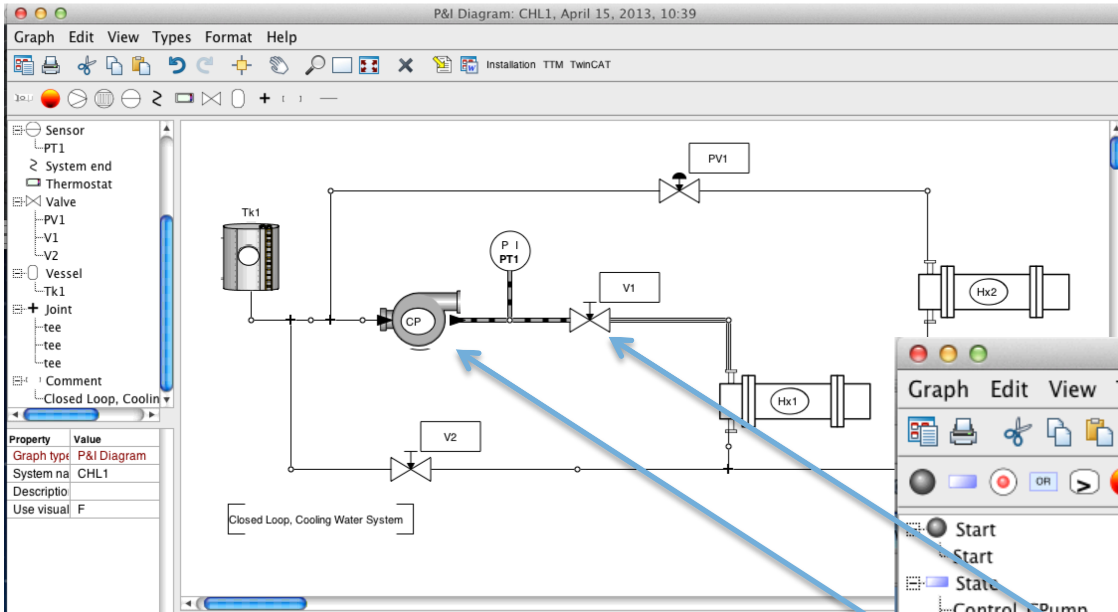
T-VEC Tabular Modeler (TTM) and T-VEC Vector Generation System (VGS)





- Model transformation
- Formal methods analysis
  - Theorem proving
  - Property checking
- Test vector generation
- Test driver generation
- Requirement-to-test traceability

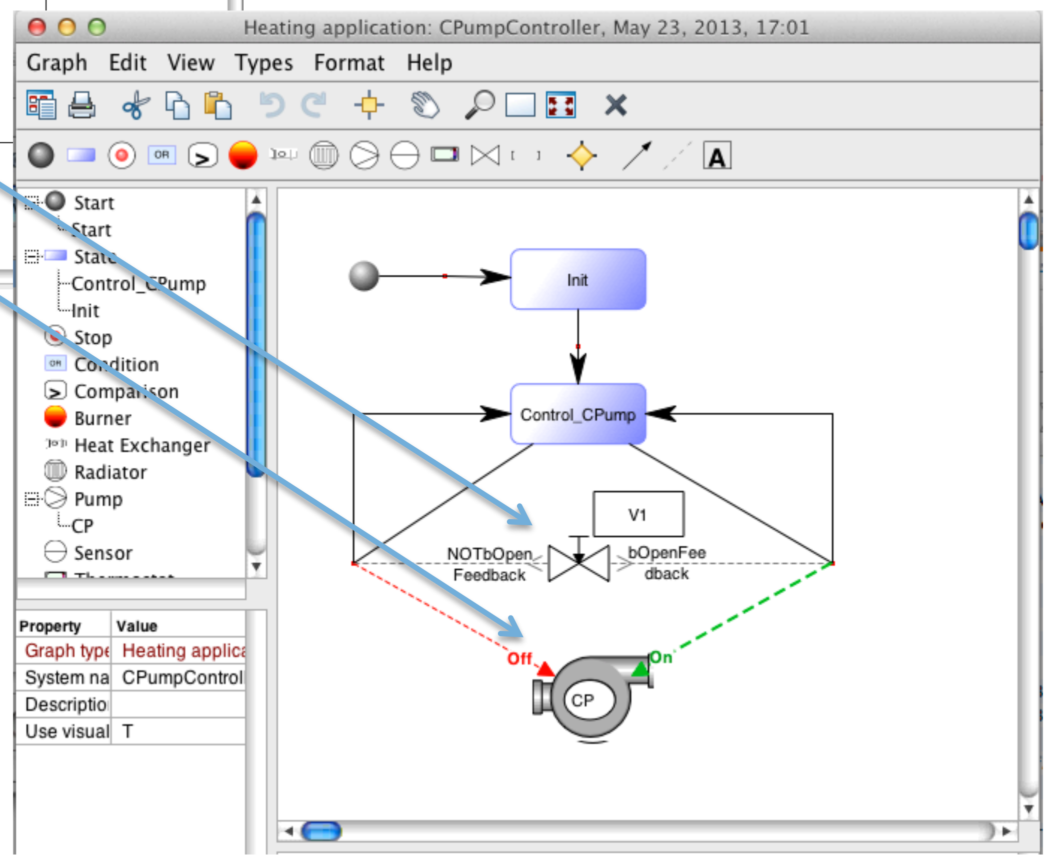
Two Different View: Structural with Flow Properties and Behavioral



Behavioral constraint:

if valve is closed then
  pump should be closed
else if value is open then
  pump can be open

Same objects: different views used to formalize different aspects of the system



Part 3: Design and Usage Details

Analysis and test generation

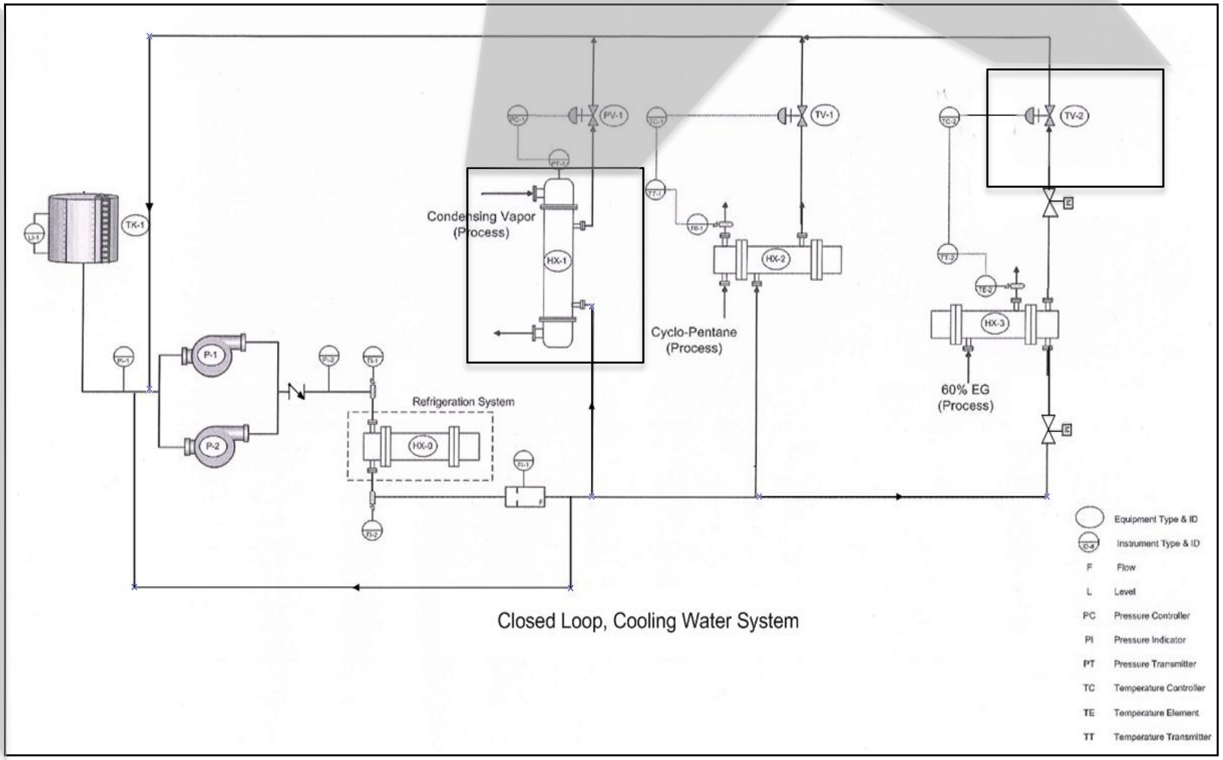
Operational scenarios

Demonstration

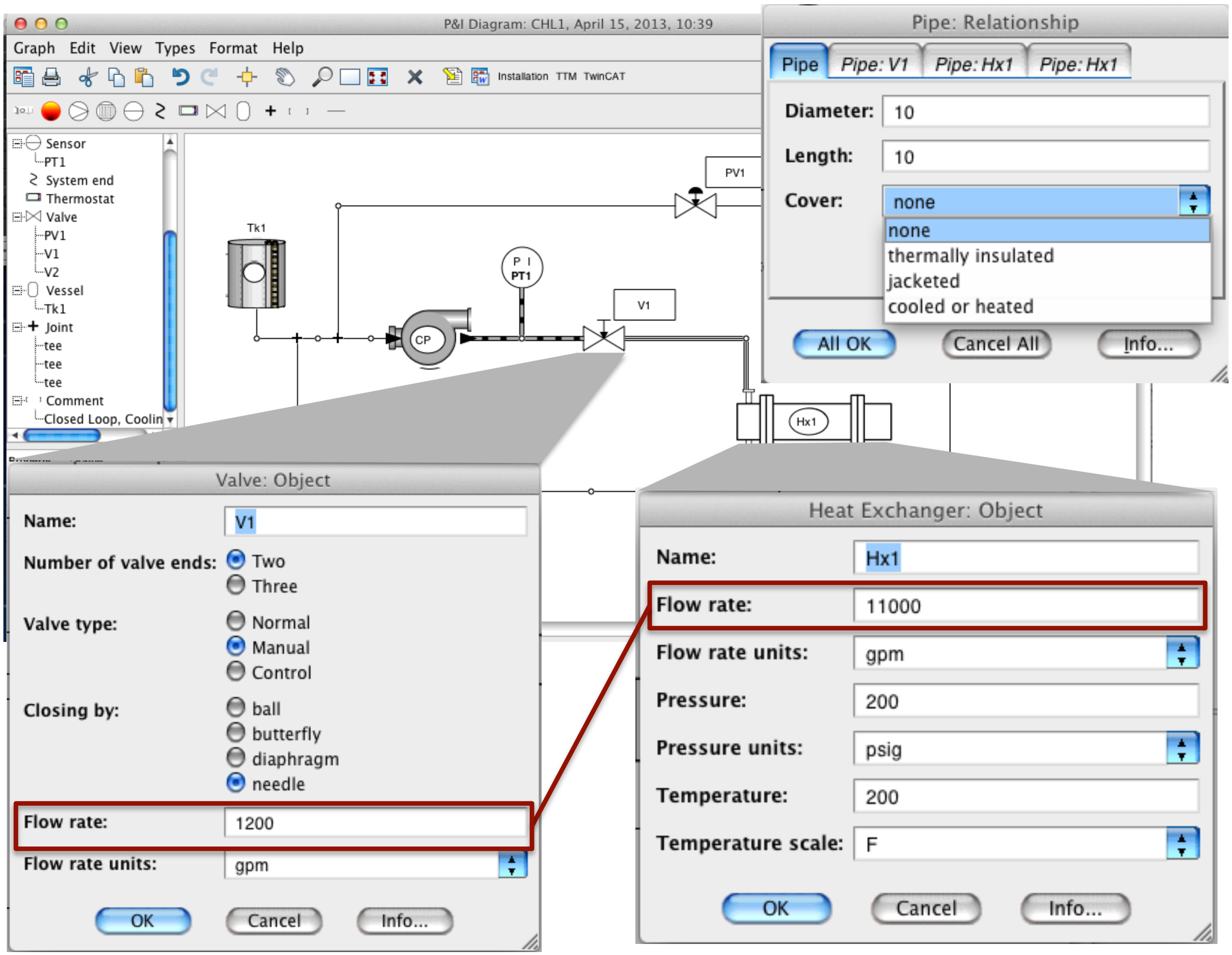
Formalize Component Behaviors and Properties and Composed System

Formalize relationships of composed elements

Formalize characteristics elements

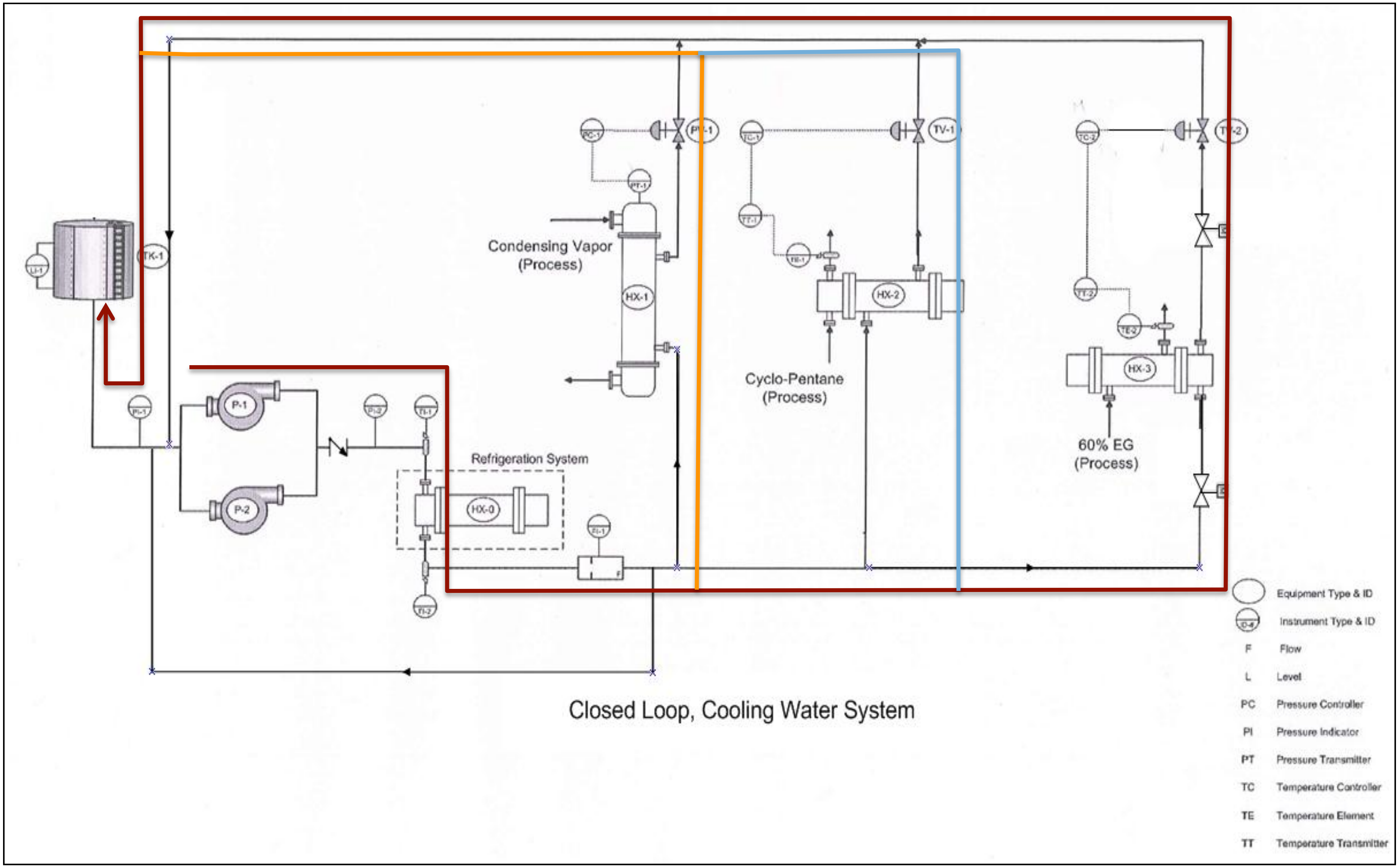


DSM Captures Component Properties (example includes seeded defect)

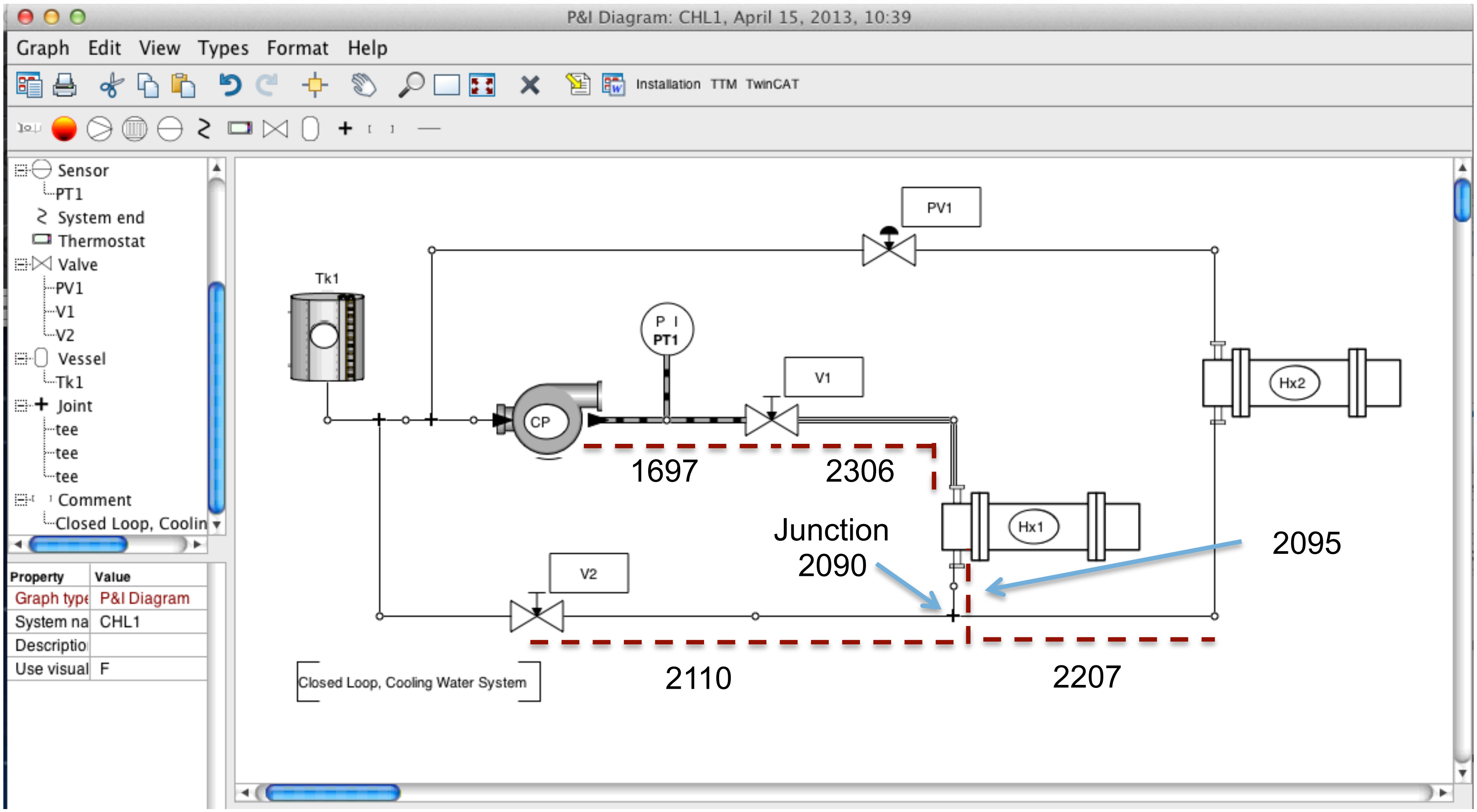


Potential Flow rate Issue

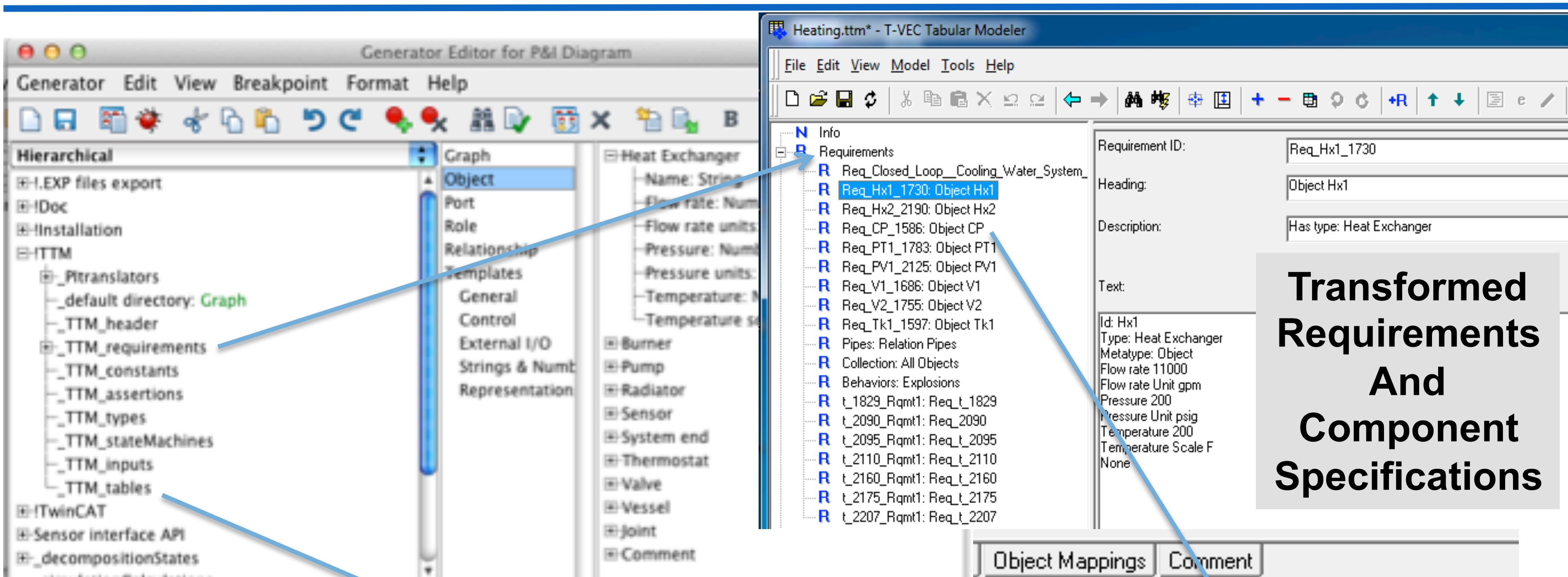
Model and Tests must cover all Flow Paths (partial view shown)



Model Transformation for Prototype Based on Pipe and Junction Relations



DSM Tool Includes Programmable Generator for Transforming Models



Transformed Requirements And Component Specifications

Detailed Behavior Needed for Analysis and Test Generation (hidden)

Behavior: t_1697 (of type: Pump_type)

#	Assignment	Condition	Requirement ID

1	flow_rate = CP.flow_rate;
	pressure = CP.pressure;
	temperature = CP.temperature;
		CP.flow_rate = 1200
		AND CP.pressure = 200
		AND CP.temperature = 200
		AND t_V1.flow_rate = CP.flow_rate
		AND t_V1.pressure = CP.pressure
		AND t_V1.temperature = CP.temperature
			Req_CP_1586: Object CP

2	flow_rate = CP.flow_rate;
	pressure = CP.pressure;
	temperature = CP.temperature;
		CP.flow_rate <= 1200
		AND CP.pressure <= 200
		AND CP.temperature <= 200
		AND t_V1.flow_rate = CP.flow_rate
		AND t_V1.pressure = CP.pressure
		AND t_V1.temperature = CP.temperature
			Req_CP_1586: Object CP

Model Analysis and Test Generation Project Status

Heating Project Status

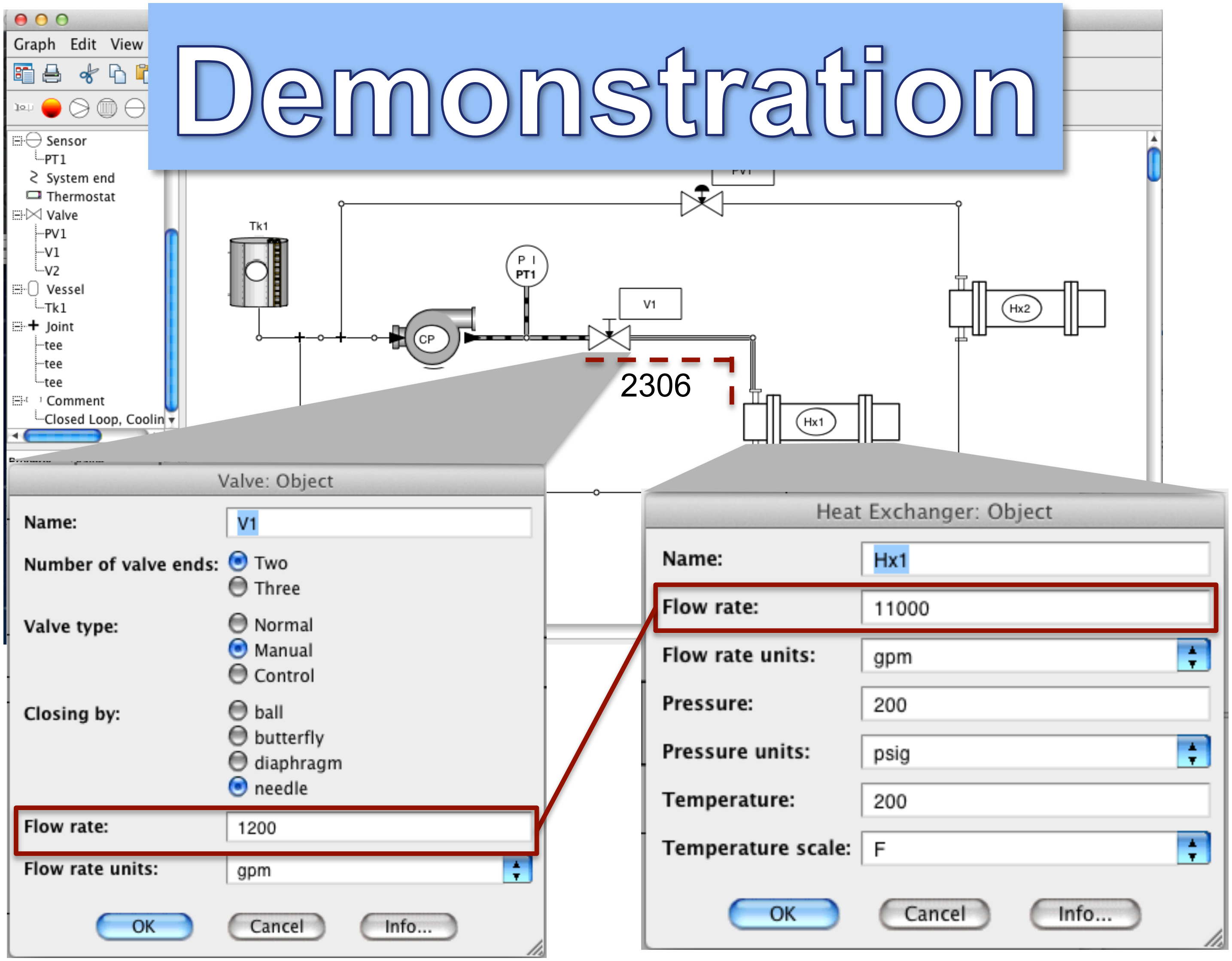
Project Filename: B:\projects\tvec_projects\metaEdit\Heating_system\Heating.prj

Total Test Case Comparisons	0
Total Test Comparison Failures	0
Total Test Vectors	31
Total Test Paths (DCPs)	21
Date Generated	07-14-13 16:37:21

Subsystem	Compilation		Test Vectors		Coverage		Test Results	
	DCPs	Warn/Err	Vectors	Warn/Err	Untested DCPs	Warn/Err	Failures	Comparisons
t_1697	2	0/0	3	0/0	0	0/0	-	-
t_1829	1	0/0	2	0/0	0	0/0	-	-
t_2090	3	0/0	2	0/4	2 of 3	0/12	-	-
t_2095	1	0/0	1	2/0	0	0/0	-	-
t_2110	1	0/0	2	0/0	0	0/0	-	-
t_2160	1	0/0	1	0/0	0	0/0	-	-
t_2175	1	0/0	2	0/0	0	0/0	-	-
t_2207	1	0/0	1	0/0	0	0/0	-	-
t_2222	2	0/0	3	0/0	0	0/0	-	-
t_2306	2	0/0	2	0/2	1 of 2	0/6	-	-
t_CP	1	0/0	2	0/0	0	0/0	-	-
t_Hx1	1	0/0	2	0/0	0	0/0	-	-
t_Hx2	1	0/0	2	0/0	0	0/0	-	-
t_PV1	1	0/0	2	0/0	0	0/0	-	-
t_V1	1	0/0	2	0/0	0	0/0	-	-

Reflects Seeded Defect (see next slide for Demo details)

Demonstration: Identify Seeded Defect between Valve and Heat Exchanger



Potential Flow rate Issue

Detailed Error Report Shows Flow rate Inconsistency

t_2306 Coverage Analysis FAILED

Vector File: B:\projects\tvec_projects\metaEdit\Heating_system\test_vectors\t_2306.TST

Time Run	07-14-13 16:36:20
Analyzer Version	4.0.0
Total Number of DCPs	2
DCPs Not Covered	1
Predicates Requiring Coverage	11
Predicates Not Covered	5 cv_t_2306_RP__1 t_2306_FR__1 t_2306 t_2306_1__LS t_2306_RP__1
Total Coverage Errors	6
Total Coverage Warnings	0
Test Generation Failures	2

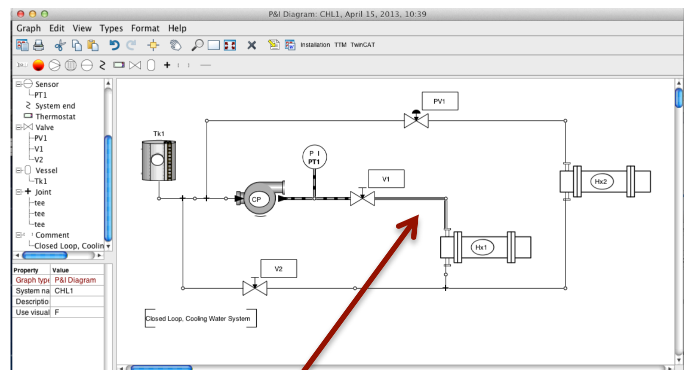
We are investigating alternatives to bring error information back to DSM Tool and Model Views

Uncovered DCP Paths

DCP Number	DCP Path	Failure Detection
1	t_2306, t_2306_FR__1, cv_t_2306_RP__1, t_2306_RP__1, t_2306_RP__0, t_2306_1__LS (goto model)	Vector Generator

Convergence Error Diagnostics

Error Number	DCP Number	Time Period	Constr Obj Boundary	DCP MultiSpace Variation	Failed Pred
1	1	T >= 0	LOW_BOUND	0	Lower level subsystem, click on error # for diagnostics.
2	1	T >= 0	HIGH_BOUND	0	Lower level subsystem, click on error # for diagnostics.



Hyperlink to detailed error

Failed Pre-Condition Relation	cv_equal_to	
Exact SS File Location	t_2306.SS	Line # 30
Input Domain At Error	Occurance at UA = 662 t_V1__VAR.flow_rate:Valve_type_flow_rateDataType [0 .. 1200] Hx1.flow_rate:Heat_Exchanger_type_flow_rateDataType [11000 .. 11000]	

Model Analysis Links Defect (Unsatisfiable Constraint) to Model

t_2306 Coverage Analysis FAILED

Vector File: B:\projects\tvec_projects\metaEdit\Heating_sy

Time Run	07-14-13 16:36:20
Analyzer Version	4.0.0
Total Number of DCPs	2
DCPs Not Covered	1
Predicates Requiring Coverage	11
Predicates Not Covered	5 — cv_t_2306_RP__1 t_2306_FR__1 t_2306 t_2306_1__LS t_2306_RP__1
Total Coverage Errors	6
Total Coverage Warnings	0
Test Generation Failures	2

Behavior: t_2306 (of type: Pump_type)

#	Assignment	Condition

1	flow_rate = Hx1.flow_rate;
	pressure = Hx1.pressure;
	temperature = Hx1.temperature;
		Hx1.flow_rate = 11000
		AND Hx1.pressure = 200
		AND Hx1.temperature = 200
		AND t_V1.flow_rate = Hx1.flow_rate
		AND t_V1.pressure = Hx1.pressure
		AND t_V1.temperature = Hx1.temperature

2	flow_rate = Hx1.flow_rate;
	pressure = Hx1.pressure;
	temperature = Hx1.temperature;
		Hx1.flow_rate <= 11000
		AND Hx1.pressure <= 200
		AND Hx1.temperature <= 200
		AND t_V1.flow_rate = Hx1.flow_rate
		AND t_V1.pressure = Hx1.pressure
		AND t_V1.temperature = Hx1.temperature

Hyperlink to model defect

Uncovered DCP Paths

DCP Number	DCP Path	Failure Detection
1	t_2306, t_2306_FR__1, cv_t_2306_RP__1, t_2306_RP__1, t_2306_RP__0, t_2306_1__LS (goto model)	Vector Generator

Convergence Error Diagnostics

Error Number	DCP Number	Time Period	Constr Obj Boundary	DCP MultiSpace Variation	Failed Pred
1	1	T >= 0	LOW_BOUND	0	Lower level subsystem, click on error # for diagnostics.
2	1	T >= 0	HIGH_BOUND	0	Lower level subsystem, click on error # for diagnostics.

Satisfiable Model Constraints Have Generated Test with Requirement Traceability

Test Vectors (Compressed) For Subsystem : t_1697

Vector File: B:\projects\tvec_projects\metaEdit\Heating_system\test_vectors\t_1697.TST

Open Model Report

Legend For Vector Table

Jump to Page: Vectors

Test # 3	ReqID(s) : Req_CP_1586: Object CP Utility ReqID(s) : Req_V1_1686: Object V1		
Name	**Value**	**Type**	**Relevant Domain**
t_1697__OUT.flow_rate	**0**	INTEGER	0..1200
t_1697__OUT.pressure	**0**	INTEGER	0..200
t_1697__OUT.temperature	**-100**	INTEGER	-100..200
CP.flow_rate	0	INTEGER	0..1200
CP.pressure	0	INTEGER	0..200
CP.temperature	-100	INTEGER	-100..200
V1.flow_rate	0	INTEGER	0..1200
V1.pressure	0		
V1.temperature	0		
V1.valve_type	"[\.]+		
V1.close_by	"[\.]+		
V1.number_valve_ends	"[\.]+		
a_V1__VAR	(TRUE =		
t_V1__VAR.flow_rate	0		
t_V1__VAR.pressure	0	INTEGER	0..200
t_V1__VAR.temperature	-100	INTEGER	-100..200
t_V1__VAR.valve_type	""	STRING	"".."z"
t_V1__VAR.close_by	""	STRING	"".."z"
t_V1__VAR.number_valve_ends	""	STRING	"".."z"

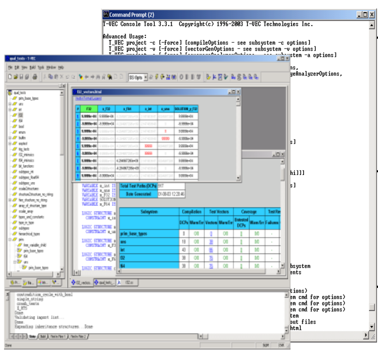
Link Requirement ID in the Model, and Requirement ID traced to generated test

Behavior: t_1697 (of type: Pump_type)

#	Assignment	Condition	Requirement ID

1	flow_rate = CP.flow_rate;
	pressure = CP.pressure;
	temperature = CP.temperature;
		CP.flow_rate = 1200
		AND CP.pressure = 200
		AND CP.temperature = 200
		AND t_V1.flow_rate = CP.flow_rate
		AND t_V1.pressure = CP.pressure
		AND t_V1.temperature = CP.temperature
			Req_CP_1586: Object CP
			☐ Req_Closed_Loop__Cooling
			☐ Req_Hx1_1730: Object Hx1
			☐ Req_Hx2_2190: Object Hx2
			☑ Req_CP_1586: Object CP
			☐ Req_PT1_1783: Object PT1

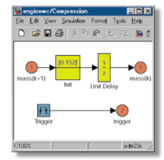
Test Driver Generation to Many Languages and Environments

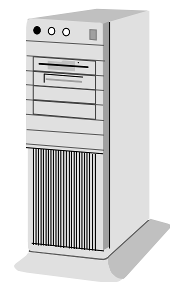


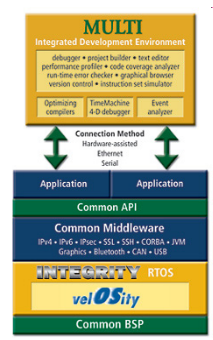
Test Driver Generation

Test Driver Languages

- SQL/ODBC/JDBC
- XML
- SOAP
- WinRunner
- JCL
- Perl
- Python
- C++
- Ada
- Basis and VB
- Custom (graphics)
- Assembler
- Java
- VectorCAST
- shell
- command languages
- emulators
- proprietary
- more . . .







Example

Matlab Simulation

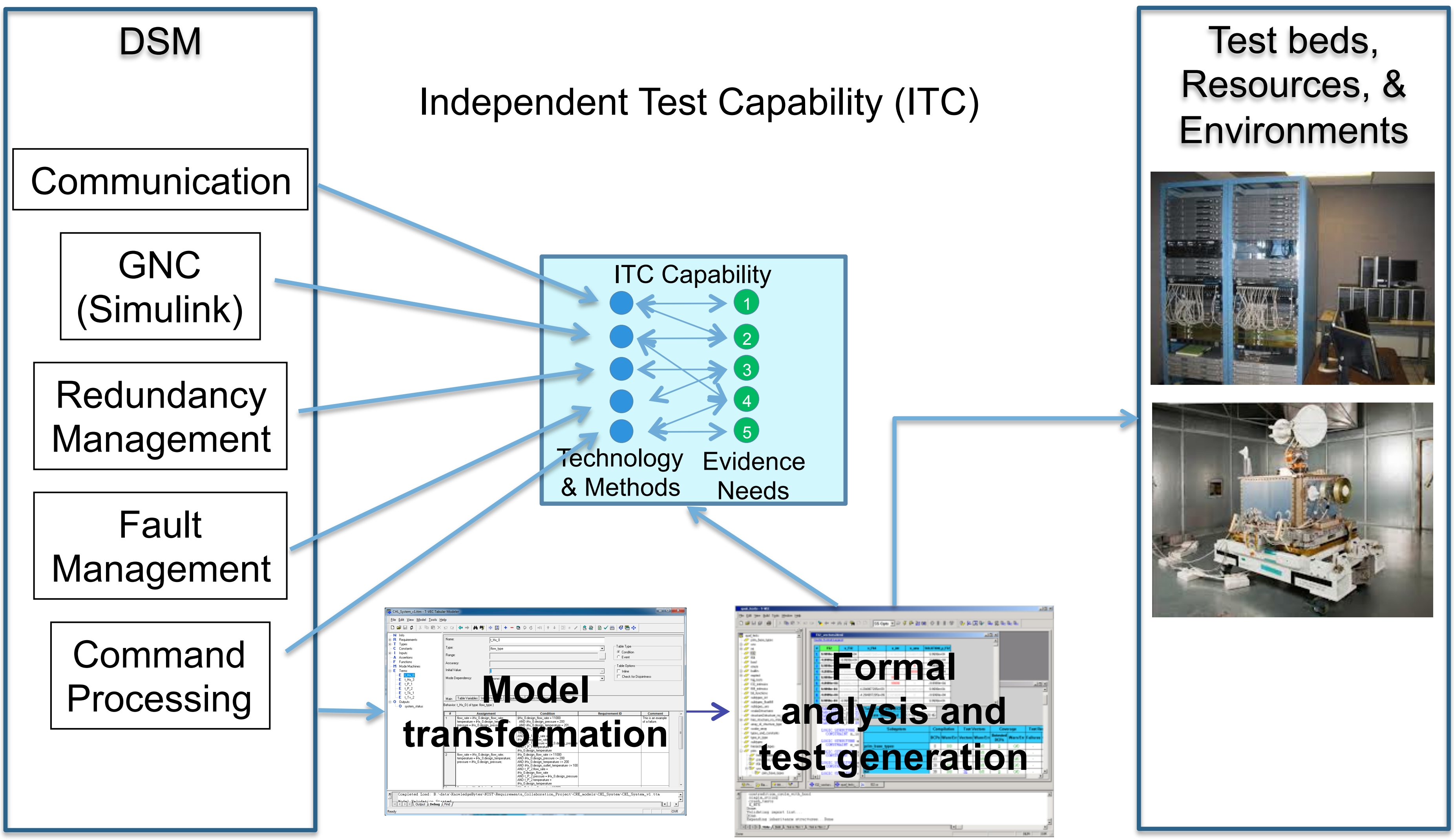
Host emulations and integrated simulation

<u>Target</u>
Windriver Workbench, GreenHills Multi IDE

Summary

- See following slide for Relevance to NASA
- DSM languages provide for graphically intuitive approaches to integrate domain-specific views across related engineering disciplines
- DSMs provide enabling technologies that integrate generators to support tool chain integration leveraging test generation and analysis tools
- DSM are evolvable allowing the modeling capabilities to keep pace with changing technologies
- Tool chains provide greater V&V coverage by leverage the most effective tools for test generation leading to greater efficiencies for V&V and software/systems engineering

DSM Transformation to Formal Analysis and Test Tools Maps to ITC Capability



Thank You

- For more information contact:
  - Mark R. Blackburn, Ph.D.
  - www.markblackburn.com
  - Blackburn@knowledgebytes.net
  - 703.431.4463

Backup

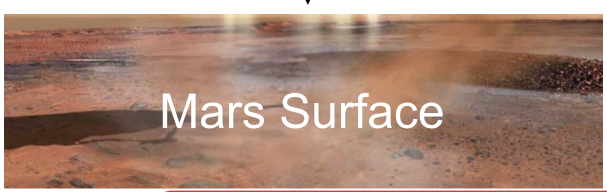
Notable Accomplishment: Discovered Mars Polar Lander Error in Fewer Than 24 Hours

(No input from Lockheed Martin) - 2000

Mars Lander Descent Path

5 KM
TDM_started = TRUE

40 Meters
CMD_disable_enable = enable



Executive

CMD_disable_enable
TDM_started

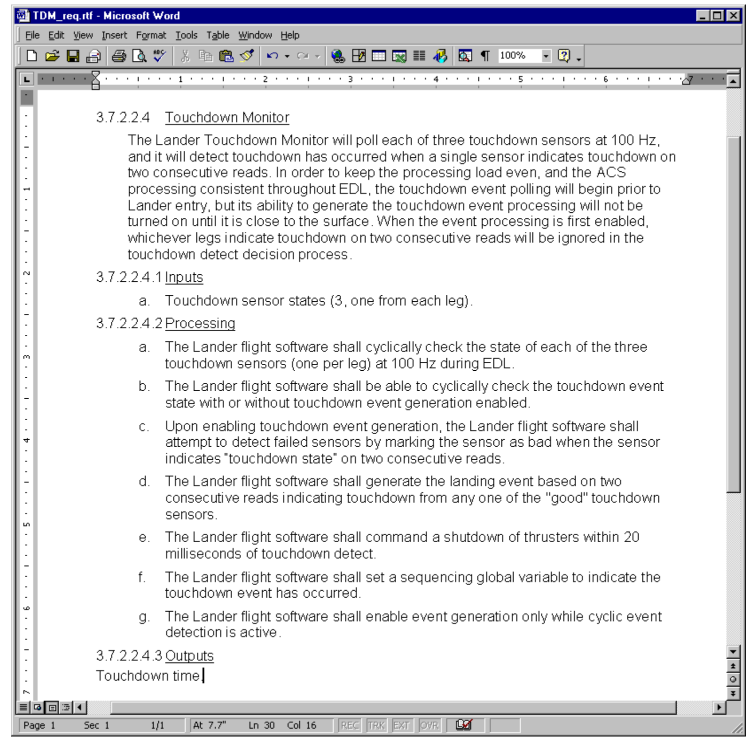
TDM Module

Sensor Value for Each of Three Legs

Leg Sensors

TDM Module Interfaces

Touchdown Monitor (TDM) Textual Requirements


TDM_req.rtf - Microsoft Word

3.7.2.2.4 Touchdown Monitor

The Lander Touchdown Monitor will poll each of three touchdown sensors at 100 Hz, and it will detect touchdown has occurred when a single sensor indicates touchdown on two consecutive reads. In order to keep the processing load even, and the ACS processing consistent throughout EDL, the touchdown event polling will begin prior to Lander entry, but its ability to generate the touchdown event processing will not be turned on until it is close to the surface. When the event processing is first enabled, whichever legs indicate touchdown on two consecutive reads will be ignored in the touchdown detect decision process.

3.7.2.2.4.1 Inputs

a. Touchdown sensor states (3, one from each leg).

3.7.2.2.4.2 Processing

a. The Lander flight software shall cyclically check the state of each of the three touchdown sensors (one per leg) at 100 Hz during EDL.

b. The Lander flight software shall be able to cyclically check the touchdown event state with or without touchdown event generation enabled.

c. Upon enabling touchdown event generation, the Lander flight software shall attempt to detect failed sensors by marking the sensor as bad when the sensor indicates "touchdown state" on two consecutive reads.

d. The Lander flight software shall generate the landing event based on two consecutive reads indicating touchdown from any one of the "good" touchdown sensors.

e. The Lander flight software shall command a shutdown of thrusters within 20 milliseconds of touchdown detect.

f. The Lander flight software shall set a sequencing global variable to indicate the touchdown event has occurred.

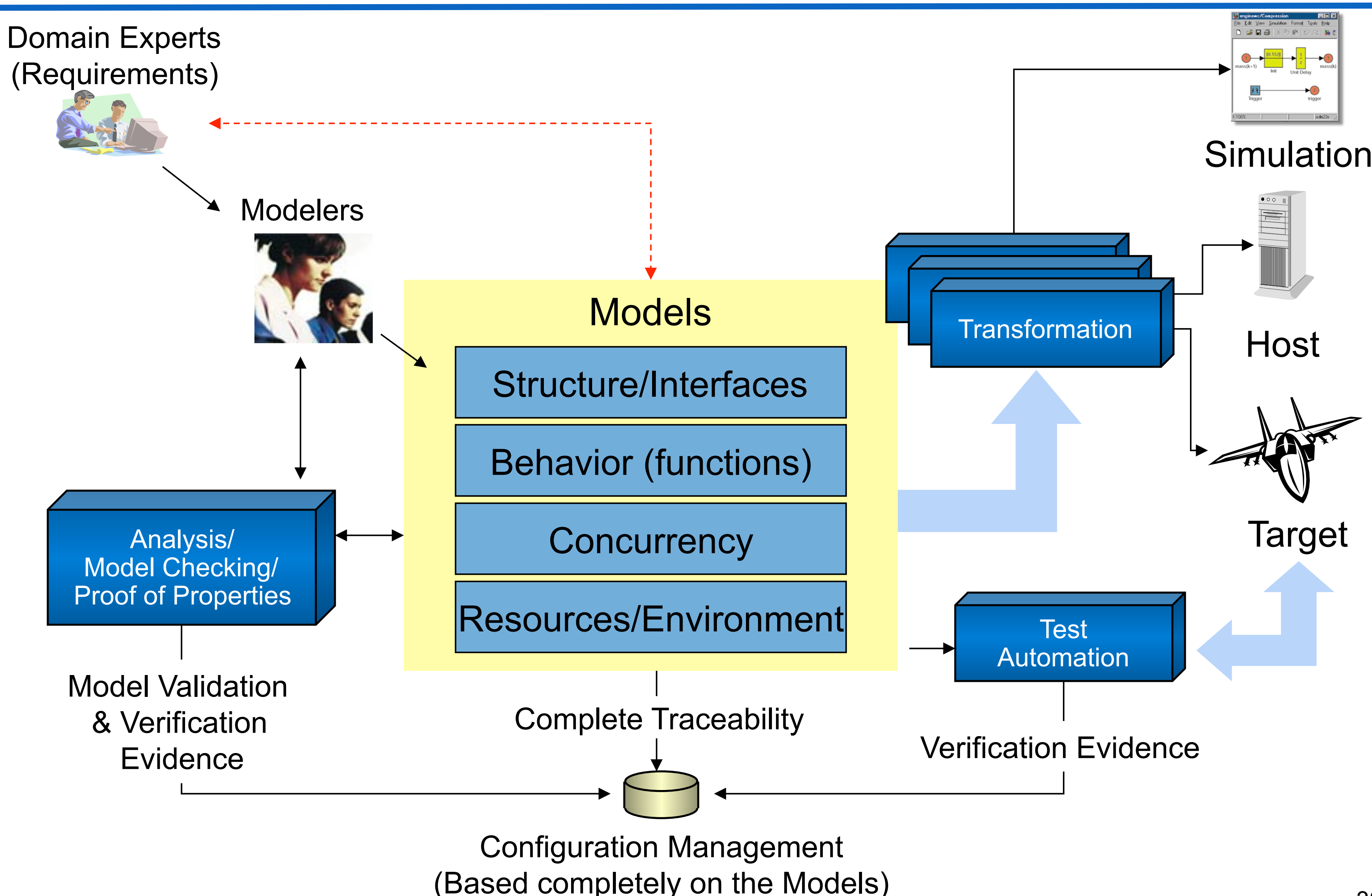
g. The Lander flight software shall enable event generation only while cyclic event detection is active.

3.7.2.2.4.3 Outputs

Touchdown time

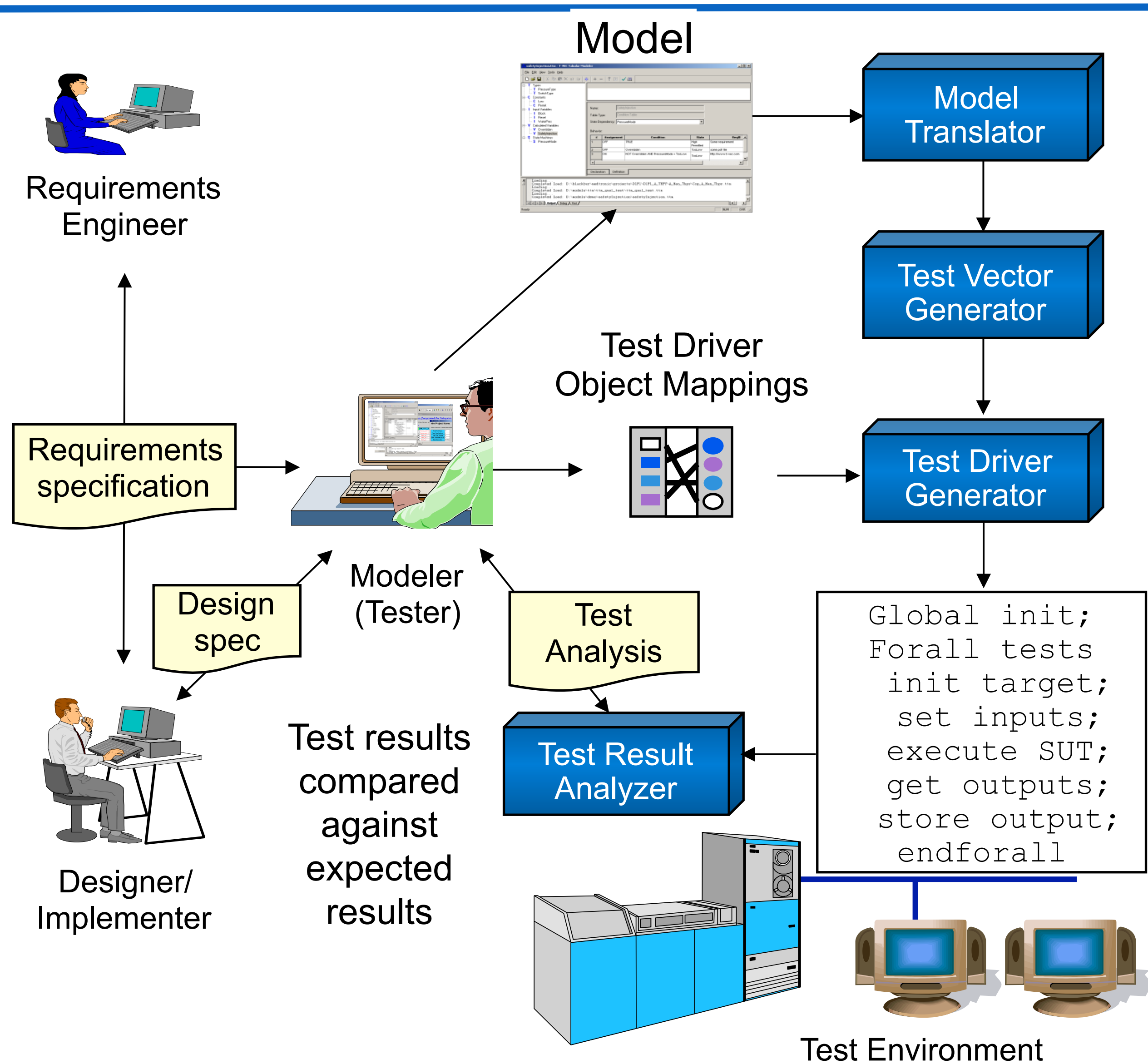
35 Million Miles at cost of $165 Million - 40 Meters from landing

“Idealized” View of Model-Driven Engineering



- Models represent requirements & design information independent of language, platform, and architecture
- Models are easier to understand than underlying code
  - Supports ability to build larger, more complex systems and improve maintainability
- Usually leverages code generation patterns proven over time – supporting more robust design
- Promotes iterative application development to better assess design/implementation early
  - Getting the right system - sometimes supported by simulation and visualization
- Tests and documentation are byproducts

Requirement-Driven Model-based Process and the Key Development Roles



Modeler captures required behavior and logical variations of data and control in terms of interfaces

Test driver schemas define a pattern for generating test scripts

<u>Test Script is Generated from Translated Model and Generated Tests</u>