VASCO

Verification Models for Advanced Human-Automation Interaction

Andreas Lüdtke
Bertram Wortelen

OFFIS

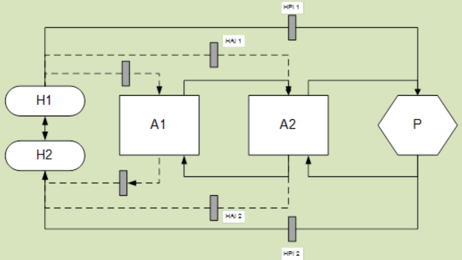
Denis Javaux

Symbio CONCEPTS & PRODUCTS sprl

Sonja Sievi

ASTRIUM AN EADS COMPANY

Human-Automation Interaction Situations



Formal Models and Analysis Questions

$$P: M_P = \langle S_P, s_{0_P}, V_P, Val_P, \delta_P \rangle,$$
$$I: M_I = \langle S_I, s_{0_I}, V_I, Val_I, \delta_I \rangle,$$
$$T: M_T = \langle S_T, s_{0_T}, V_T, Val_T, \delta_T \rangle,$$

$$\exists \phi: S_I \rightarrow S_T \forall \sigma \in Tr(M_P) \forall s_T \in S_T, s_{0_T} \overset{\sigma}{\Rightarrow} s_T \wedge s_I \in S_I, s_{0_I} \overset{\sigma}{\Rightarrow} s_I : s_T = \phi(s_I)$$

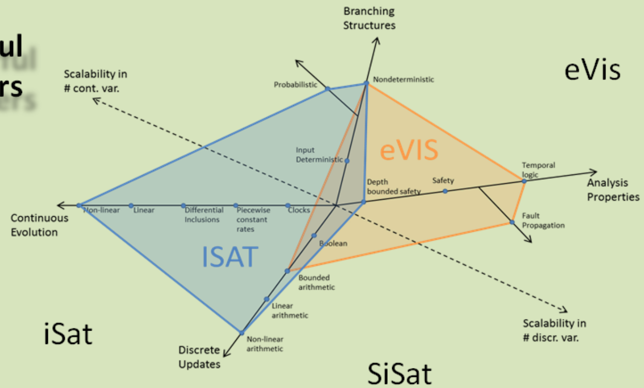
Set of Powerful Model Checkers

eVis

iSat-ODE

iSat

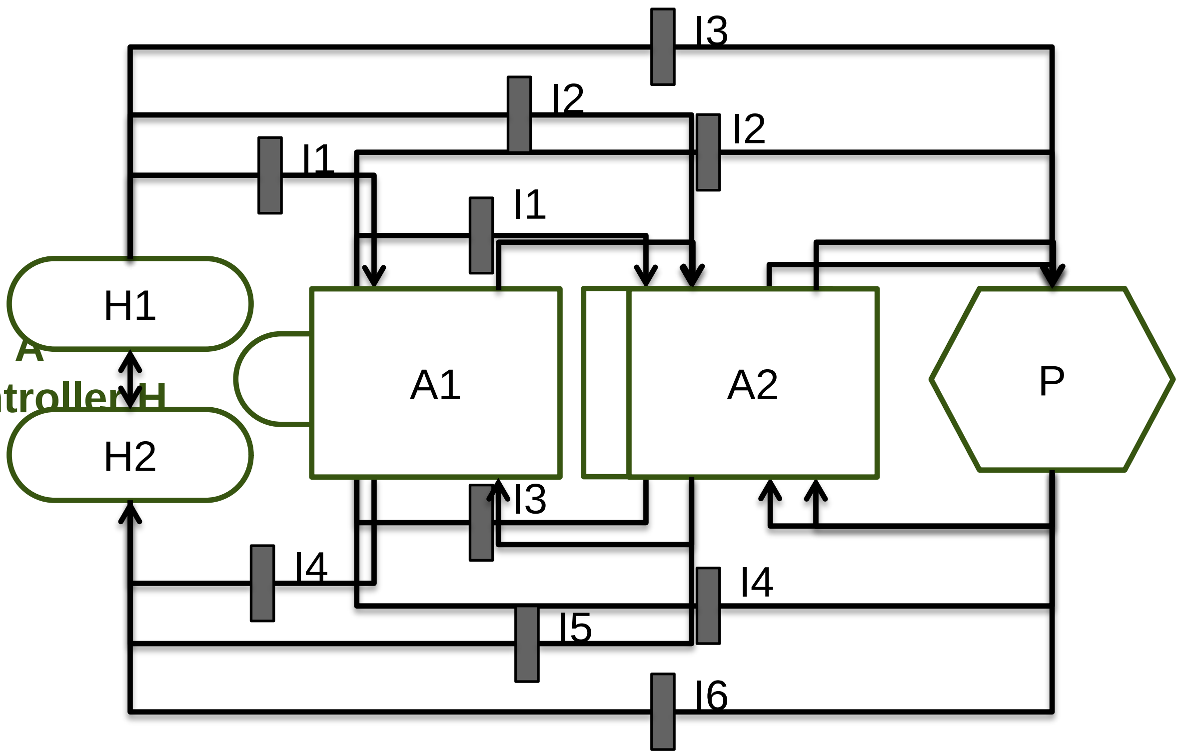
SiSat



2 Designing Human-Automation Interaction

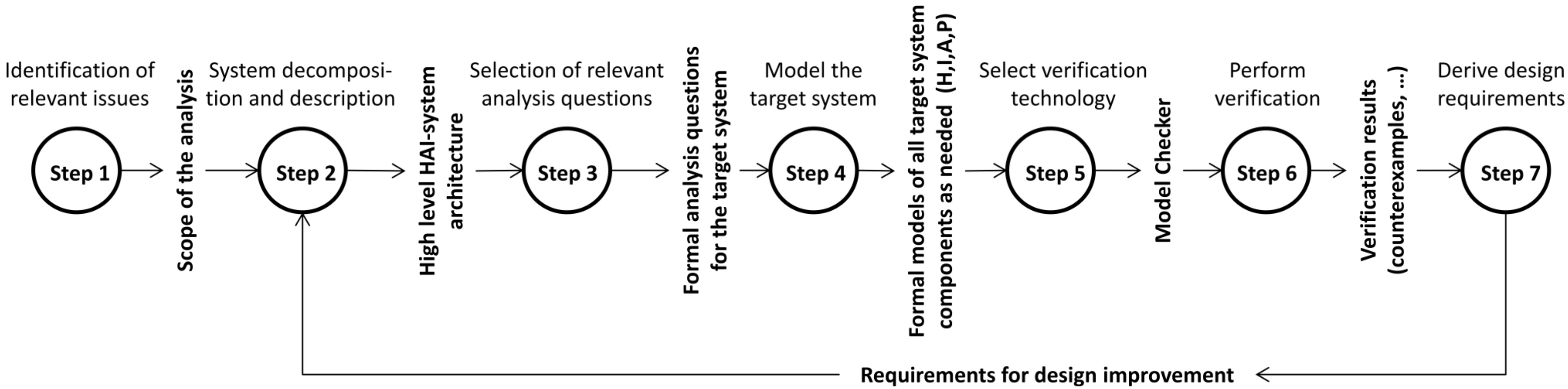
- Incorporate human models in model based system engineering!
- Consider Human-Automation Interaction early in the design process!

System of a **Process P**
controlled by an **Automation A**
supervised by a **Human Controller H**
via **Interface Elements I**



3 Verification Methodology

Stepwise approach for analyzing the Human-Automation Interaction (HAI) design

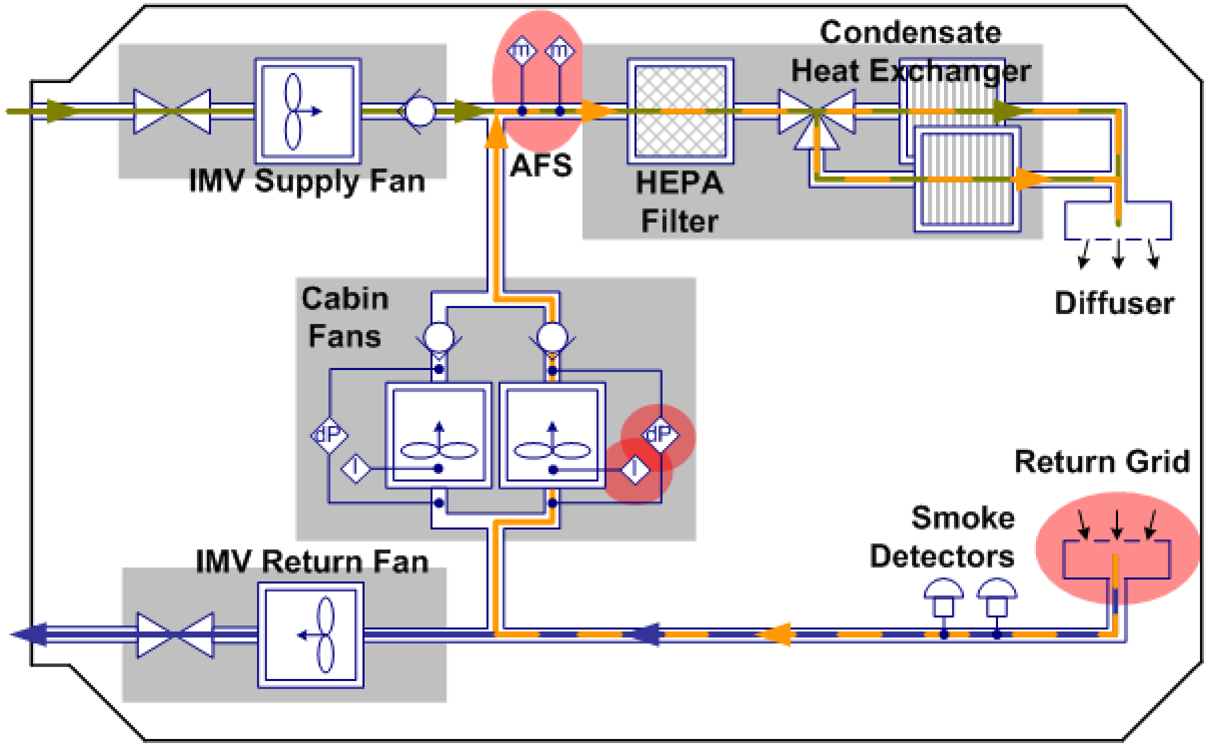


1) Identify the relevant Human-Factors issues for the target Human-Automation Interaction system
2) Decompose the human-automation target system.
3) Select relevant analysis questions, amongst the 38 questions in the AQDB, so that they cover the Human-Factors issues identified in Step 1.
4) Model the target system using adequate modelling techniques and associated editors.
5) Select adequate formal verification techniques based on the nature and complexity of the models.
6) Perform verification of the analysis questions.
7) Interpret the results and derive requirements for design improvements.

4 Case Study System

Columbus Environmental Control and Life Support (ECLS) System – Airloop Subsystem

- Inter Module Ventilation provided by Nasa via Node 2
- Fan definition
  - Power
  - Fan Speed
  - Delta Pressure
  - Input Current
- Smoke Detection
- Automation
  - Hot Redundancy of CFA1 & CFA2
  - Smoke Detectors
  - Automatic Monitoring of sensor values



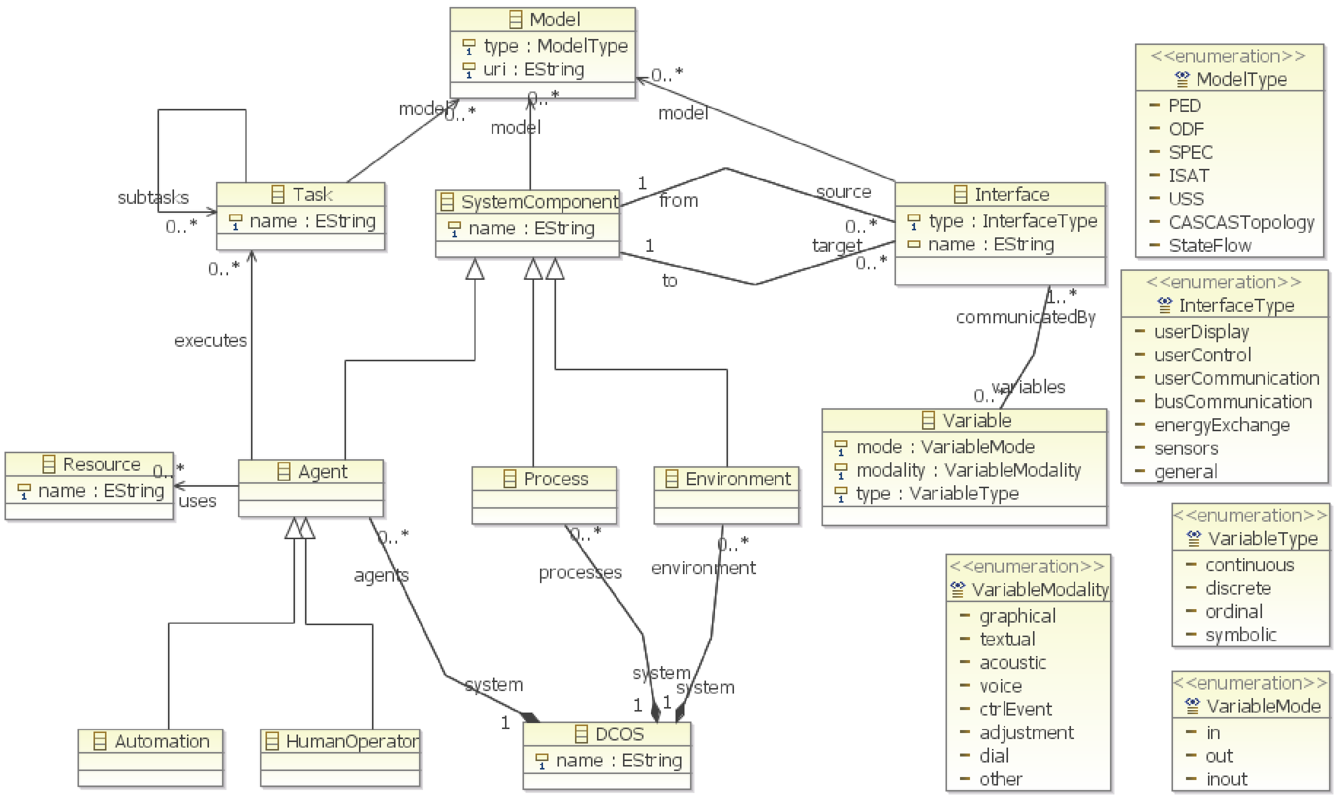
5 Step 1

Identification of relevant issues

- Applicable for existing systems with known issues
- Columbus Flight Note System
  - Collection of all unplanned incidents and anomalies occuring during operations
- Cabin Air Return Grid Glooging Caution
  - Requires mandatory crew involvement
- CFN5115 – Cabin Air Return Grid Clogging
  - ***Activity*** Preparation of maintenance task “Cleaning of Smoke Detector 2”
  - ***Procedure*** “ESA SODF: ECLSS: NOMINAL: 2.102 Prep for COL1D1 Rotate” procedure
- Hints to problems with inconsistent automation behaviour and mode awareness

6 STEP 2: System Decomposition and Description

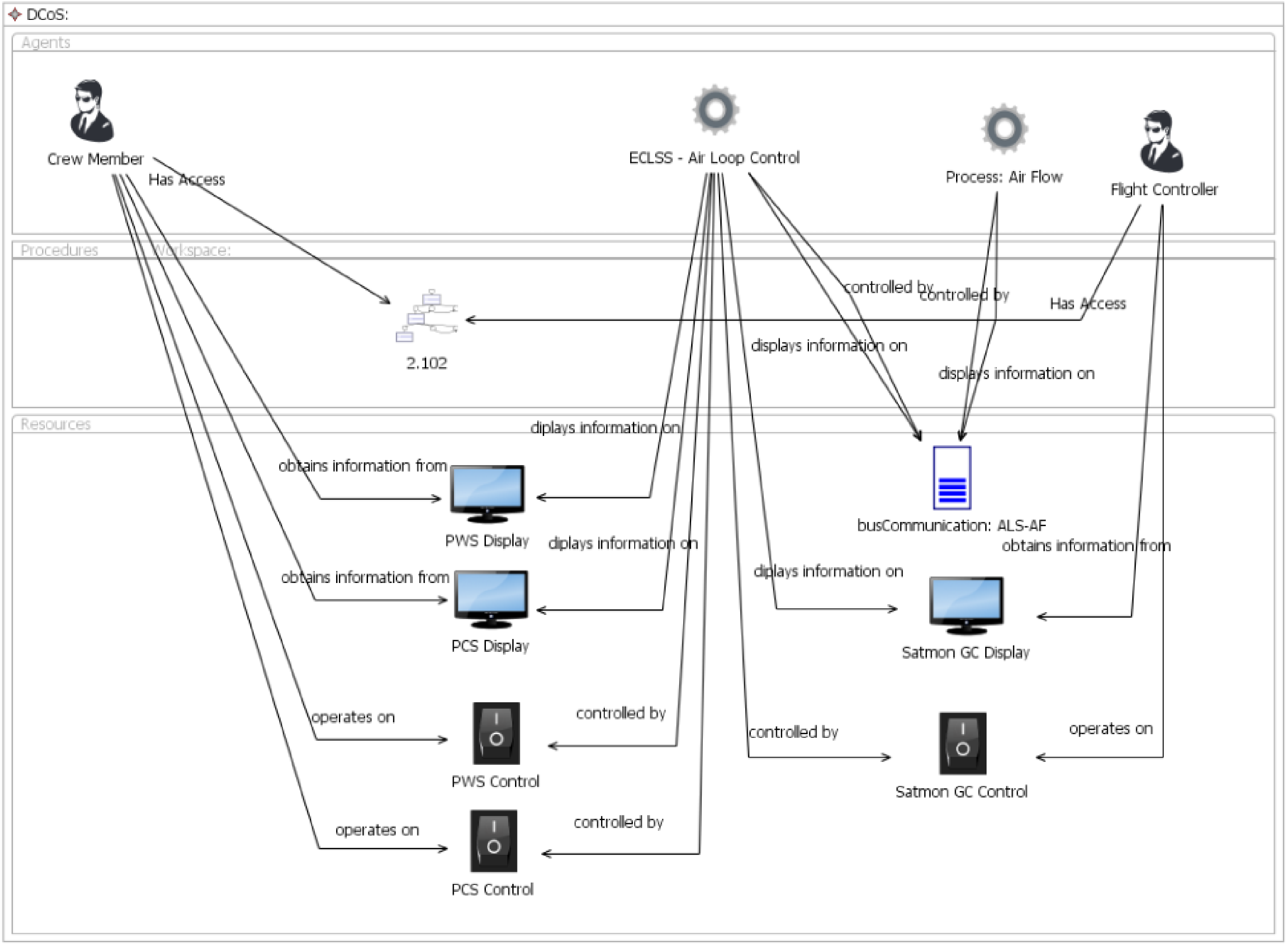
System Description Language



System Description Language

7 STEP 2: System Decomposition and Description

Case Study: Global (sub)System



8 STEP 3: Selection of Relevant Analysis Questions

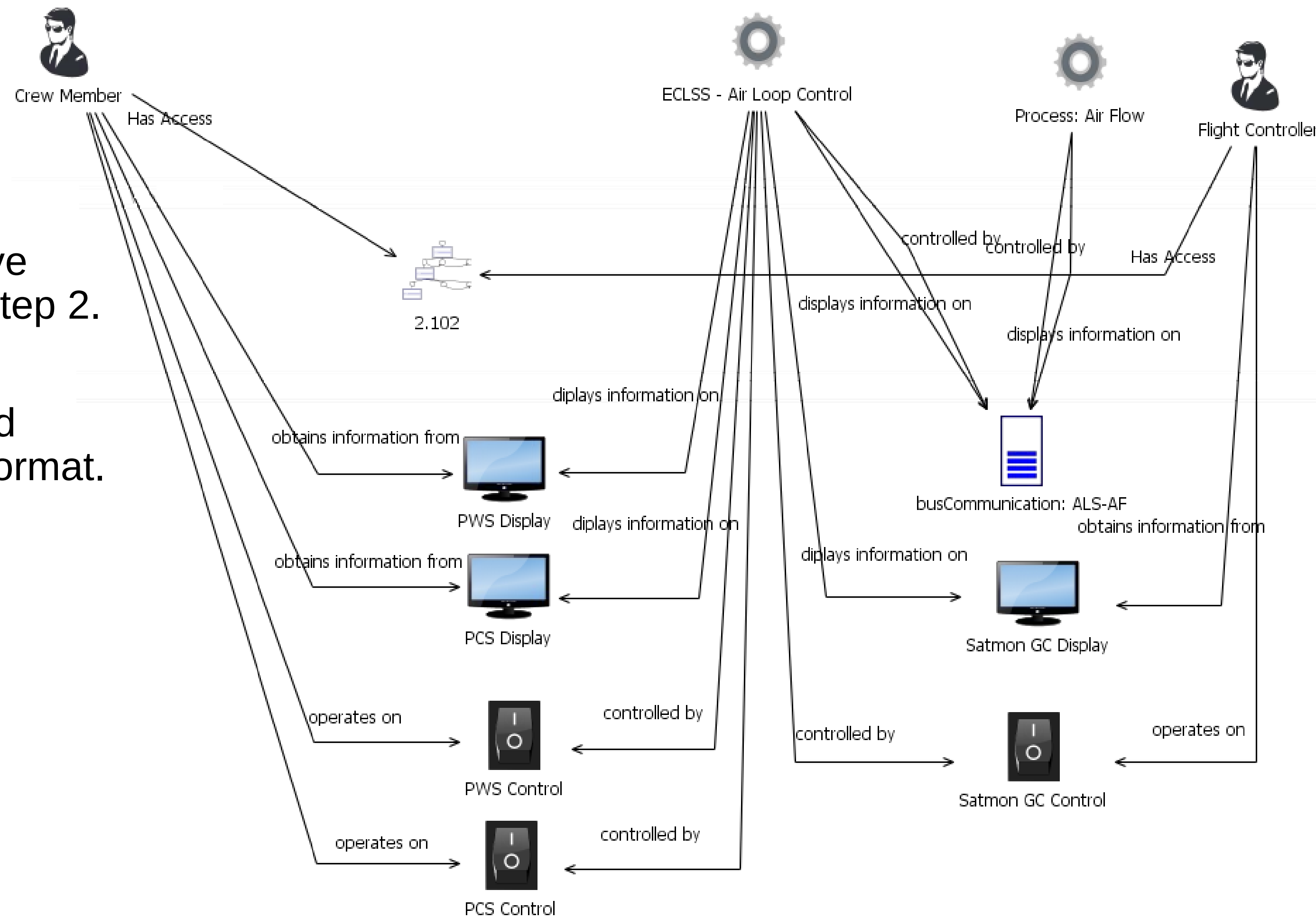
- 7 categories
  - Information on Automation States and Behaviours
  - Issuing Commands towards Automation
  - Understanding Automation Complexity Issues
  - Situation Awareness and Out-of-the-Loop problem
  - Workload changes
  - Vigilance
  - Skill Degradation
  - Trust

- 6 questions selected for case study
  - C1.3: is the information on automation state sufficient to interact efficiently with automation?
  - C1.4: does a given action cause consistent effects?
  - C1.5: Is the operator informed when state transitions (e.g., mode transitions) occur?
  - C2.6: does a given action provide feedback?
  - C3.3: can the automation, as presented on the UI, be considered as a deterministic state machine for the operator?
  - C3.9: is the operator able to detect whether equipment or process is in abnormal mode?

9 Step 4: System Modelling

General Idea

- Identify input models for each of the components that have been addressed in Step 2.
- Need to be translated into model checker format.



10 Step 4: System Modeling

Human

- Task
- Operations Data File (ODF) Standards
  - 2 representation formats
    - XML
    - Textual/graphical

5. DEACTIVATING ACTIVE CTCU

PWS

ECLSS: ECLSS Commands

ECLSS Commands

'Activation Commands'

cmd CTCU1(2) Deactivation **Execute**

PWS

ECLSS: CTCU 1(2)

CTCU 1(2)

Verify CTCU1(2) Pwr – Off
Verify Health Status – Error

Verify Health Status – Background white

On Crew GO after preparation for COL1D1 rack rotate is complete.

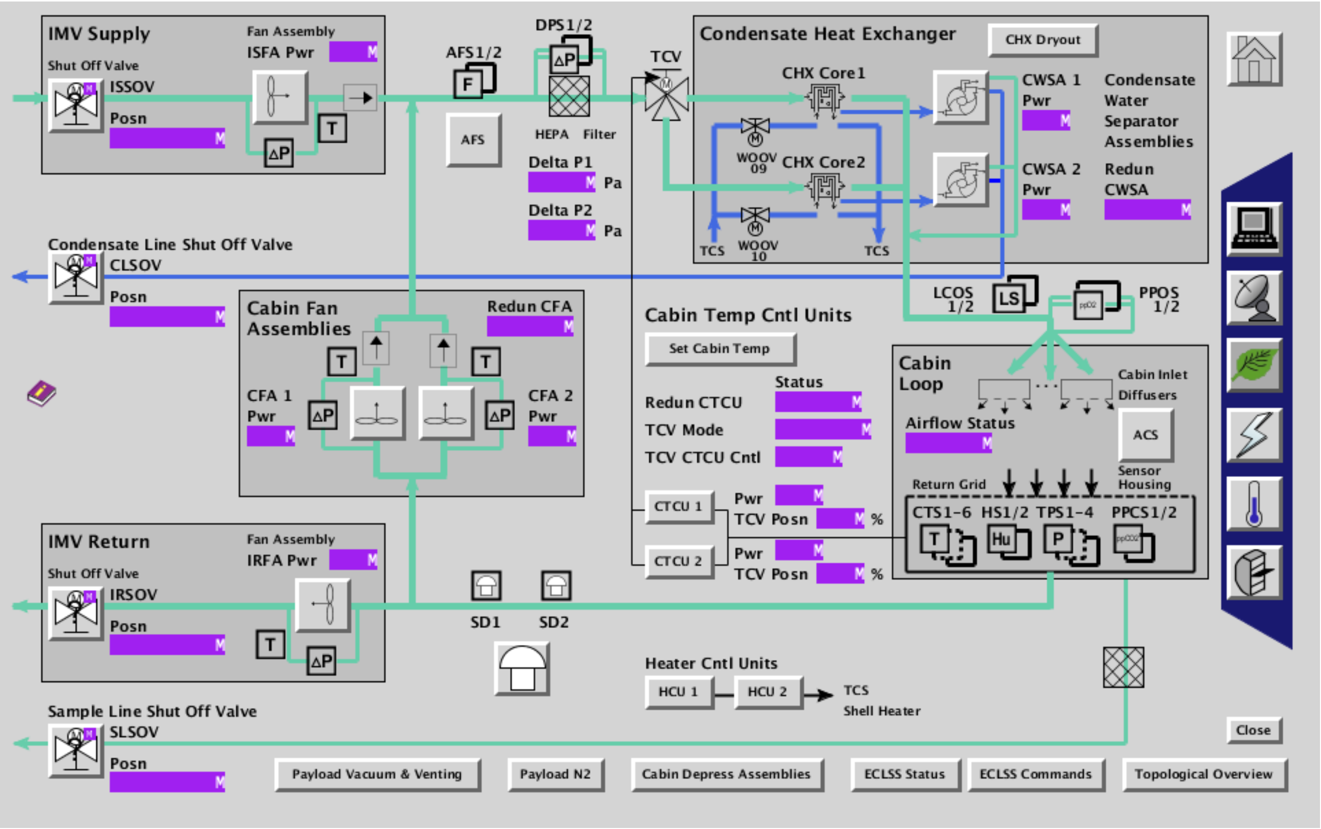

```xml
<Step stepId="i1415">
  <StepTitle>
    <StepNumber>5</StepNumber>
    <Text>DEACTIVATING ACTIVE CTCU</Text>
  </StepTitle>
  <StepContent itemId="i1416">
    <LocationInfo><Location><Text>PWS</Text></Location></LocationInf
    <NavInfo singleLine="false">
      <NavPath><Text>ECLSS</Text></NavPath>
      <NavPath><Text>ECLSS Commands</Text></NavPath>
      <DisplayName><Text>ECLSS Commands</Text></DisplayName>
      <GraphicalLocationIndicator><Text>Activation Commands</Text></
    </NavInfo>
  </StepContent>
  <StepContent itemId="i1420">
    <Instruction>
      <CmdCallout cmdType="cmdExecute">
        <CmdAction>
          <Text>CTCU</Text>
          <ChoiceReference refId="ctcu" refIndex="0"/>
          <Text> Deactivation</Text>
        </CmdAction>
      </CmdCallout>
    </Instruction>
  </StepContent>
  <StepContent itemId="i1421">
  <LocationInfo><Location><Text>PWS</Text></Location></LocationInfo>
  <NavInfo singleLine="false">
    <NavPath><Text>ECLSS</Text></NavPath>
    <NavPath>
      <Text>CTCU</Text>
```

Step 4: System Modeling

User Interface

- User Interfaces for ISS crew
- ISS display designs are based on the Displays and Graphics Commonality Standard (DGCS)
- Representation format:

  Unified Synoptic System (USS)



12 Step 4: System Modeling

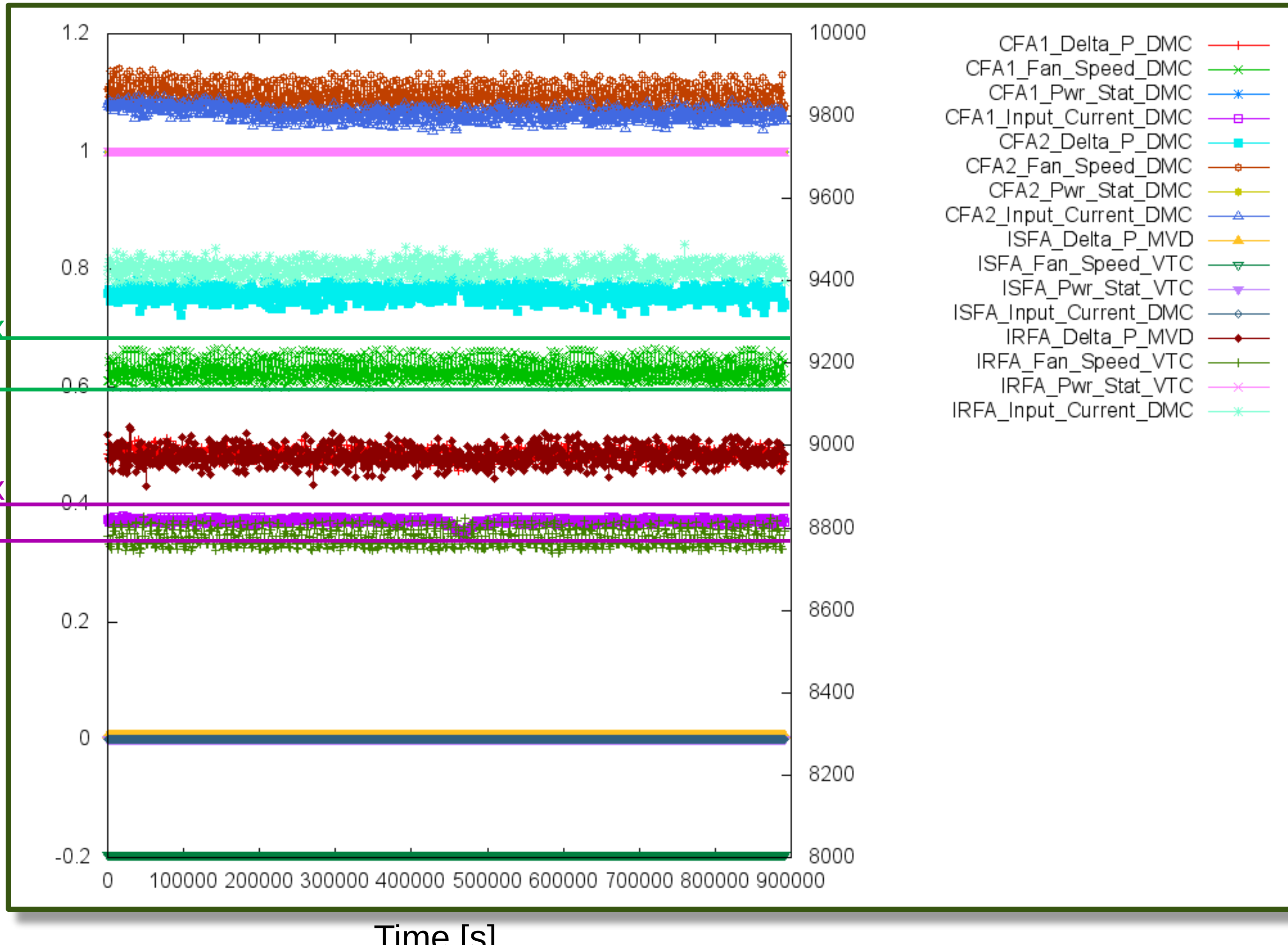
Automation: Air Flow Control

- Use design and implementation models
- Low-detail model for the case study:
  - Mode definitions and transitions
    - Fan modes (On/Off) and speeds for IRFA, ISFA, CFA1 and CFA2
  - Warning system
    - Warning definitions
  - Flight Automated Procedures (FLAPs)
    - As needed by investigated procedure

13 Step 4: System Modeling

Process: Air Flow

- Air Flow Actuators: ISFA, IRFA, CFA1 and CFA2
- No explicit model available

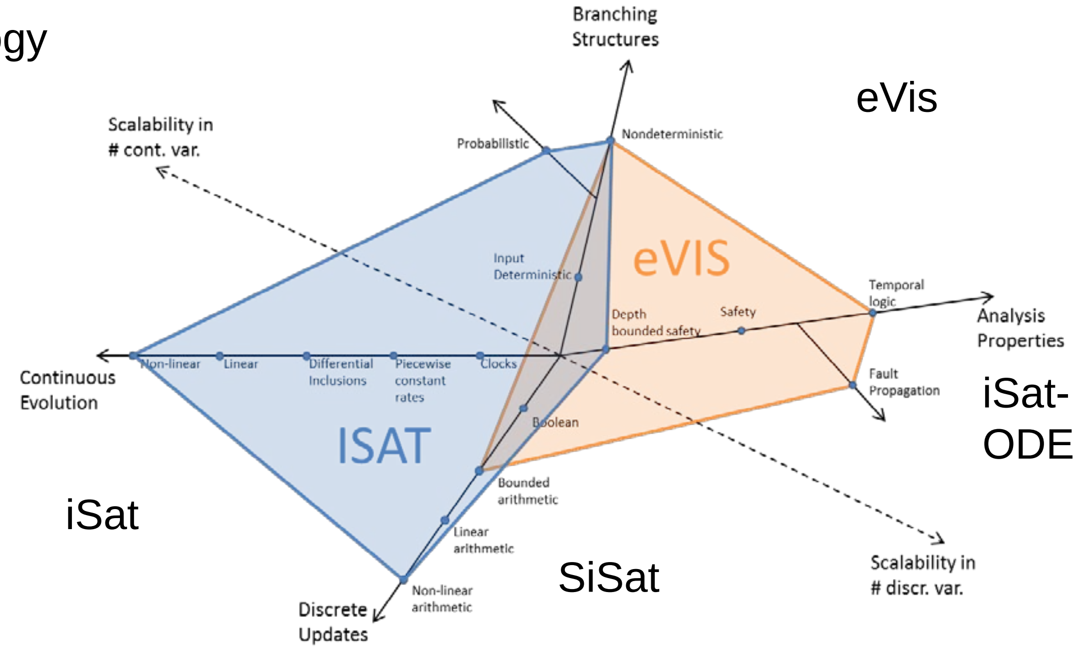


- Database with all telemetry data since start Columbus operation in 2008
- Used to train a model

Step 5: Verification Technology

General Idea

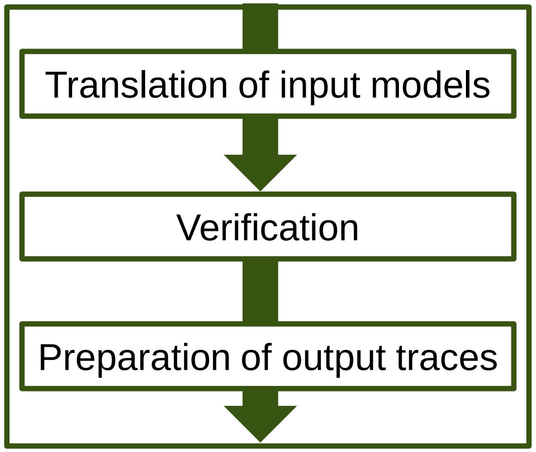
- Select suitable verification technology
- Model properties
- Input model formats
- iSAT for the case study
  - BMC for Boolean combinations of linear and non-linear arithmetic constraints over real- and integer-valued variables.



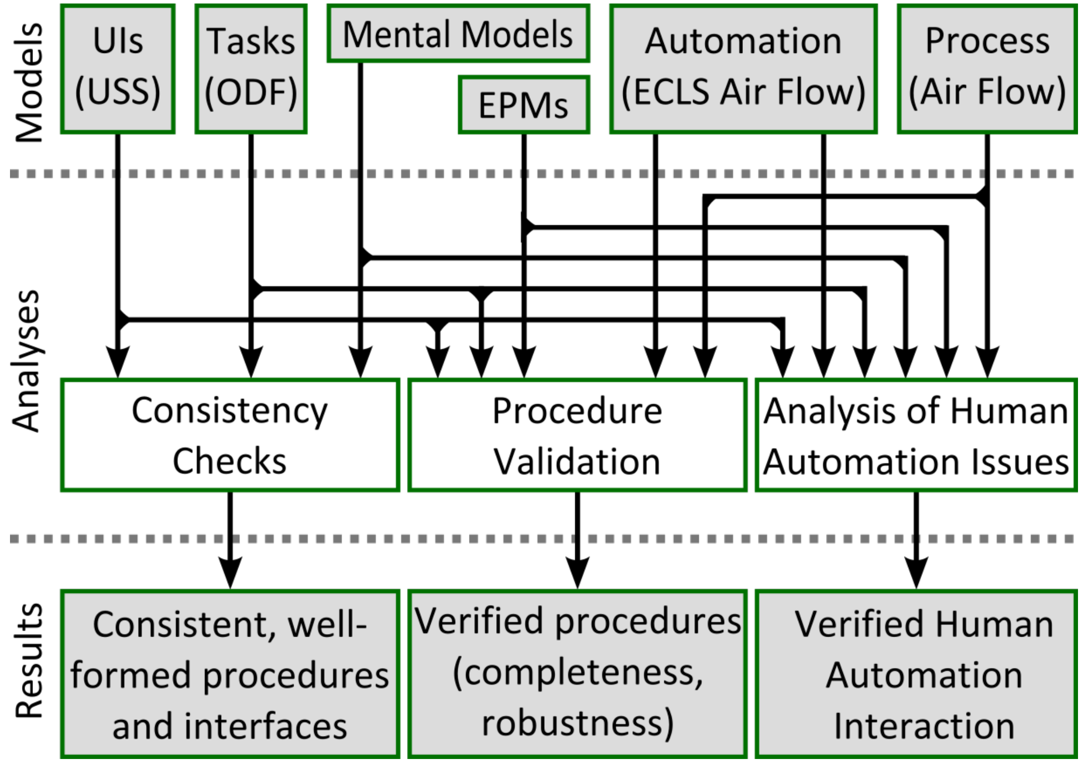
Step 6: Verification

General Idea

- General approach



- Model based analyses enables
  - Consistency checks
  - Procedure validation



Step 6: Verification

Robustness Analysis based on Error Production Mechanisms (EPMs)

- Analysis question: Is it possible, that the system gets into a critical state, if the operator makes one (or two, three, …, n) plausible errors?
- Inject human errors into nominal procedures
  - Error Production Mechanisms (EPMs)
  - describe error prone structures, which might lead to operator errors.
- For each instantiation of an EPM, the procedure is modified in a way that injects an error:

```
(errorFlagX = 0) -> (<nominal task execution>);
(errorFlagX = 1) -> (<incorrect task execution>);
```

- Case Study:
  - **Error of omission**
    - Sequence of common instructions with no direct effect or only little effect
    - Possibility to omit a step (except the first one)
  - **Step confusion**
    - Reference to a GUI element in an instruction: If similar GUI elements exists, the operator might erroneously use the wrong one.

17 Step 6: Verification

Procedure Validation – Error injection

- Error of omission
- List of inhibit monitoring actions
- Error containment is different for PCS and PWS display system.
- PCS more robust against this kind of error

PCS

3.2 Inhibiting VTC Monitoring for ISFA

COL: ECLSS: Air Loop: ISFA
COL IMV Fans
'IMV Supply Fan Assembly'

cmd Fan Speed Monitoring – Inh
cmd Fan dP Monitoring – Inh
cmd Fan Temp Monitoring – Inh

Verify Speed Mon Ena – Inhibited
Verify dP Mon Ena – Inhibited
Verify Temp Mon Ena – Inhibited

PWS

3.1 Inhibiting DMS Monitoring for ISFA

ECLSS: ISFA
ISFA
right click Delta P
pick Change Monitoring Values
Monitoring ISFA_Delta_P_MVD
cmd Inhibit Monitoring **Execute**
Verify Current Value – Background white

Inhibit monitoring for ISFA_Delta_P_MVD

ISFA
right click Input Current
pick Change Monitoring Values
Monitoring ISFA_Input_Current_DMC
cmd Inhibit Monitoring **Execute**
Verify Current Value – Background white

Inhibit monitoring for ISFA_Input_Current_DMC

18 Step 6: Verification

Procedure Validation – Error injection

- Step confusion
- Nearly identical groups of GUI elements
- Potential for confusion

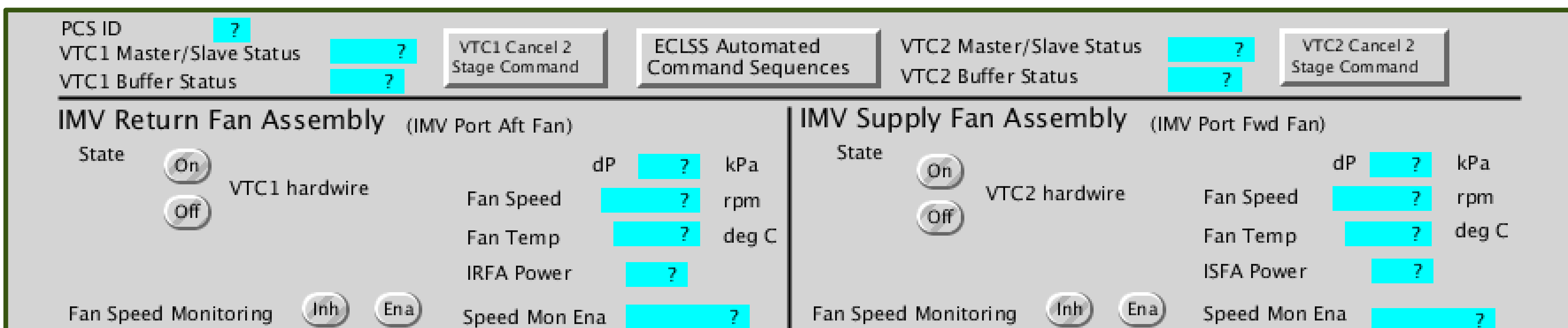
3.2 Inhibiting VTC Monitoring for ISFA

PCS

COL: ECLSS: Air Loop: ISFA
COL IMV Fans
'IMV Supply Fan Assembly'

cmd Fan Speed Monitoring – Inh
cmd Fan dP Monitoring – Inh
cmd Fan Temp Monitoring – Inh

Verify Speed Mon Ena – Inhibited
Verify dP Mon Ena – Inhibited
Verify Temp Mon Ena – Inhibited



Step 6: Verification

Analysis Questions for Human-Automation Issues

- General Idea
  - Group analysis questions into sets of questions which can be addressed in similar ways

- Q1) Does the UI present all the information needed by the human agent?
- Q2) Is the information on the UI well presented?
- Q3) Is a component of the HAI system deterministic from the human agents point of view?
- Q4) Does a state machine present some required temporal properties?
- Q5) Is a task cognitively complex?
- Q6) Is the human able to build a predictive mental model of something?
- Q7) Does the overall human-automation situation present some structural properties?

20 # Step 6: Verification

Analysis Questions – C1.4 – Does a given action cause consistent effects?

- Parallel composition of two identical systems
- Synchronous procedure execution.
- Is it possible to observe different effects?

```
((A_p2_102_state = P2_102_STATE_i1325) and (B_p2_102_state = P2_102_STATE_i1325))
  -> ((A_LOSS_CFA1 = B_LOSS_CFA1) and (A_LOSS_CFA2 = B_LOSS_CFA2) and
      (A_LOSS_ISFA = B_LOSS_ISFA) and (A_LOSS_IRFA = B_LOSS_IRFA) and
      (A_RETURN_GRID_CLOGGING = B_RETURN_GRID_CLOGGING));
```

- Result: Warning occurred in one system copy and not in the other dependent on the initial state of the system
- Problem: Meaningful definition of effects

CFA1@9200, CFA-off, IRFA@8784, ISFA@9960

```
CFA1_Pwr_Stat_DMC = 1;
CFA1_SetSpeed_DMC = 9200;
CFA2_Pwr_Stat_DMC = 0;
CFA2_SetSpeed_DMC = 8900;
IRFA_Pwr_Stat_DMC = 1;
IRFA_SetSpeed_VTC= 8784;
ISFA_Pwr_Stat_DMC= 1;
ISFA_SetSpeed_VTC= 9960;
MonEnab_CFA2_Input_Current_DMC = 0;
MonEnab_CFA2_Fan_Speed_DMC = 0;
MonEnab_CFA2_Delta_P_DMC = 0;
```

CFA1@9200, CFA-off, IRFA-off, ISFA@9960

```
CFA1_Pwr_Stat_DMC = 1;
CFA1_SetSpeed_DMC = 9200;
CFA2_Pwr_Stat_DMC = 0;
CFA2_SetSpeed_DMC = 8900;
IRFA_Pwr_Stat_DMC = 0;
IRFA_SetSpeed_VTC= 8784;
ISFA_Pwr_Stat_DMC= 1;
ISFA_SetSpeed_VTC= 9960;
MonEnab_CFA2_Input_Current_DMC = 0;
MonEnab_CFA2_Fan_Speed_DMC = 0;
MonEnab_CFA2_Delta_P_DMC = 0;
MonEnab_IRFA_Input_Current_DMC = 0;
MonEnab_IRFA_Delta_P_VTC = 0;
MonEnab_IRFA_Fan_Speed_VTC = 0;
```

Step 7: Derivation of Design Requirements

Case Study

- Analyse the verification trace and identify countermeasures
- If the same or similar problems occur often:
  - Analyze and potentially improve the socio-technical system that is designing the HAI system

- Case study:
  - Separate actions and verify instructions if possible
  - Do not use ambiguous labels on the same display
  - Define valid initial state for procedure

22 Recommendations
Case Study

- Future work
  - Use the methodology during design phase of a system
  - Reuse design and implementation models of the automation
  - Increase level of detail
    - Especially more detailed time model

- General recommendations
  - Consider human-automation Interaction early in the design process
    - Design the entire human-automation interaction system, not just the technical system
  - Incorporate human models in model based system engineering
    - task models, mental models, EPM
  - Work towards a Reference Technology Platform
    - Enable re-use of models and better chain of tools and workflows
  - Do not rely solely on formal verification methods. It complements other methods like human-in-the-loop simulations.

23 Thank you.

Dr. Andreas Lüdtke
luedtke@offis.de

Bertram Wortelen
wortelen@offis.de

Denis Javaux
denis.javaux@symbio.pro

Sonja Sievi
sonja.sievi@astrium.eads.net

Step 6: Verification

Procedure Validation

- Execution of Procedure 2.102 in the air flow configuration that was active in the CFN5115
- Analysis question: *Does a Warning occur? Are any Verify or Check instructions violated?*
- Initialization:

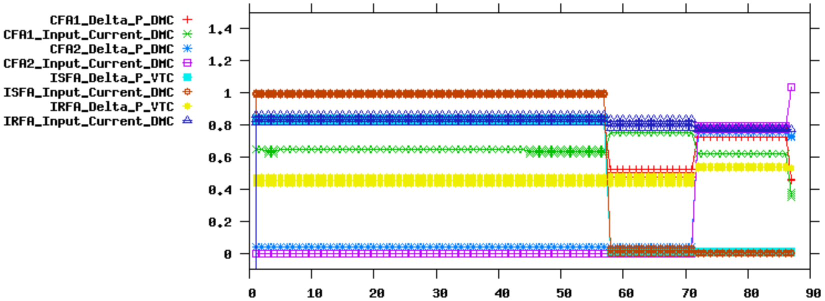
```
CFA1_Pwr_Stat_DMC = 1;
CFA1_SetSpeed_DMC = 9200;
CFA2_Pwr_Stat_DMC = 0;
CFA2_SetSpeed_DMC = 8900;
IRFA_Pwr_Stat_DMC = 1;
IRFA_SetSpeed_VTC= 8784;
ISFA_Pwr_Stat_DMC= 1;
ISFA_SetSpeed_VTC= 9960;
MonEnab_CFA2_Input_Current_DMC = 0
MonEnab_CFA2_Fan_Speed_DMC = 0
MonEnab_CFA2_Delta_P_DMC = 0
```

Fan Assemblies are in Nominal Configuration 1:

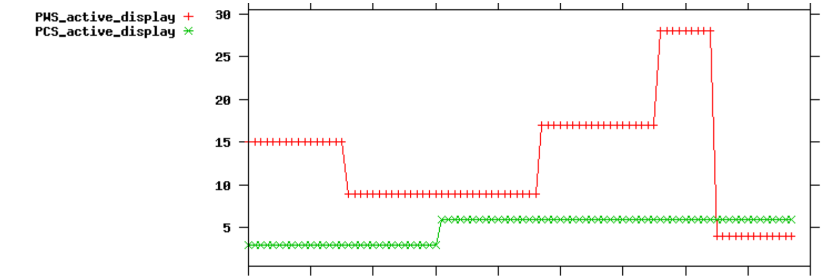
CFA1 @ 9200 ISFA @ 9960
CFA2 - Off IRFA @ 8784

CFA2 is Off.
Therefore Monitoring for its sensor values is turned off

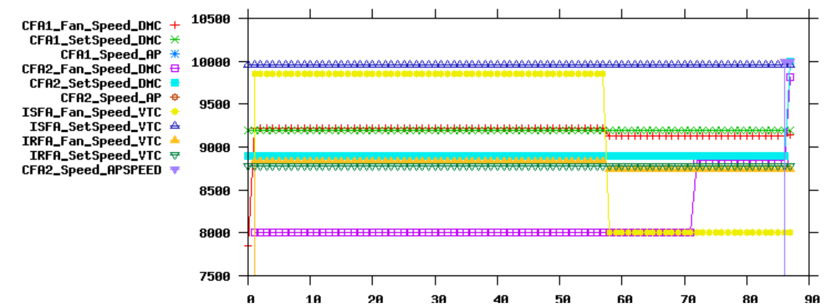
- Result: Return Grid Clogging Warning occurred like it was reported
- Starting in a different initial state with degraded IRFA did not show the warning (like reported)



Sensor values the process m[...] for some sens[...] of the fan assemblies

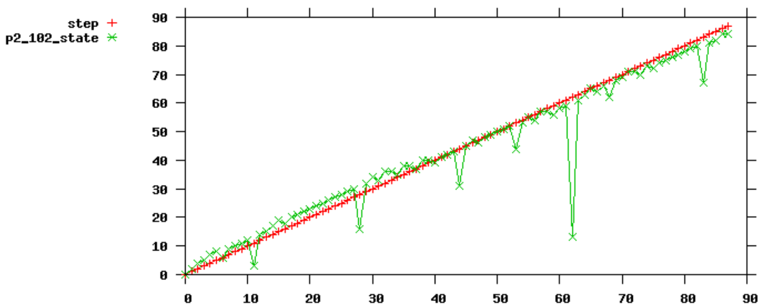


variables show[...] on which displ[...] the operator is working on (for PCS and P[...] (symbolic varia[...]

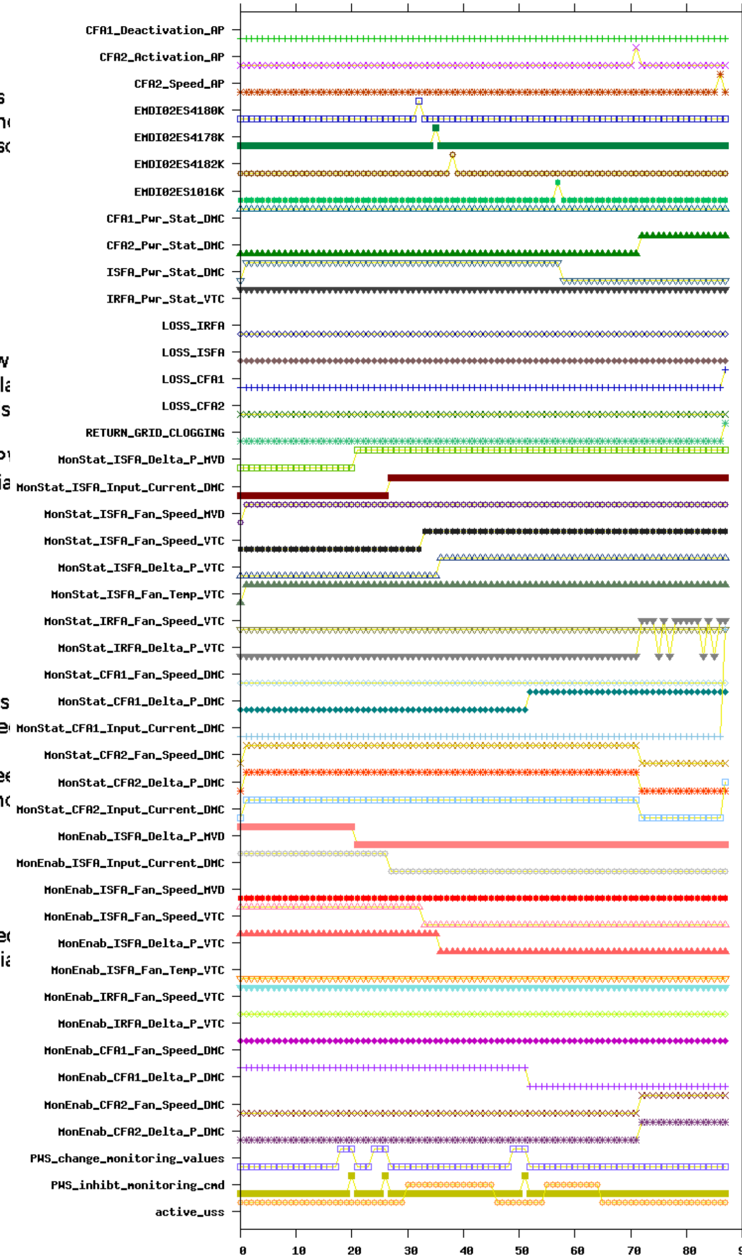


Fan Speeds (Target Speeds and Actual Spe[...]

(Fans with Spee[...] of 0 rpm are n[...] shown)



State of proce[...] (symbolic vari[...]



Flags showing that a FLAP has been triggered

Power state of fan assemblies 1=on, 0=off

Flags showing that there is a failure in one fan assemblie or a return grid clogging

Monitoring state of monitored fan assembly sensor values
0 = in limits
1 = not monitored
2 = nom.HighViol.
3 = limit Violation
4 = nom.LowViol.
5 = dang.HighViol.
6 = dang.LowViol.
7 = Invalid Value
8 = Undefined

Boolean flag showing whether monitoring is enabled for a specific sensor value

Flags for context menu actions

Current display system.
1=PCS
2=PWS

26 Step 6: Verification

Procedure Validation – Error injection

- Rerun procedure validation
- Enabled errors introduced by EPMs
- E.g.: Analyze **robustness of procedure** if up to 1 error is made:
  - Init

    ```
    errors = errorFlag1 + errorFlag2 + … + errorFlagN;
    errors <= 1;
    ```
  - Transition Relation

    ```
    errorFlag1‘ = errorFlag1;
    ...
    errorFlagN‘ = errorFlagN;
    ```

27 Step 6: Verification

Procedure Validation – Error injection

- Error of omission
- Error containment is different for PCS and PWS.
- PCS more robust for this kind of error

3.2 Inhibiting VTC Monitoring for ISFA

PCS

COL: ECLSS: Air Loop: ISFA
COL IMV Fans
'IMV Supply Fan Assembly'

cmd Fan Speed Monitoring – Inh
cmd Fan dP Monitoring – Inh
cmd Fan Temp Monitoring – Inh

Verify Speed Mon Ena – Inhibited
Verify dP Mon Ena – Inhibited
Verify Temp Mon Ena – Inhibited

3.1 Inhibiting DMS Monitoring for ISFA

PWS

ECLSS: ISFA
ISFA
right click Delta P
pick Change Monitoring Values
Monitoring ISFA_Delta_P_MVD
cmd Inhibit Monitoring **Execute**
Verify Current Value – Background white

Inhibit monitoring for ISFA_Delta_P_MVD

ISFA
right click Input Current
pick Change Monitoring Values
Monitoring ISFA_Input_Current_DMC
cmd Inhibit Monitoring **Execute**
Verify Current Value – Background white

Inhibit monitoring for ISFA_Input_Current_DMC

28 Step 6: Verification

Procedure Validation – Error injection

- Step confusion
- Nearly identical groups of GUI elements
- Potential for confusion

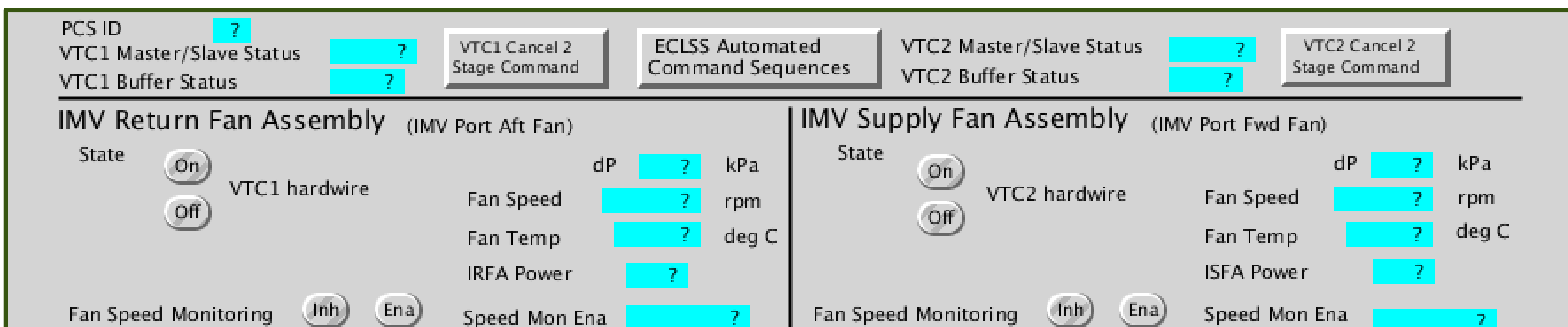
3.2 Inhibiting VTC Monitoring for ISFA

PCS

COL: ECLSS: Air Loop: ISFA
COL IMV Fans
'IMV Supply Fan Assembly'

cmd Fan Speed Monitoring – Inh
cmd Fan dP Monitoring – Inh
cmd Fan Temp Monitoring – Inh

Verify Speed Mon Ena – Inhibited
Verify dP Mon Ena – Inhibited
Verify Temp Mon Ena – Inhibited



Step 6: Verification

Analysis Questions – C1.3

- C1.3: is the information on automati

$p_{\text{mode automation}}$: (ISFA_mode = W)
(CFA2_mode = Z)

$q_{\text{mode mental}}$: (ISFA_mental_mo
(CFA1_mental_mo

r: (steps_taken < n)

- It doesn't take the operator more tha

Identify current air loop target configuration

1. Obtain CFA1 mode
2. Obtain CFA2 mode
3. Obtain ISFA mode
4. Obtain IRFA mode
5. Derive air loop target configuration
    1. Recall CFA1 mode
    2. Recall CFA2 mode
    3. Recall IRFA mode
    4. Recall ISFA mode
    5. Parallel evaluation:
        1. If (CFA1_mode = On@9200) and
           (CFA2_mode = Off) and
           (IRFA_mode = On@8784) and
           (ISFA_Pwr_Stat_VTC = On@ 9960)
           Then (Air_Loop_Target_Configuration = Nominal1)
        2. If (CFA1_Pwr_Stat_DMC = Off) and
           (CFA2_Pwr_Stat_DMC = On@9200) and
           (IRFA_Pwr_Stat_VTC = On@8784) and
           (ISFA_Pwr_Stat_VTC = On@9960) and
           Then (Air_Loop_Target_Configuration = Nominal2)
        3. If (CFA1_Pwr_Stat_DMC = On@10000) and
           (CFA2_Pwr_Stat_DMC = On@10000) and
           (IRFA_Pwr_Stat_VTC = Off) and
           (ISFA_Pwr_Stat_VTC = Off)
           Then (Air_Loop_Target_Configuration = Isolation)
        4. Else (Air_Loop_Target_Configuration = Non-nominal)

Identify current air loop target configuration

1. Obtain CFA1 mode
2. Obtain CFA2 mode
3. Obtain ISFA mode
4. Obtain IRFA mode
5. Derive air loop target configuration
    1. Recall CFA1 mode
    2. Recall CFA2 mode
    3. Recall IRFA mode
    4. Recall ISFA mode
    5. Parallel evaluation:
        1. If (CFA1_mode = On@9200) and
           (CFA2_mode = Off) and
           (IRFA_mode = On@8784) and
           (ISFA_Pwr_Stat_VTC = On@ 9960)
        Then (Air_Loop_Target_Configuration = Nominal1)
        2. If (CFA1_Pwr_Stat_DMC = Off) and
           (CFA2_Pwr_Stat_DMC = On@9200) and
           (IRFA_Pwr_Stat_VTC = On@8784) and
           (ISFA_Pwr_Stat_VTC = On@9960) and
        Then (Air_Loop_Target_Configuration = Nominal2)
        3. If (CFA1_Pwr_Stat_DMC = On@10000) and
           (CFA2_Pwr_Stat_DMC = On@10000) and
           (IRFA_Pwr_Stat_VTC = Off) and
           (ISFA_Pwr_Stat_VTC = Off)
        Then (Air_Loop_Target_Configuration = Isolation)
        4. Else (Air_Loop_Target_Configuration = Non-nominal)

6. Identify current air loop target configuration

6.1 Goto Airloop Overview

PWS ECLSS
ECLSS Functional Overview

6.2 Verify CFA 1 mode

PWS ECLSS:CFA 1
CFA 1

Verify Pwr – On
Verify Fan Speed > 8800
Verify Fan Speed < 9600

cmd close **Execute**

6.3 Verify CFA 2 mode

PWS ECLSS:CFA 2
CFA 2

Verify Pwr – Off
Verify Fan Speed > 7800
Verify Fan Speed < 8200

cmd close **Execute**

6.4 Verify ISFA mode

PWS ECLSS:ISFA
ISFA

Verify Pwr – On
Verify Fan Speed > 9500
Verify Fan Speed < 10500

cmd close **Execute**

6.5 Verify IRFA mode

PWS ECLSS:IRFA
IRFA

Verify Pwr – On
Verify Fan Speed > 8400
Verify Fan Speed < 9000

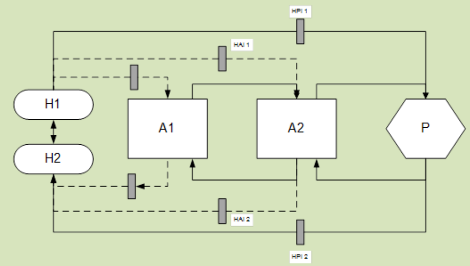
cmd close **Execute**

END OF PROCEDURE

VASCO

AOB

Human-Automation Interaction Situations

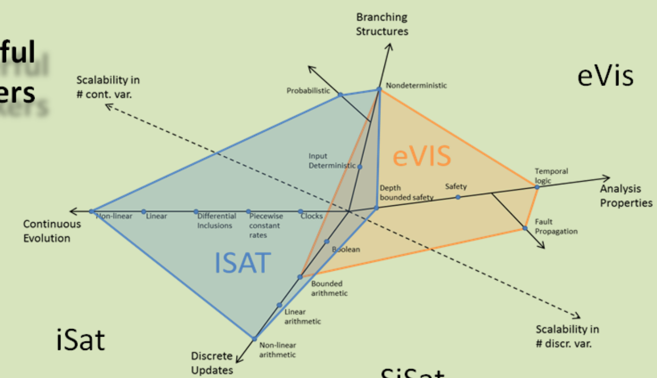


$$P{:}\ M_P = \langle S_P, s_{0_P}, V_P, Val_P, \delta_P \rangle,$$
$$I{:}\ M_I = \langle S_I, s_{0_I}, V_I, Val_I, \delta_I \rangle\ ,$$
$$T{:}\ M_T = \langle S_T, s_{0_T}, V_T, Val_T, \delta_T \rangle,$$

Formal Models and Analysis Questions

$$\exists \phi{:} S_I \rightarrow S_T \forall \sigma \in Tr(M_P) \forall s_T \in S_T, s_{0_T} \overset{\sigma}{\Rightarrow} s_T \wedge s_I \in S_I, s_{0_I} \overset{\sigma}{\Rightarrow} s_I{:}\ s_T = \phi(s_I)$$

Set of Powerful Model Checkers



eVis

iSat-ODE

iSat

SiSat

32 Any other business

- Final Presentation
  - Duration talk/discussion
  - Date for non-public presentation

- Software

VASCO

Case Study System

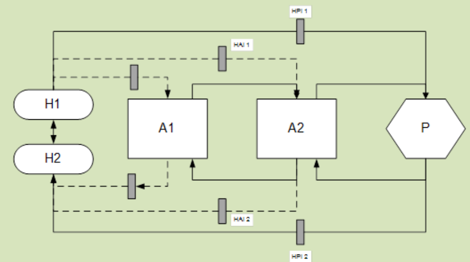
ECLSS

Environmental Control

and

Life Support System

Human-Automation Interaction Situations

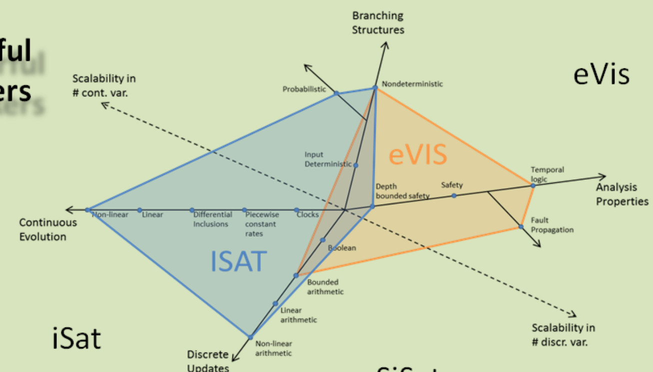


Formal Models and Analysis Questions

$$P: M_P = \langle S_P, s_{0_P}, V_P, Val_P, \delta_P \rangle,$$
$$I: M_I = \langle S_I, s_{0_I}, V_I, Val_I, \delta_I \rangle,$$
$$T: M_T = \langle S_T, s_{0_T}, V_T, Val_T, \delta_T \rangle,$$

$$\exists \phi: S_I \rightarrow S_T \forall \sigma \in Tr(M_P) \forall s_T \in S_T, s_{0_T} \overset{\sigma}{\Rightarrow} s_T \wedge s_I \in S_I, s_{0_I} \overset{\sigma}{\Rightarrow} s_I : s_T = \phi(s_I)$$

Set of Powerful Model Checkers



eVis

iSat-ODE

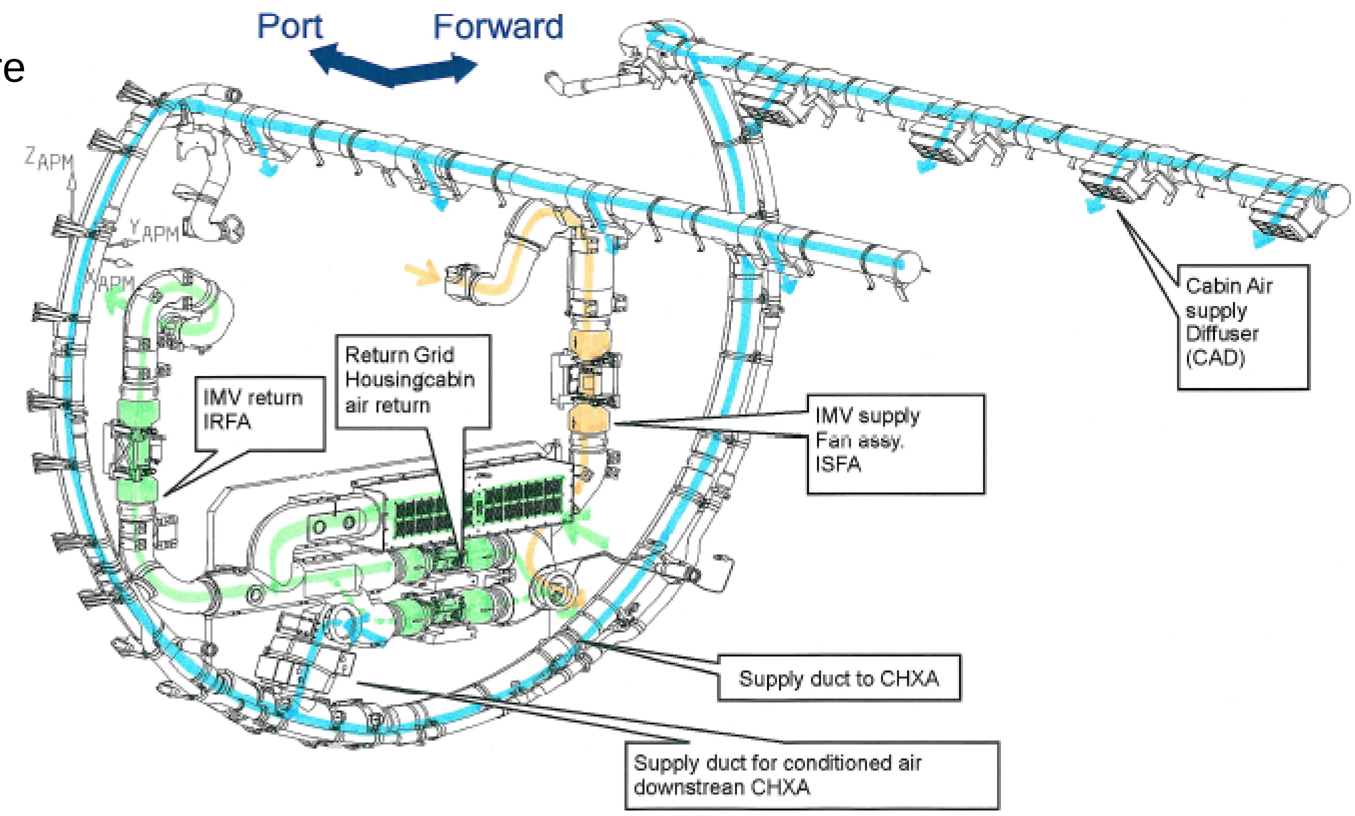
iSat

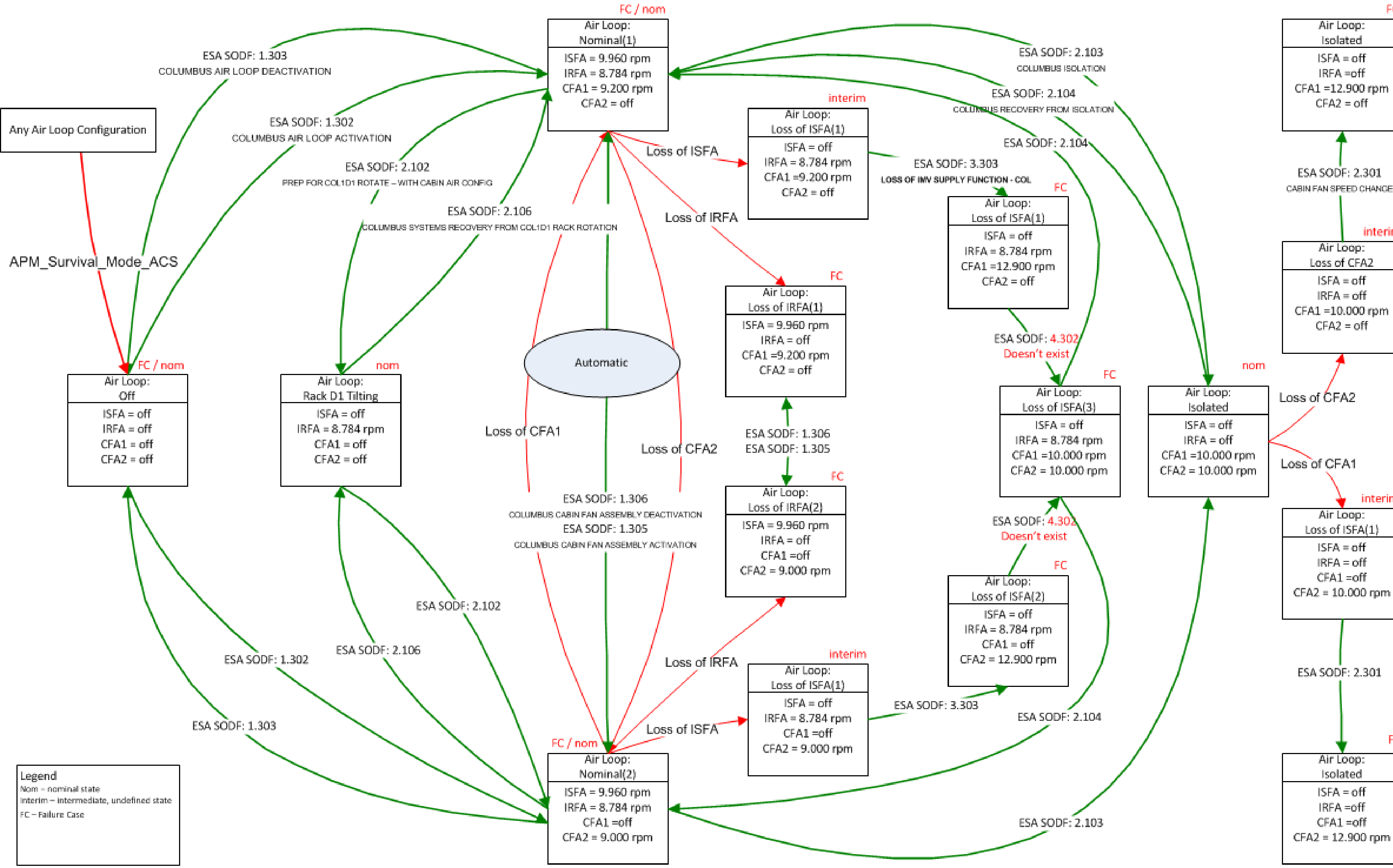
SiSat

34 Case Study System: ECLSS

Main Functions

- Air Condition
- Atmosphere Pressure Control
- Payload Supply
- Fire Detection & Suppression (FDS)




FC / nom
Air Loop:
Nominal(1)
ISFA = 9.960 rpm
IRFA = 8.784 rpm
CFA1 = 9.200 rpm
CFA2 = off
ESA SODF: 1.303
COLUMBUS AIR LOOP DEACTIVATION
Any Air Loop Configuration
ESA SODF: 1.302
COLUMBUS AIR LOOP ACTIVATION
ESA SODF: 2.102
PREP FOR COL1D1 ROTATE – WITH CABIN AIR CONFIG
ESA SODF: 2.106
COLUMBUS SYSTEMS RECOVERY FROM COL1D1 RACK ROTATION
APM_Survival_Mode_ACS
FC / nom
Air Loop:
Off
ISFA = off
IRFA = off
CFA1 = off
CFA2 = off
nom
Air Loop:
Rack D1 Tilting
ISFA = off
IRFA = 8.784 rpm
CFA1 = off
CFA2 = off
Automatic
Loss of ISFA
Loss of IRFA
Loss of CFA1
Loss of CFA2
interim
Air Loop:
Loss of ISFA(1)
ISFA = off
IRFA = 8.784 rpm
CFA1 =9.200 rpm
CFA2 = off
FC
Air Loop:
Loss of IRFA(1)
ISFA = 9.960 rpm
IRFA = off
CFA1 =9.200 rpm
CFA2 = off
ESA SODF: 1.306
ESA SODF: 1.305
FC
Air Loop:
Loss of IRFA(2)
ISFA = 9.960 rpm
IRFA = off
CFA1 =off
CFA2 = 9.000 rpm
ESA SODF: 1.306
COLUMBUS CABIN FAN ASSEMBLY DEACTIVATION
ESA SODF: 1.305
COLUMBUS CABIN FAN ASSEMBLY ACTIVATION
ESA SODF: 2.103
COLUMBUS ISOLATION
ESA SODF: 2.104
COLUMBUS RECOVERY FROM ISOLATION
ESA SODF: 2.104
ESA SODF: 3.303
LOSS OF IMV SUPPLY FUNCTION - COL
FC
Air Loop:
Loss of ISFA(1)
ISFA = off
IRFA = 8.784 rpm
CFA1 =12.900 rpm
CFA2 = off
ESA SODF: 4.302
Doesn't exist
FC
Air Loop:
Loss of ISFA(3)
ISFA = off
IRFA = 8.784 rpm
CFA1 =10.000 rpm
CFA2 = 10.000 rpm
nom
Air Loop:
Isolated
ISFA = off
IRFA = off
CFA1 =10.000 rpm
CFA2 = 10.000 rpm
FC
Air Loop:
Isolated
ISFA = off
IRFA =off
CFA1 =12.900 rpm
CFA2 = off
ESA SODF: 2.301
CABIN FAN SPEED CHANGE
interim
Air Loop:
Loss of CFA2
ISFA = off
IRFA = off
CFA1 =10.000 rpm
CFA2 = off
Loss of CFA2
Loss of CFA1
interim
Air Loop:
Loss of ISFA(1)
ISFA = off
IRFA = off
CFA1 =off
CFA2 = 10.000 rpm
ESA SODF: 2.301
FC
Air Loop:
Isolated
ISFA = off
IRFA =off
CFA1 =off
CFA2 = 12.900 rpm
ESA SODF: 4.302
Doesn't exist
FC
Air Loop:
Loss of ISFA(2)
ISFA = off
IRFA = 8.784 rpm
CFA1 = off
CFA2 = 12.900 rpm
ESA SODF: 3.303
ESA SODF: 2.104
ESA SODF: 2.103
interim
Air Loop:
Loss of ISFA(1)
ISFA = off
IRFA = 8.784 rpm
CFA1 =off
CFA2 = 9.000 rpm
Loss of IRFA
Loss of ISFA
FC / nom
Air Loop:
Nominal(2)
ISFA = 9.960 rpm
IRFA = 8.784 rpm
CFA1 =off
CFA2 = 9.000 rpm
ESA SODF: 2.102
ESA SODF: 2.106
ESA SODF: 1.302
ESA SODF: 1.303
Legend
Nom – nominal state
Interim – intermediate, undefined state
FC – Failure Case

37 ECLSS Emergency, Warnings & Caution Events

Emergencies

- **FIRE Smoke Detector 1 Cabin-COL**
- **FIRE Smoke Detector 2 Cabin-COL**
- **FIRE Smoke Detector ISPR Location-COL**
- (inhibited for F3 and O2)
- FIRE Smoke Detector SUP 1 and 4 - COL
- **FIRE Manual Alarm-COL**
- **RAPID DEPRESS Manual Alarm-COL**
- TOXIC ATMOSPHERE ISPR Location-COL
- **TOXIC ATMOSPHERE Manual Alarm-COL**
- TOXIC ATMOSPHERE SUP 1 and 4 - COL

Warnings

- **Cabin Air Flow Sensor 1 Low**
- **Cabin Air Flow Sensor 2 Low**
- **Total Pressure Sensor 1 Low**
- **Total Pressure Sensor 3 Low**
- ppCO2 Sensor 1 High
- ppCO2 Sensor 2 High

Cautions

- **Loss of CFA Redundancy**
- **Loss of IMV Supply Function**
- **Loss of IMV Return Function**
- Loss of CTCU Redundancy
- Loss of CWSA Redundancy
- **Cabin Air Return Grid Clogging***
- Smoke Detector Failures (Cabin SDs or ISPR SDs, 3 types: Fail, Lens Contamination, Active BIT Fail)
- ppO2 Sensor 1 or 2 Low
- ppO2 Sensor 1 or 2 High
- CDA 1, 2, 3, or 4 Valve 1 Failure
- CDA 1, 2, 3, or 4 Valve 2 Failure
- HCU 1 or 2 Failure
- PPRA 1 or 2 Valve Failure
- VAMRV Failure
- VEMRV Failure

* requires crew response within 1 orbit

Step 1

Identification of relevant issues

- Columbus Flight Note System
  - Collection of all unplanned incidents and anomalies occuring during real time operations
- Prescreened by ECLSS System Engineers
- Grouped in different categories
  - System Failure
  - Hardware Errors
  - Operator Errors
- Assumption was that Operator Errors could lead us to Human Factor and Automation Interaction Issues
- Cabin Air Return Grid Glooging Caution Event is of interest because mandatory crew involvement

39 STEP 2: System Decomposition and Description

Objectives

- The Human-Machine System is decomposed into the following possible components
  - Agents: **Human agents** (operators, users,…) and **Machine agents** (automated systems)
  - **Processes** (upon which the agents act)
  - **Environments** (in which the agents and processes are immersed)
  - **Interfaces**, between all types of components: Human-human interfaces, Human-machine interfaces, Machine-machine interfaces, …

Step 4: System Modeling

Human

- Tasks
- Operations Data File (ODF) Standards
  - 2 representation formats
    - Textual/graphical
    - XML
  - 5 procedure formats
    - Checklist
    - Logic Flow
    - Parallel Activity
    - Joint Vehicle Operations
    - Buss Loss Subsystem
- Error Production Mechanisms
  - Erroneous execution of the procedure at the procedural level
    - Error or omission (omitting a step in a list of similar steps)
    - Step confusion (replacing a step with a rather similar one)

43 Step 4: System Modeling

User Interface

- 3 User Interfaces:

- Satmon:

  Ground Control
- PWS:

  Portable Work Station
- PCS:

  Portable Computer System

- ISS display designs are based on the Displays and Graphics Commonality Standard (DGCS)
- Representation format:

  Unified Synoptic System (USS)


```xml
<?xml version="1.0" ?>
<USSObject>
  <Generator>uss-2.19.0</Generator>
  <FormatVersion>6</FormatVersion>
  <Display>
    <Title>CFA 1</Title>
    <Width>260</Width>
    <Height>520</Height>
    <ExecuteButton>true</ExecuteButton>
    <TargetSystem>PWS</TargetSystem>
    <DatabaseAlias>ECLSS_CFA1_MCD</DatabaseAlias>
    <Description>
      <Format>PLAIN</Format>
      <Text>Product version: OSD 3.1.1_base</Text>
    </Description>
    <ChangeLog>
    <Source>
      <Context>Perforce</Context>
      <Properties>
        <property name="Date" value="$DateTime: 2012/05/16 12:29:37 $" /
        <property name="Id" value="$Id: //esaodf/displays/pws/main/disp
        <property name="CM" value="Perforce" />
        <property name="Revision" value="$Revision: #1 $" />
        <property name="Change" value="$Change: 82436 $" />
      </Properties>
    </Source>
    <Elements>
      <Label>
        <Text>ECLSS Cabin Fan Assembly 1</Text>
        <TextStyle>
          <Fontname>Lucida Sans</Fontname>
```

Step 6: Verification

Process: Air Flow

- For each air loop configuration contained in the data a HySat clause is created

Configuration ➡

Process envelopes ➡

```
(((CFA1_Pwr_Stat_DMC == 1) and (CFA1_FanSpeed == 9200) and
(CFA2_Pwr_Stat_DMC == 1) and (CFA2_FanSpeed == 9900) and
(IRFA_Pwr_Stat_DMC == 1) and (IRFA_FanSpeed == 8784) and
(ISFA_Pwr_Stat_DMC == 0)))
->
(((CFA1_Delta_P_DMC >= 0.45707834) and (CFA1_Delta_P_DMC <= 0.5126963) and
(CFA1_Fan_Speed_DMC >= 9142.745) and (CFA1_Fan_Speed_DMC <= 9236.52) and
(CFA1_Input_Current_DMC >= 0.34733704) and (CFA1_Input_Current_DMC <= 0.38104492) and
(CFA2_Delta_P_DMC >= 0.72202224) and (CFA2_Delta_P_DMC <= 0.7928088) and
(CFA2_Fan_Speed_DMC >= 9810.889) and (CFA2_Fan_Speed_DMC <= 9916.385) and
(CFA2_Input_Current_DMC >= 1.0346845) and (CFA2_Input_Current_DMC <= 1.0933069) and
(IRFA_Delta_P_MVD >= 0.43179744) and (IRFA_Delta_P_MVD <= 0.5319098) and
(IRFA_Fan_Speed_VTC >= 8740.297) and (IRFA_Fan_Speed_VTC <= 8826.257) and
(IRFA_Input_Current_DMC >= 0.7708838) and (IRFA_Input_Current_DMC <= 0.8426962) and
(ISFA_Delta_P_MVD >= 0.0070782523) and (ISFA_Delta_P_MVD <= 0.008089488) and
(ISFA_Fan_Speed_VTC >= 8005.7305) and (ISFA_Fan_Speed_VTC <= 8005.7305) and
(ISFA_Input_Current_DMC >= -5.9660937E-7) and (ISFA_Input_Current_DMC <= 0.0014649631))
or
(…))
;
```

45 Step 6: Verification

Process: Air Flow

- For each air loop configuration contained in the data a HySat clause is created
- Configurations not contained in the data
  - no clause
  - process behaviour is not restricted
- No explicit dynamic behaviour
  - Model Checker can arbitrary choose within envelopes in each step
- No step semantic
  - new envelopes get immediately active,

    if the target configuration changes

Configuration →

```
(((CFA1_Pwr_Stat_DMC == 1) and (
(CFA2_Pwr_Stat_DMC == 1) and (CF
(IRFA_Pwr_Stat_DMC == 1) and (IR
(ISFA_Pwr_Stat_DMC == 0)))
->
```

Process envelopes →

```
(((CFA1_Delta_P_DMC >= 0.4570783
(CFA1_Fan_Speed_DMC >= 9142.745)
(CFA1_Input_Current_DMC >= 0.347
(CFA2_Delta_P_DMC >= 0.72202224)
(CFA2_Fan_Speed_DMC >= 9810.889)
(CFA2_Input_Current_DMC >= 1.034
(IRFA_Delta_P_MVD >= 0.43179744)
(IRFA_Fan_Speed_VTC >= 8740.297)
(IRFA_Input_Current_DMC >= 0.770
(ISFA_Delta_P_MVD >= 0.007078252
(ISFA_Fan_Speed_VTC >= 8005.7305
(ISFA_Input_Current_DMC >= -5.96
or
(…))
;
```

47 Step 5: Verification Technology

HySat / iSAT

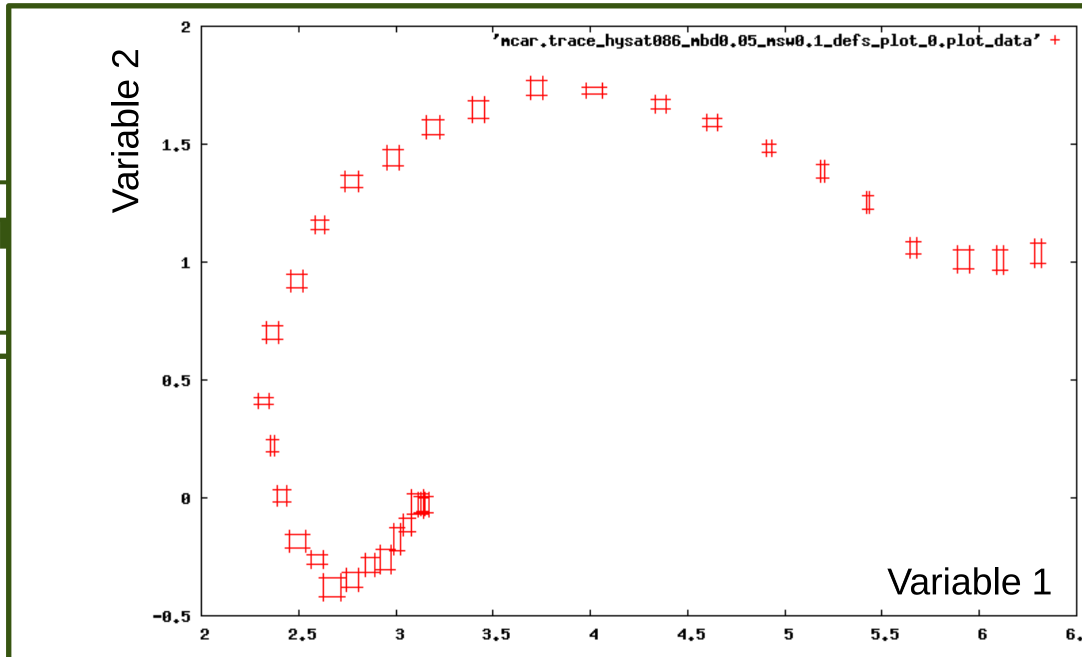
- Bounded model checker
  - Probabilism
  - Non-determinism
  - Non-linear arithmetic
- No direct support for temporal logic
  - Depth bounded
- Simple input format
  - Declarations
  - Initializations
  - Transition function
  - Target property
- Property
  - ensured for defined depths of analysis (k)
  - or counterexample provided

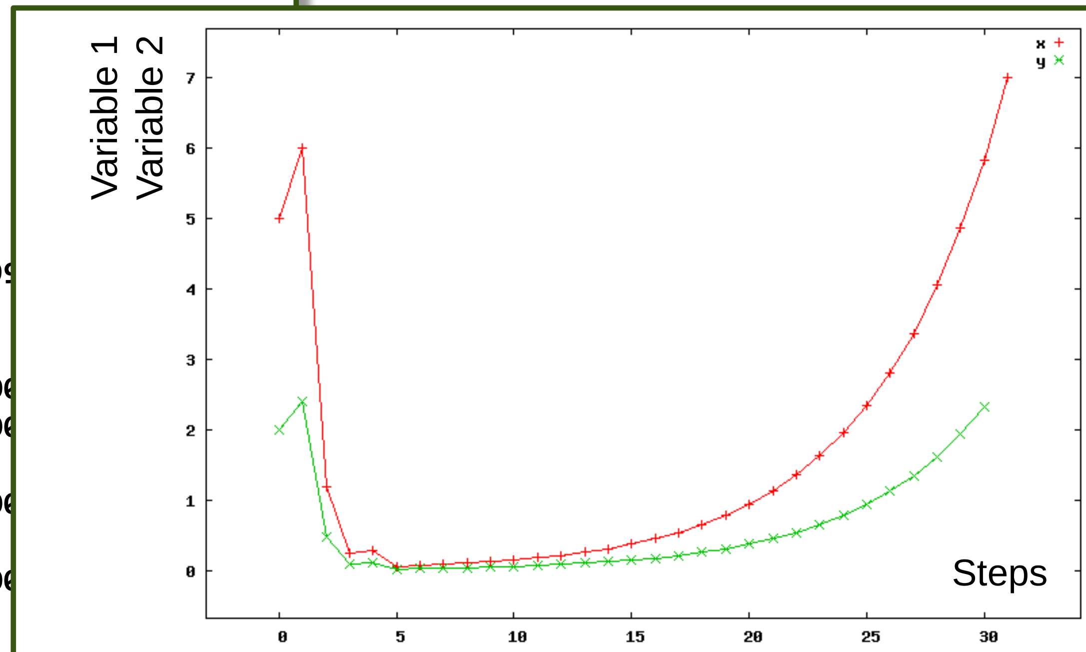
```
DECL
  define f = 2.0;
  float [0, 1000] x;
  boole jump ;

INIT
  x = 0.6;
  ! jump ;

TRANS
  jump ’ <-> ! jump ;

  jump -> f * x’ = x;
  ! jump -> x’ = x + 2;

TARGET
  x > 3.5;
```

48 Step 5: Verification Techn

Traces

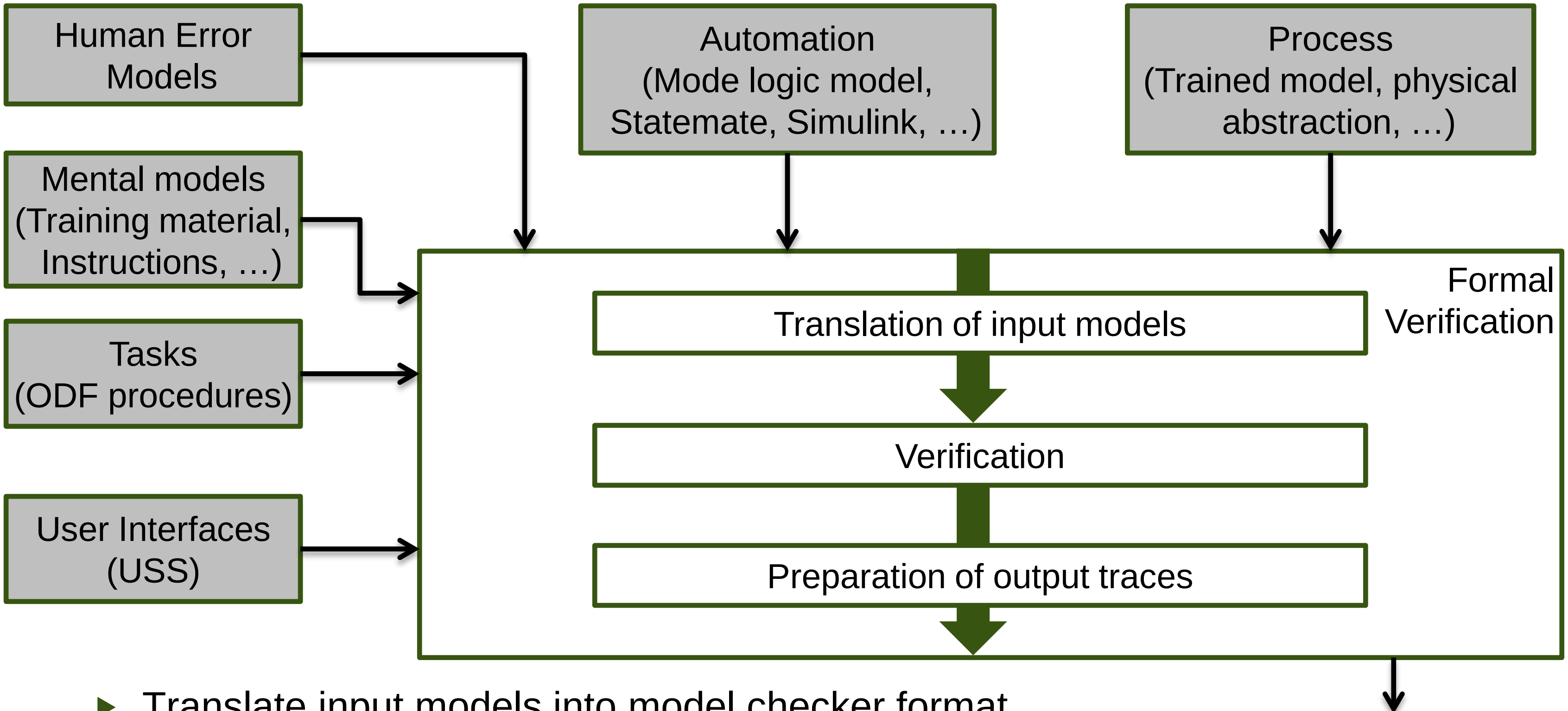
```
SOLUTION:
  CFA2_Fan_Speed_DMC (int):
    @0: [0, 0]
    @1: [0, 0]
    @2: [0, 0]
    @3: [0, 0]
    @4: [0, 0]
    @5: [0, 0]
    @6: [0, 0]
    @7: [0, 0]
    @8: [9000, 9000]
    @9: [10000, 10000]
    @10: [10000, 10000]
  CFA1_Delta_P_DMC (float):
    @0: [0.75, 0.77000000000000001776]
    @1: [0.71999999999999997335, 0.72999999
    @2: [0.75, 0.79000000000000003553]
    @3: [0.75, 0.77000000000000001776]
    @4: [0.76000000000000000888, 0.77000000
    @5: [0.76000000000000000888, 0.77000000
    @6: [0.75, 0.77624999999999988454]
    @7: [0.47999999999999998224, 0.55000000
    @8: [0.75, 0.77000000000000001776]
    @9: [0.45000000000000000111, 0.52000000
...
```





Step 6: Verification

General Idea



- Translate input models into model checker format
- Translate analysis question into model checker format
- Run verification process
- Prepare traces

- ✓ C1.3 Information on automation states
- ✗ C1.4 Consistent effects
- ✓ C2.6 Feedback
- ✓ C3.3 Deterministic automation

Step 6: Verification (Translation of input models)

Process: Air Flow

- Approach
  - Columbus telemetry data
  - Data period: 01.01.2009 – 31.12.2010
  - For each of the four fans:
    - Delta Pressure (Delta_P)
    - Measured Fan Speed (Fan_Speed)
    - Power Status (Pwr_Stat)
    - Input Current (Input_Current)
  - > 10^9 data samples

  - Splitting data sets, at times where target configuration changes (Pwr_Stat, Fan_Speed)

51 Step 6: Verification (Translation of input models)

Process: Air Flow

- For each air loop configuration contained in the data a HySat clause is created

Configuration

Process envelopes

```
(((CFA1_Pwr_Stat_DMC == 1) and (CFA1_FanSpeed == 9200) and
(CFA2_Pwr_Stat_DMC == 1) and (CFA2_FanSpeed == 9900) and
(IRFA_Pwr_Stat_DMC == 1) and (IRFA_FanSpeed == 8784) and
(ISFA_Pwr_Stat_DMC == 0)))
->
(((CFA1_Delta_P_DMC >= 0.45707834) and (CFA1_Delta_P_DMC <= 0.5126963) and
(CFA1_Fan_Speed_DMC >= 9142.745) and (CFA1_Fan_Speed_DMC <= 9236.52) and
(CFA1_Input_Current_DMC >= 0.34733704) and (CFA1_Input_Current_DMC <= 0.38104492) and
(CFA2_Delta_P_DMC >= 0.72202224) and (CFA2_Delta_P_DMC <= 0.7928088) and
(CFA2_Fan_Speed_DMC >= 9810.889) and (CFA2_Fan_Speed_DMC <= 9916.385) and
(CFA2_Input_Current_DMC >= 1.0346845) and (CFA2_Input_Current_DMC <= 1.0933069) and
(IRFA_Delta_P_MVD >= 0.43179744) and (IRFA_Delta_P_MVD <= 0.5319098) and
(IRFA_Fan_Speed_VTC >= 8740.297) and (IRFA_Fan_Speed_VTC <= 8826.257) and
(IRFA_Input_Current_DMC >= 0.7708838) and (IRFA_Input_Current_DMC <= 0.8426962) and
(ISFA_Delta_P_MVD >= 0.0070782523) and (ISFA_Delta_P_MVD <= 0.008089488) and
(ISFA_Fan_Speed_VTC >= 8005.7305) and (ISFA_Fan_Speed_VTC <= 8005.7305) and
(ISFA_Input_Current_DMC >= -5.9660937E-7) and (ISFA_Input_Current_DMC <= 0.0014649631))
or
(…))
;
```

52 Step 6: Verification (Translation of input models)

Automation

- Warning system
  - Every warning definition is translated into HySat clauses:

```
RETURN_GRID_CLOGGING'
<->
(((MONIT_ENABL_CFA1_Delta_P_DMC = 1) and (CFA1_Delta_P_DMC <= 0.61)) or
((MONIT_ENABL_CFA1_Input_Current_DMC = 1) and (CFA1_Input_Current_DMC <= 0.48)) or
((MONIT_ENABL_CFA2_Delta_P_DMC = 1) and (CFA2_Delta_P_DMC <= 0.65)) or
((MONIT_ENABL_CFA2_Input_Current_DMC = 1) and (CFA2_Input_Current_DMC <= 0.53)));
```

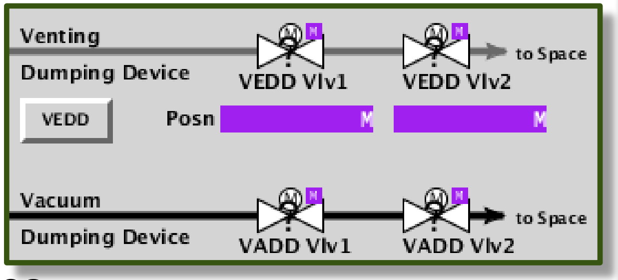
- Flight Automated Procedures (FLAPs) were created manually

53 Step 6: Verification (Translation of input models)

User Interface

- Focus on onboard PWS and PCS interfaces created in USS format.
- Considered elements
  - ECLSS relevant
  - Dynamic
  - Static labeling elements

- Dynamic elements describe which information and commands are available
- Labeling elements describe how the dynamic elements are addressed in the procedures



1. CYCLING VENTING DUMPING DEVICE VALVES

ECLSS: Payload Vacuum & Venting

Payload Vacuum & Venting

'Venting Dumping Device'

Verify VEDD Vlv1 – Closed
Verify VEDD Vlv2 – Closed

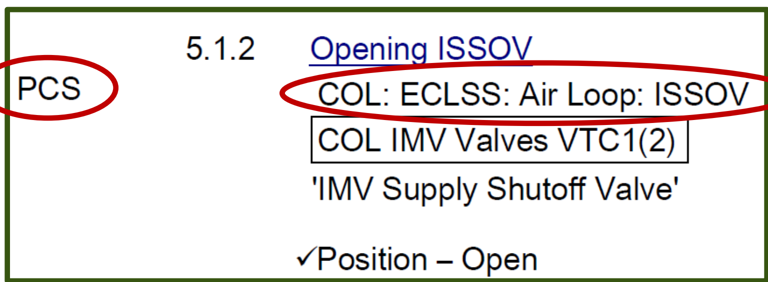
Element	Type	ECLSS relevant
Arc	static	yes
BarGraph		no
CAGShape		no
CheckValve		no
ComboBox		no
CommandList	dynamic	yes
CommandButton	dynamic	yes
Compound	static	yes
Ellipse	static	yes
EllipticTickMeter		no
ExternalImage	static	yes
Field	Dynamic	yes
FileChooser		no
InputField		no
Label	static	yes
LineGraph		no
LinearTickMeter		no
NavigationButton	dynamic	yes
Pipe		no
PlaceHolder		no
Polygon	static	yes
Polyline	static	yes
Rectangle	static	yes
StripGraph		no
Symbol	static	yes
TankMeter		no
Thermometer		no
Valve	dynamic	yes

54 Step 6: Verification (Translation of input models)

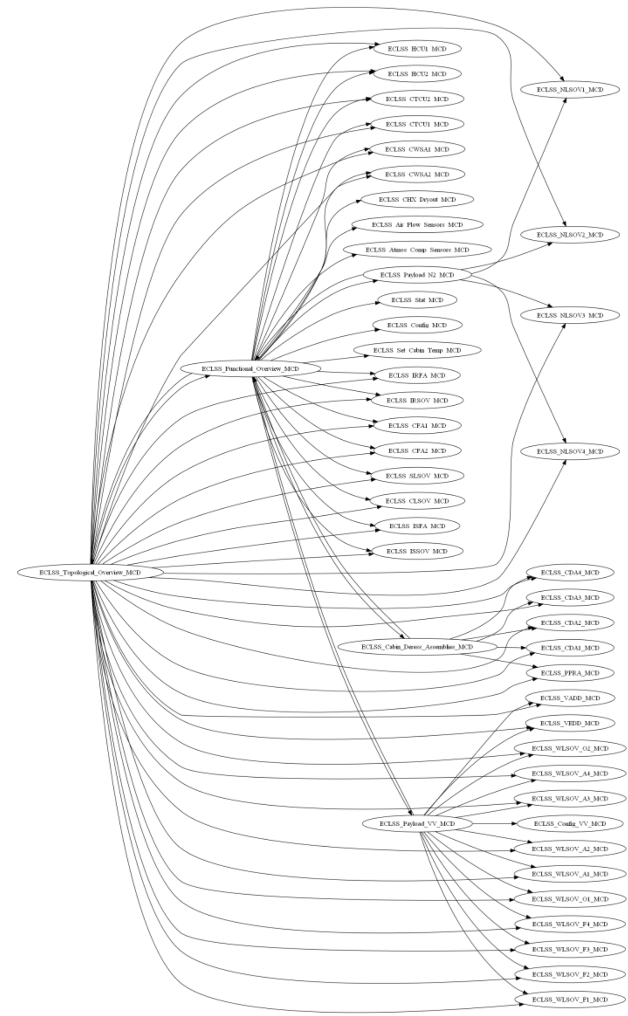
User Interface

- Navigation Button
  - Opens new window or closes current one
- Navigation graph is derived from Navigation Buttons
- Used to create the action sequence for moving from one display to another
- Procedures also refer to these navigation paths

Display System



Navigation Path



- Missing formalisms
  - Explicit links between labels and labeled components
- Combined translation of procedures and displays reveals inconsistencies or structures that do not comply to standards

Step 6: Verification (Translation of input models)

User Interface

- Fields
  - Displaying numerical values or modes of equipment
  - Background color is set by the monitoring system
    - Depends on monitoring state and the nominal/danger limits defined for the displayed sensor value
    - Each display system has its own colour coding
    - HySat clauses are automatically added to reflect the monitoring behaviour, based on the monitoring state variables of the automation (automated monitoring) e.g.:

```
(MonStat_ISFA_Input_Current_DMC’ = MON_STATE_DANGER_HIGH_LIMIT_VIOLATION) or
(MonStat_ISFA_Input_Current_DMC’ = MON_STATE_DANGER_LOW_LIMIT_VIOLATION)) ->
(PWS_ISFA_PrimCurrent_bg_color’ = BG_COLOR_PWS_ORANGE)) and ...
```

- Command Button
  - Activates FLAP
  - Manual translation of FLAPS

Step 6: Verification (Translation of input models)

Procedure

- Translation of a single subtask
  - Step variable to describe current state of the task

7. DEACTIVATING ACTIVE CWSA

ECLSS: CWSA 1(2)

CWSA 1(2)

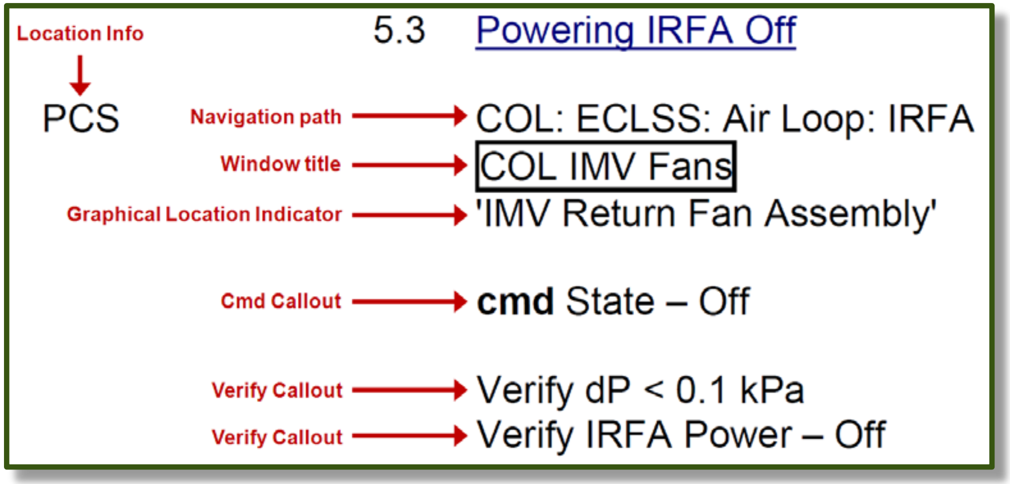
Verify Pwr – Off
Verify Motor Speed – 3300 to 3700 rpm

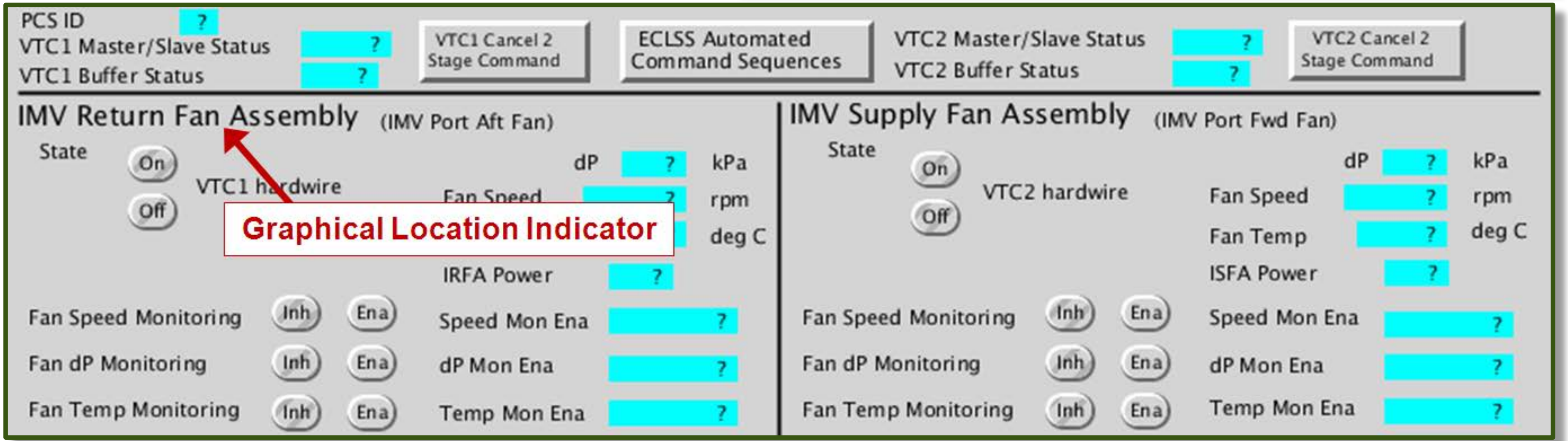
```
-- Step: 7;
 -- DEACTIVATING ACTIVE CWSA;
 (p2_102_state = P2_102_STATE_i1503)
   -> ((p2_102_state' = P2_102_STATE_i1503_locinfo) and (active_uss' = ACTIVE_USS_PWS));
 (p2_102_state = P2_102_STATE_i1503_locinfo)
   -> ((p2_102_state' = P2_102_STATE_i1504) and (PWS_active_display' = PWS_ACTIVE_DISPLAY_ECLSS_Config_MCD));
 (p2_102_state = P2_102_STATE_i1504)
   -> ((p2_102_state' = P2_102_STATE_i1506) and (PWS_active_display' = PWS_ACTIVE_DISPLAY_ECLSS_CWSA1_MCD));

 ((p2_102_state = P2_102_STATE_i1506) and !(CWSA1_Pwr_Stat_DMC = Off))
   -> ((p2_102_state' = P2_102_STATE_i1507));
 ((p2_102_state = P2_102_STATE_i1506) and  (CWSA1_Pwr_Stat_DMC = Off))
   -> (error_flag_verify_failed' = 1);

 ((p2_102_state = P2_102_STATE_i1507) and !((CWSA1_Motor_Speed_DMC >= 3300) and (CWSA1_Motor_Speed_DMC <= 3700)))
   -> ((p2_102_state' = P2_102_STATE_i2516));
 ((p2_102_state = P2_102_STATE_i1507) and ((CWSA1_Motor_Speed_DMC >= 3300) and (CWSA1_Motor_Speed_DMC <= 3700)))
   -> (error_flag_verify_failed' = 1);
```

57 Step 6: Verification (Translation of input models)

Procedure

- References the used User Interface
- All labels and descriptions reference elements of the user interface



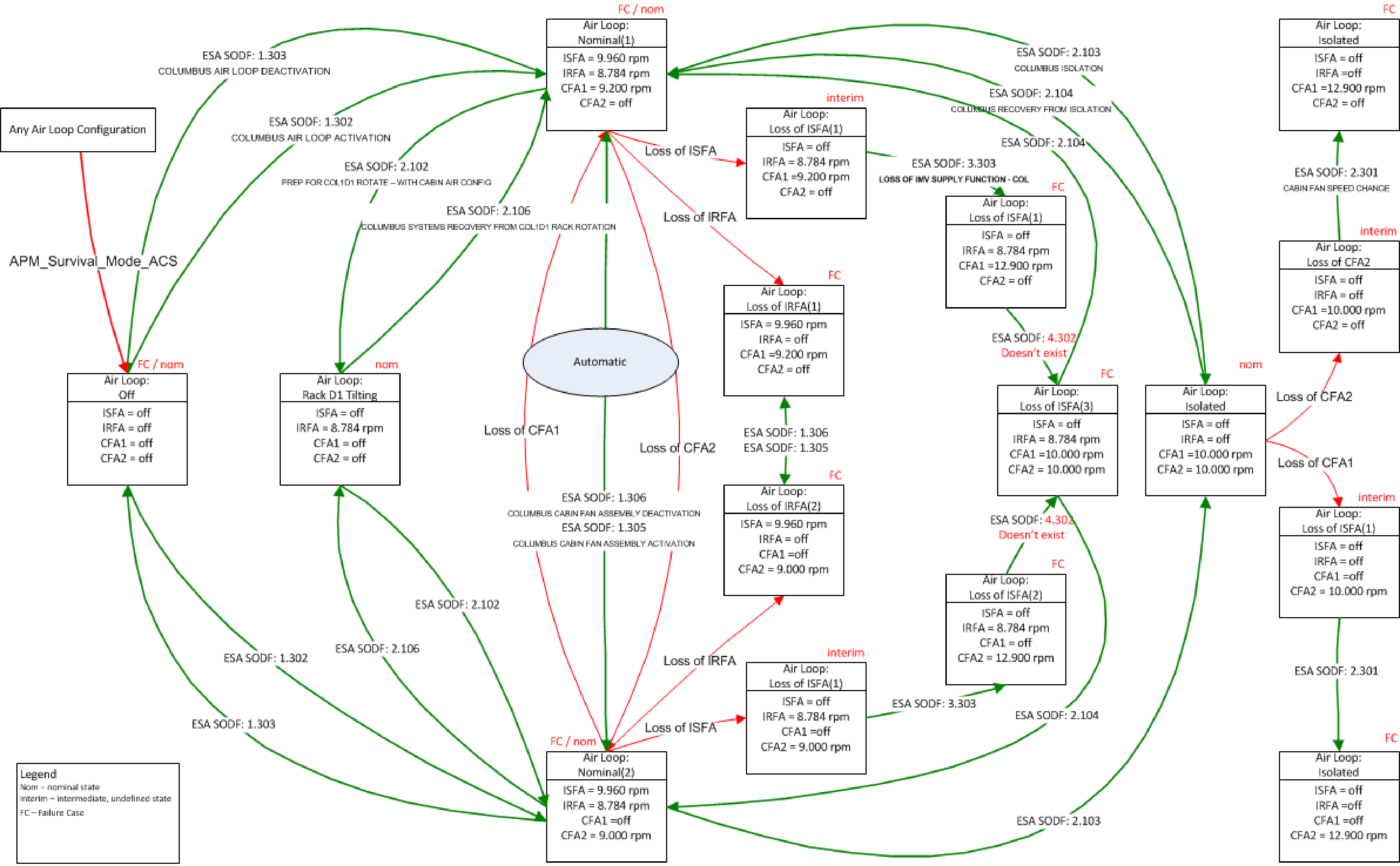


58 Step 4: System Modeling

Automation: Air Flow Control

- Implementation is to complex for being used in the VASCO case study
- High Level description:
  - Mode definitions and transitions
    - Fan modes (On/Off) and speeds for IRFA, ISFA, CFA1 and CFA2

VASCO



Any Air Loop Configuration
APM_Survival_Mode_ACS
FC / nom
Air Loop: Nominal(1)
ISFA = 9.960 rpm
IRFA = 8.784 rpm
CFA1 = 9.200 rpm
CFA2 = off
ESA SODF: 1.303
COLUMBUS AIR LOOP DEACTIVATION
ESA SODF: 1.302
COLUMBUS AIR LOOP ACTIVATION
ESA SODF: 2.102
PREP FOR COL1D1 ROTATE – WITH CABIN AIR CONFIG
ESA SODF: 2.106
COLUMBUS SYSTEMS RECOVERY FROM COL1D1 RACK ROTATION
interim
Air Loop: Loss of ISFA(1)
ISFA = off
IRFA = 8.784 rpm
CFA1 =9.200 rpm
CFA2 = off
Loss of ISFA
Loss of IRFA
ESA SODF: 3.303
LOSS OF IMV SUPPLY FUNCTION - COL
FC
Air Loop: Loss of ISFA(1)
ISFA = off
IRFA = 8.784 rpm
CFA1 =12.900 rpm
CFA2 = off
ESA SODF: 2.103
COLUMBUS ISOLATION
ESA SODF: 2.104
COLUMBUS RECOVERY FROM ISOLATION
ESA SODF: 2.104
FC
Air Loop: Isolated
ISFA = off
IRFA =off
CFA1 =12.900 rpm
CFA2 = off
ESA SODF: 2.301
CABIN FAN SPEED CHANGE
interim
Air Loop: Loss of CFA2
ISFA = off
IRFA = off
CFA1 =10.000 rpm
CFA2 = off
FC / nom
Air Loop: Off
ISFA = off
IRFA = off
CFA1 = off
CFA2 = off
nom
Air Loop: Rack D1 Tilting
ISFA = off
IRFA = 8.784 rpm
CFA1 = off
CFA2 = off
Automatic
Loss of CFA1
Loss of CFA2
FC
Air Loop: Loss of IRFA(1)
ISFA = 9.960 rpm
IRFA = off
CFA1 =9.200 rpm
CFA2 = off
ESA SODF: 1.306
ESA SODF: 1.305
FC
Air Loop: Loss of IRFA(2)
ISFA = 9.960 rpm
IRFA = off
CFA1 =off
CFA2 = 9.000 rpm
ESA SODF: 4.302
Doesn't exist
FC
Air Loop: Loss of ISFA(3)
ISFA = off
IRFA = 8.784 rpm
CFA1 =10.000 rpm
CFA2 = 10.000 rpm
nom
Air Loop: Isolated
ISFA = off
IRFA = off
CFA1 =10.000 rpm
CFA2 = 10.000 rpm
Loss of CFA2
Loss of CFA1
interim
Air Loop: Loss of ISFA(1)
ISFA = off
IRFA = off
CFA1 =off
CFA2 = 10.000 rpm
ESA SODF: 1.306
COLUMBUS CABIN FAN ASSEMBLY DEACTIVATION
ESA SODF: 1.305
COLUMBUS CABIN FAN ASSEMBLY ACTIVATION
ESA SODF: 4.302
Doesn't exist
FC
Air Loop: Loss of ISFA(2)
ISFA = off
IRFA = 8.784 rpm
CFA1 = off
CFA2 = 12.900 rpm
ESA SODF: 2.102
ESA SODF: 2.106
ESA SODF: 1.302
ESA SODF: 1.303
Loss of IRFA
interim
Air Loop: Loss of ISFA(1)
ISFA = off
IRFA = 8.784 rpm
CFA1 =off
CFA2 = 9.000 rpm
Loss of ISFA
ESA SODF: 3.303
ESA SODF: 2.104
ESA SODF: 2.301
FC / nom
Air Loop: Nominal(2)
ISFA = 9.960 rpm
IRFA = 8.784 rpm
CFA1 =off
CFA2 = 9.000 rpm
ESA SODF: 2.103
FC
Air Loop: Isolated
ISFA = off
IRFA =off
CFA1 =off
CFA2 = 12.900 rpm
Legend
Nom – nominal state
Interim – intermediate, undefined state
FC – Failure Case

Step 6: Verification

Procedure Validation and Robustness Analysis

- Initializing the system in nominal configuration:

```
CFA1_Pwr_Stat_DMC = 1;
CFA1_SetSpeed_DMC = 9200;
CFA2_Pwr_Stat_DMC = 0;
CFA2_SetSpeed_DMC = 8900;
IRFA_Pwr_Stat_DMC = 1;
IRFA_SetSpeed_VTC= 8784;
ISFA_Pwr_Stat_DMC= 1;
ISFA_SetSpeed_VTC= 9960;
MonEnab_CFA2_Input_Current_DMC = 0
MonEnab_CFA2_Fan_Speed_DMC = 0
MonEnab_CFA2_Delta_P_DMC = 0
```

Fan Assemblies are in Nominal Configuration 1:

CFA1 @ 9200	ISFA @ 9960
CFA2 - Off	IRFA @ 8784

CFA2 is Off.
Therefore Monitoring for its sensor values is turned off

- Analysis question:

```
LOSS_IRFA or LOSS_ISFA or LOSS_CFA1 or LOSS_CFA2 or RETURN_GRID_CLOGGING
or error_flag_verify_failed or error_flag_check_failed
or ((step > (P2_102_STATE_i2537 * 2))
     and !(p2_102_state = P2_102_STATE_FINISHED))
```

Is it possible, that
→ a warning occurs
→ a verify instruction is violated
→ the procedure is not finished within the expected maximum number of steps

Step 6: Verification

Robustness analysis

- Analysis question: Is it possible, that the system gets into a critical state, if the operator makes one (or two, three, …, n) plausible errors?

- Inject human errors into nominal procedures
  - Error Production Mechanisms (EPMs)

- Initialize the HAI system
  - in nominal configuration
  - enable model checker to activate up to n injected human errors

```
omission_error_1 + omission_error_2 +...+ omission_error_m <= n
```

Step 6: Verification

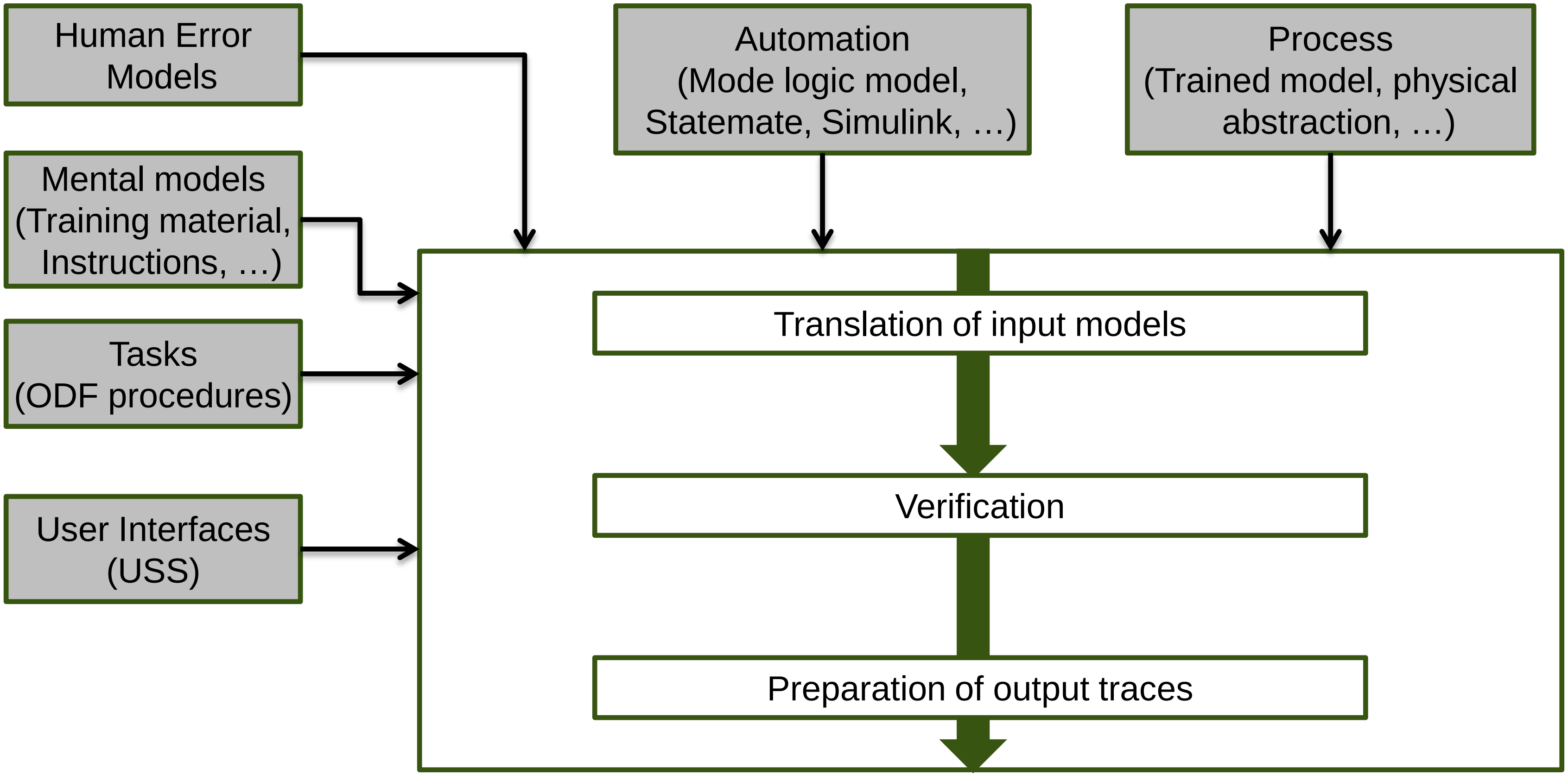
Error Production Mechanism (EPM)

- Error of omission:

```
((p2_102_state = P2_102_STATE_i1553) and (omission_error_2 = 1))
  -> (p2_102_state' = P2_102_STATE_i1554);

((p2_102_state = P2_102_STATE_i1553) and (omission_error_2 = 0))
  -> ((p2_102_state' = P2_102_STATE_i1553_FLAP_EXE) and (EMDI02ES4178K' = 1));

((p2_102_state = P2_102_STATE_i1553_FLAP_EXE) and (EMDI02ES4178K = 0))
  -> (p2_102_state' = P2_102_STATE_i1554);

((p2_102_state = P2_102_STATE_i1553_FLAP_EXE) and (EMDI02ES4178K = 1))
  -> (p2_102_state' = p2_102_state);
```

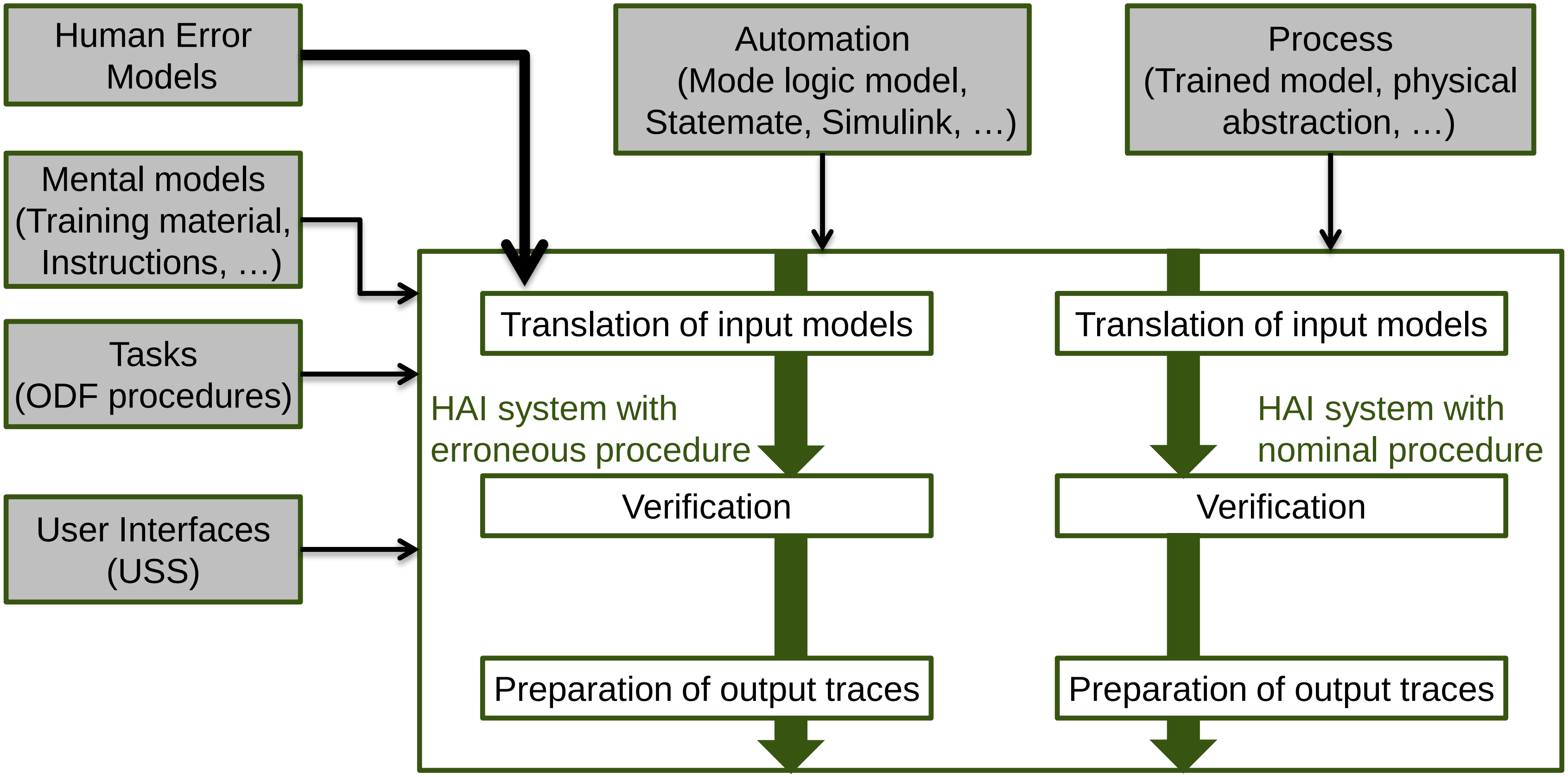
Step 6: Verification

Procedure Validation vs. Robustness Analysis



Step 6: Verification

Procedure Validation vs. Robustness Analysis



Step 6: Verification

Analysis Questions – Core question 1

- Q1) Does the UI present all the information needed by the human agent?

$$G\ (p \rightarrow q)$$

p: addressed situation

q: required information presented

  - Presentation within time frame

$$G\ (p \rightarrow (r\ U\ q))$$

Every time a certain situation (p) is encountered, the required information (q), will be received by the human operator before the maximum allowed amount of time (r) has passed

Step 6: Verification

Analysis Questions – C1.3

- C1.3: is the information on automation state sufficient to interact efficiently with automation?

$p_{\text{mode automation}}$:	(ISFA_mode = W) and (IRFA_mode = X) and (CFA1_mode = Y) and (CFA2_mode = Z)
$q_{\text{mode mental}}$:	(ISFA_mental_mode = W) and (IRFA_mental_mode = X) and (CFA1_mental_mode = Y) and (CFA2_mental_mode = Z)
r:	(steps_taken < n)

- It doesn't take the operator more than n steps to identify the current air loop mode

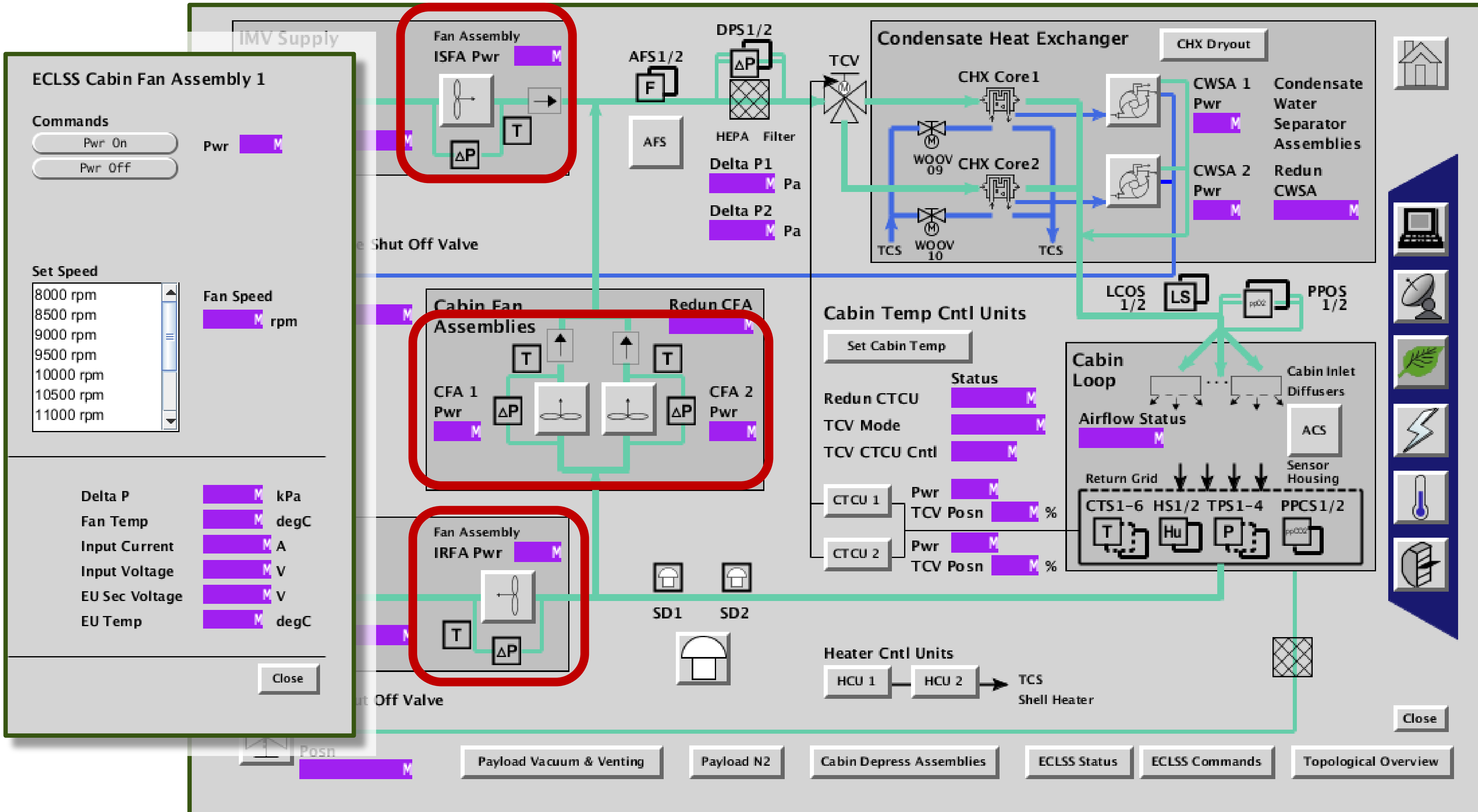
ASCO

67 Step 6: Verification

Analysis Questions – C1.3 - Satmon

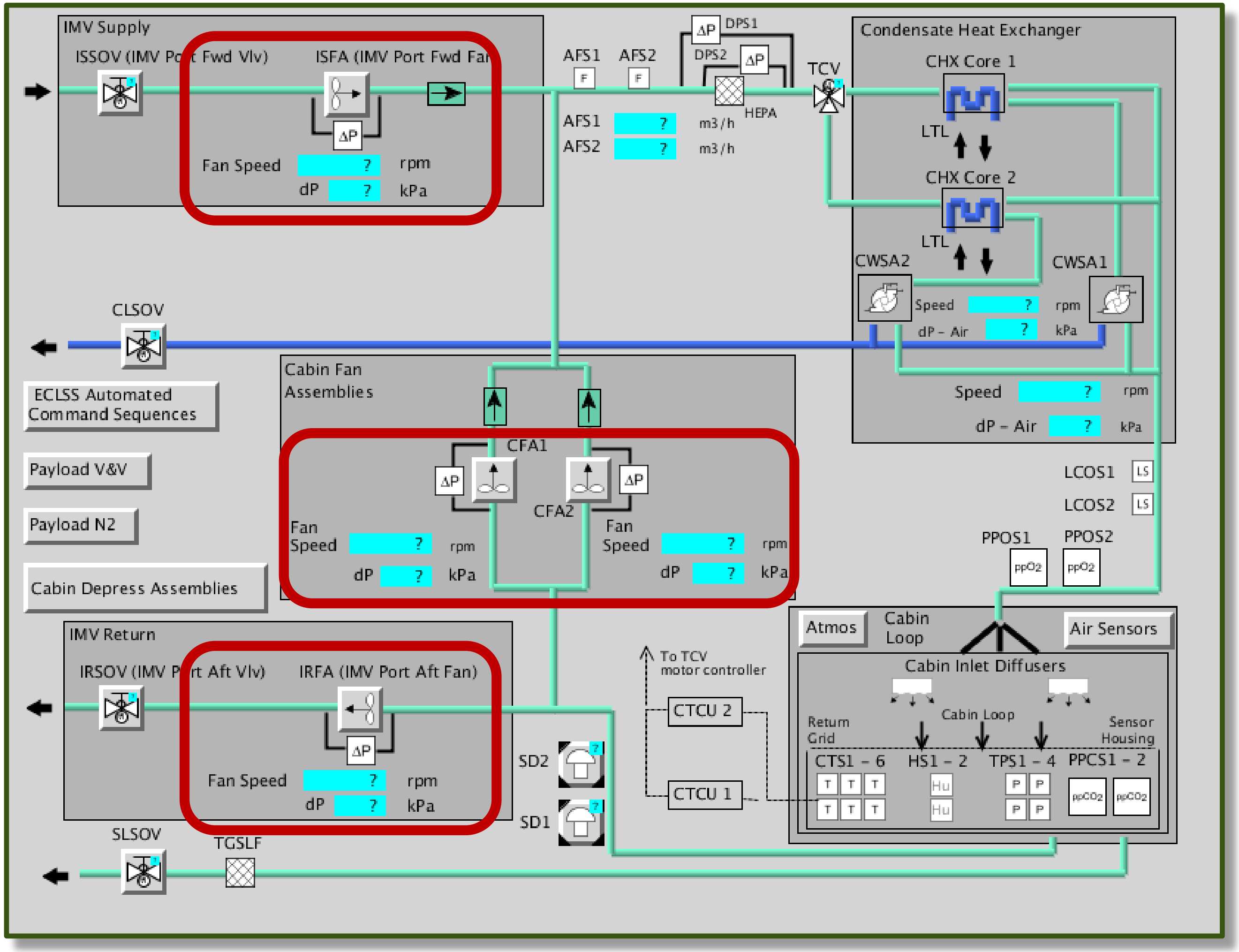
Step 6: Verification

Analysis Questions – C1.3 - PWS



Step 6: Verification

Analysis Questions – C1.3 - PCS



70 Step 6: Verification

Analysis Questions – C1.4 – Does a given action cause consistent effects?

- Parallel composition of two identical systems
- Synchronous procedure execution.
- Is it possible to observe different effects?

```
((A_p2_102_state = P2_102_STATE_i1325) and (B_p2_102_state = P2_102_STATE_i1325))
  -> ((A_LOSS_CFA1 = B_LOSS_CFA1) and (A_LOSS_CFA2 = B_LOSS_CFA2) and
      (A_LOSS_ISFA = B_LOSS_ISFA) and (A_LOSS_IRFA = B_LOSS_IRFA) and
      (A_RETURN_GRID_CLOGGING = B_RETURN_GRID_CLOGGING));
```

- Result: Warning occurred in one system copy and not in the other dependent on the initial state of the system
- Problem: Meaningful definition of effects

CFA1@9200, CFA-off, IRFA@8784, ISFA@9960

```
CFA1_Pwr_Stat_DMC = 1;
CFA1_SetSpeed_DMC = 9200;
CFA2_Pwr_Stat_DMC = 0;
CFA2_SetSpeed_DMC = 8900;
IRFA_Pwr_Stat_DMC = 1;
IRFA_SetSpeed_VTC= 8784;
ISFA_Pwr_Stat_DMC= 1;
ISFA_SetSpeed_VTC= 9960;
MonEnab_CFA2_Input_Current_DMC = 0;
MonEnab_CFA2_Fan_Speed_DMC = 0;
MonEnab_CFA2_Delta_P_DMC = 0;
```

CFA1@9200, CFA-off, IRFA-off, ISFA@9960

```
CFA1_Pwr_Stat_DMC = 1;
CFA1_SetSpeed_DMC = 9200;
CFA2_Pwr_Stat_DMC = 0;
CFA2_SetSpeed_DMC = 8900;
IRFA_Pwr_Stat_DMC = 0;
IRFA_SetSpeed_VTC= 8784;
ISFA_Pwr_Stat_DMC= 1;
ISFA_SetSpeed_VTC= 9960;
MonEnab_CFA2_Input_Current_DMC = 0;
MonEnab_CFA2_Fan_Speed_DMC = 0;
MonEnab_CFA2_Delta_P_DMC = 0;
MonEnab_IRFA_Input_Current_DMC = 0;
MonEnab_IRFA_Delta_P_VTC = 0;
MonEnab_IRFA_Fan_Speed_VTC = 0;
```

71 Step 6: Verification

Analysis Questions – C3.3

- Same approach for C3.3:

  Can the automation, as presented on the UI, be considered as a deterministic state machine for the operator?

```
((A_p2_102_state = B_p2_102_state))
  -> ((A_LOSS_CFA1 = B_LOSS_CFA1) and (A_LOSS_CFA2 = B_LOSS_CFA2) and
      (A_LOSS_ISFA = B_LOSS_ISFA) and (A_LOSS_IRFA = B_LOSS_IRFA) and
      (A_RETURN_GRID_CLOGGING = B_RETURN_GRID_CLOGGING));
```

- Same result

CFA1@9200, CFA-off, IRFA@8784, ISFA@9960

```
CFA1_Pwr_Stat_DMC = 1;
CFA1_SetSpeed_DMC = 9200;
CFA2_Pwr_Stat_DMC = 0;
CFA2_SetSpeed_DMC = 8900;
IRFA_Pwr_Stat_DMC = 1;
IRFA_SetSpeed_VTC= 8784;
ISFA_Pwr_Stat_DMC= 1;
ISFA_SetSpeed_VTC= 9960;
MonEnab_CFA2_Input_Current_DMC = 0;
MonEnab_CFA2_Fan_Speed_DMC = 0;
MonEnab_CFA2_Delta_P_DMC = 0;
```

CFA1@9200, CFA-off, IRFA-off, ISFA@9960

```
CFA1_Pwr_Stat_DMC = 1;
CFA1_SetSpeed_DMC = 9200;
CFA2_Pwr_Stat_DMC = 0;
CFA2_SetSpeed_DMC = 8900;
IRFA_Pwr_Stat_DMC = 0;
IRFA_SetSpeed_VTC= 8784;
ISFA_Pwr_Stat_DMC= 1;
ISFA_SetSpeed_VTC= 9960;
MonEnab_CFA2_Input_Current_DMC = 0;
MonEnab_CFA2_Fan_Speed_DMC = 0;
MonEnab_CFA2_Delta_P_DMC = 0;
MonEnab_IRFA_Input_Current_DMC = 0;
MonEnab_IRFA_Delta_P_VTC = 0;
MonEnab_IRFA_Fan_Speed_VTC = 0;
```

72 Step 6: Verification

Analysis Questions – C1.5 – Is the operator informed when state transitions occur?

- Approach: searches for a situation, in which one of the sensor values exceeds a limit that indicates potential equipment loss:

```
!((((ISFA_EU_Temp_DMC >= 60) or
      (ISFA_Fan_Speed_VTC <= 8225) or
      (ISFA_Delta_P_VTC <= 0.400)) -> LOSS_ISFA)
and (((IRFA_Delta_P_VTC <= 0.200) or
      (IRFA_Fan_Speed_VTC <= 8225) or
      (IRFA_EU_Temp_DMC >= 60)) -> LOSS_IRFA)
and (((CFA1_Delta_P_DMC <= 0.610) or
      (CFA1_Fan_Speed_DMC <= 8225) or
      (CFA1_Input_Current_DMC <= 0.48)) -> LOSS_CFA1)
and (((CFA2_Fan_Speed_DMC <= 8225) or
      (CFA2_Input_Current_DMC <= 0.53) or
      (CFA2_Delta_P_DMC <= 0.650)) -> LOSS_CFA2));
```

- Independent of procedure
- Counterexamples found
  - Monitoring inhibited for any of the values
- Gets the operator always informed, if he wants to get informed, i.e., if he/she enabled monitoring?
- yes

```
 !(((((MonEnab_ISFA_EU_Temp_DMC and (ISFA_EU_Temp_DMC >= 60)) or
      (MonEnab_ISFA_Fan_Speed_VTC and (ISFA_Fan_Speed_VTC <= 8225) or
      (MonEnab_ISFA_Delta_P_VTC and (ISFA_Delta_P_VTC <= 0.400)) -> LOSS_ISFA)
and (... );
```

Step 6: Verification

Analysis Questions – C2.6 – Does a given action provide feedback?

- Not analyzed

74 Step 6: Verification

Analysis Questions – C3.9

Is the operator able to detect whether equipment or process is in abnormal mode?

- Is the operator always able to perform the steps required to identify the current mode?

```
(step > P1_STATE_24 * 3) and (p1_state != P1_STATE_FINISHED)
```

- yes

6. Identify current air loop target configuration

6.1 Goto Airloop Overview

PWS ECLSS
ECLSS Functional Overview

6.2 Verify CFA 1 mode

PWS ECLSS:CFA 1
CFA 1

Verify Pwr – On
Verify Fan Speed > 8800
Verify Fan Speed < 9600

cmd close **Execute**

6.3 Verify CFA 2 mode

PWS ECLSS:CFA 2
CFA 2

Verify Pwr – Off
Verify Fan Speed > 7800
Verify Fan Speed < 8200

cmd close **Execute**

6.4 Verify ISFA mode

PWS ECLSS:ISFA
ISFA

Verify Pwr – On
Verify Fan Speed > 9500
Verify Fan Speed < 10500

cmd close **Execute**

6.5 Verify IRFA mode

PWS ECLSS:IRFA
IRFA

Verify Pwr – On
Verify Fan Speed > 8400
Verify Fan Speed < 9000

cmd close **Execute**

END OF PROCEDURE

Step 6: Verification

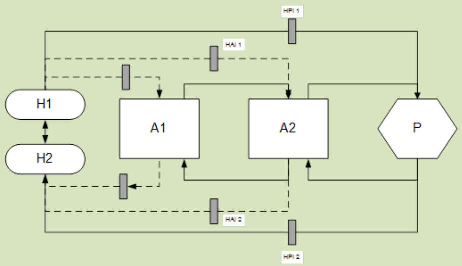
Potential for Improvement

- Approach to process modeling
  - does not guarantee completeness of process behaviour
  - Practical way if no process model available
  - Can be enhanced with further restrictions representing actual causal relationships
  - Direct use of experiences and data from system tests and operational use
- Higher level of formalism/standardization
- No detailed time model used yet
- Support for trace interpretation
- More complete model coverage (no expected-value approach)
- Model annotations specific to analysis questions: e.g., action effects, mode definitions

VASCO
STEP 7

Step 7:
Derivation of Design Requirements

Human-Automation Interaction Situations

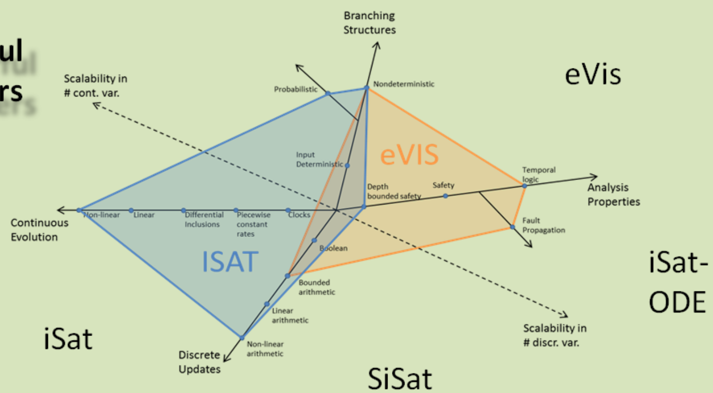


$$P{:}\ M_P = \langle S_P, s_{0_P}, V_P, Val_P, \delta_P \rangle,$$
$$I{:}\ M_I = \langle S_I, s_{0_I}, V_I, Val_I, \delta_I \rangle\ ,$$
$$T{:}\ M_T = \langle S_T, s_{0_T}, V_T, Val_T, \delta_T \rangle,$$

Formal Models and Analysis Questions

$$\exists \phi{:} S_I \rightarrow S_T \forall \sigma \in Tr(M_P) \forall s_T \in S_T, s_{0_T} \overset{\sigma}{\Rightarrow} s_T \wedge s_I \in S_I, s_{0_I} \overset{\sigma}{\Rightarrow} s_I{:}\ s_T = \phi(s_I)$$

Set of Powerful Model Checkers



STEP 7: Derivation of Design Requirements

Case Study

Analyses Questions

- **C1.3**. is the information on automation state sufficient to interact safely and efficiently with automation?
  - very hard to interpret the results above in a strict and enforcing way (no fixed, non-contextual acceptable threshold for the number of steps)
  - procedure & display (navigation) could be optimized to reduce the number of steps
- **C1.4.** does a given action cause consistent effects? and **C3.3.** can the automation, as presented on the UI, be considered as a deterministic state machine for the operator?
  - The analysis of C1.4 in Step 6 show that this question is not verified. The same action (increasing the fan speed) can provide different - and therefore inconsistent - effects, depending on the initial state (IRFA activated vs IRFA not activated).
  - The procedure is thus incompletely specified ⇨ the procedure should incorporate some definition of the appropriate execution contexts or be modified in order to induce the appropriate initial states (inhibit the monitoring of the CFA1 input current sensor values in this case)
  - More generally, better pay attention to human factors issues when designing the procedures and verify them (with a methodology like VASCO).

78 STEP 7: Derivation of Design Requirements

Case Study

Analyses Questions (continued)

- **C1.5.** Is the operator informed when state transitions (e.g., mode transitions) occur?
  - The expected property is always verified. No improvement needed.
  - Though that question hints at the importance of monitoring in the ECLSS case
- **C2.6.** Does a given action provide feedback?
  - Not handled in Step 6.
- **C3.9**. Is the operator able to detect whether equipment or process is in abnormal mode?
  - The expected property is always verified. No improvement needed.

Robustness Analysis

- Analysis shows the procedure sports some safety nets that prevent the propagation of an error (e.g., Step 3.2 of procedure 2.102.). It also shows some non homogeneity in the way the procedures and the displays are designed.
- Make visual confusions between interactive objects (on the displays) less likely
- Better support the detection of erroneous actions
- Make actions on the wrong object(s) impossible
- Improve the feedback on these actions

79 STEP 7: Derivation of Design Requirements

Objectives

- The design requirements must be derived from the results of the formal validation
- They are therefore related to the AQDB questions the formal validation was addressing
- The selected AQDB questions are used as the "design checklist".
  - CASE 1: Partial selection of AQDB questions. The questions are peculiar to the issues addressed or selected in Step 1
  - CASE 2: Complete selection of AQDB questions. A complete "check up" of the H-A system is provided.
- The requirements are about (re)designing the H-A system in terms of
  - user interface
  - automation (including allocation between H & A)
  - tasks, procedures (~ user "automation")
  - learning (including operational documentation)
  - training
  - unforeseen additional means, such as artefacts

STEP 7: Derivation of Design Requirements

Methods (1/2)

- How to proceed?
  - **Single trace analysis**
    - Consists in examining each (counterexample) trace separately
    - Executing the trace on some kind of simulator should be insightful
    - The objective is to understand why and where in the trace the associated AQDB question fails
    - Example:
      - H-A Issue: Experience feedback shows user does not execute a procedure adequately
      - Modeling, verification and trace: show the user cannot perceive a given key information to determine the mode the system is in, leading to the errors observed during operations
      - Design requirements: improve the perception of the system mode
      - Design solution: increase the salience of the system mode information, for example by highlighting when it changes

81 STEP 7: Derivation of Design Requirements

Methods (2/2)

- How to proceed?
  - **Global trace analysis**
    - Analyze multiple trace together, possibly including satisfying and non satisfying (counterexamples) ones.
    - Attempt to identify common causal factors behind the counterexamples.
    - Example:
      - H-A Issues: user makes inappropriate decisions that compromise safety
      - Modeling, verification and traces:
        - Show that the Situation Awareness AQDB questions are frequently violated
        - Multi-trace analysis show that this occur in some scenarios only, when user workload is high (many tasks to perform).
      - Design requirements: reduce user workload during the corresponding phase of operation
      - Design solutions: automate some user tasks, prepare some user tasks earlier, add an additional user (e.g., assistant).