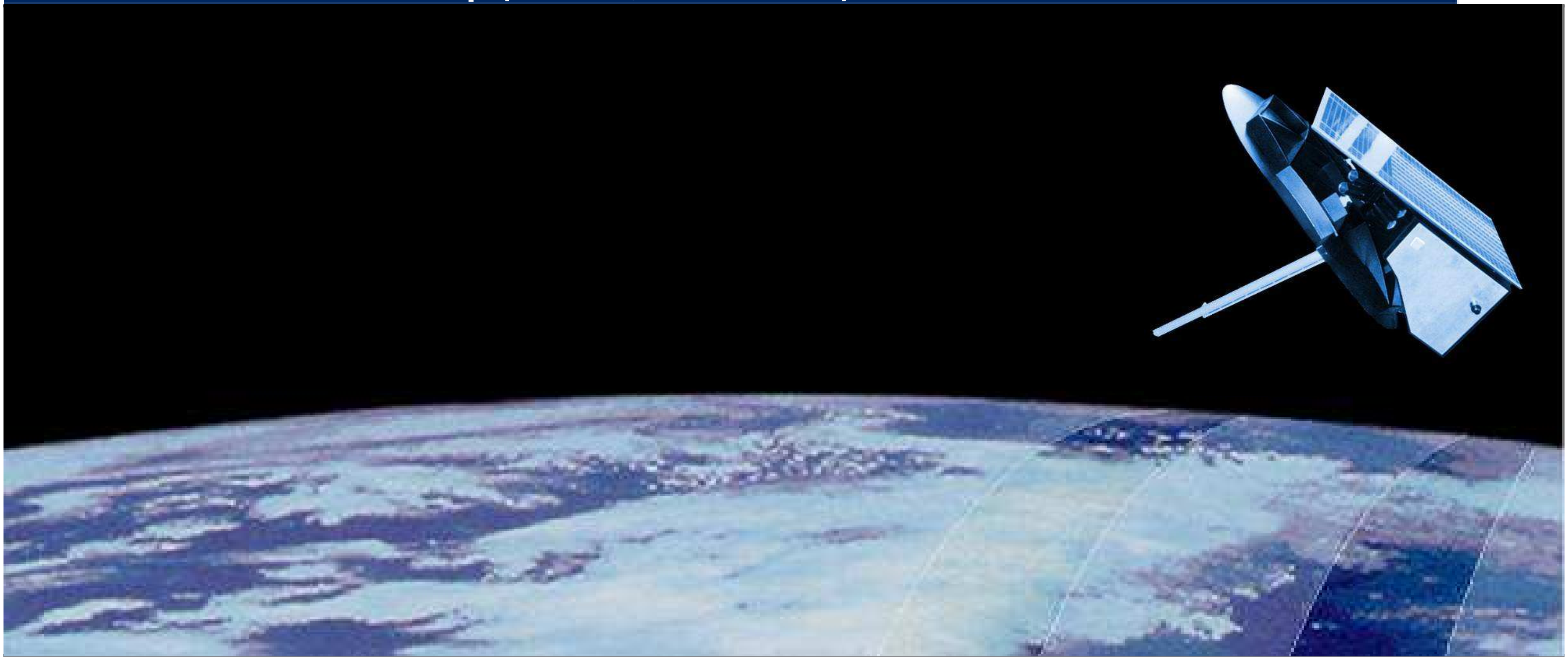
FDIR Development Practices Based on a Probabilistic Reasoning Approach

ESA ADCSS Workshop (ESTEC, 26.10. 2011)



SCOPE

- **FDIR Background and Challenges**
  - FDIR Definition & Prior art
  - Requirements of the proposed FDIR methodology
- **FDIR & Probabilistic Reasoning using Bayesian Networks**
  - Bayesian Networks & Troubleshooting
  - An illustrative example
  - How Bayesian approach can meet FDIR requirements
- **Future Developments**
  - Bayesian Based FDIR C.A.S(ystem).E.
  - Bayesian Based FDIR SW Building Blocks (Satellite OSW & Ground Segment)
- **Project Partnership**

FDIR Background and Challenges

FDIR Concept & Prior Art - 1

- **Failure tolerant technology & FDIR**
- **The FDIR functions to satisfy the specific autonomy, reliability and availability needs for a space mission.**
- **FDIR as a relevant topic dealt with the entire project life-time.**
- **FDIR functionality improvement and update throughout the whole mission**
- **Current FDIR development practices:**
  - RAMS/FMECA/FTA
  - FDIR specification and development
- **Lack of a strong analytical methodology supporting system-level conception and implementation of diagnostic procedures**

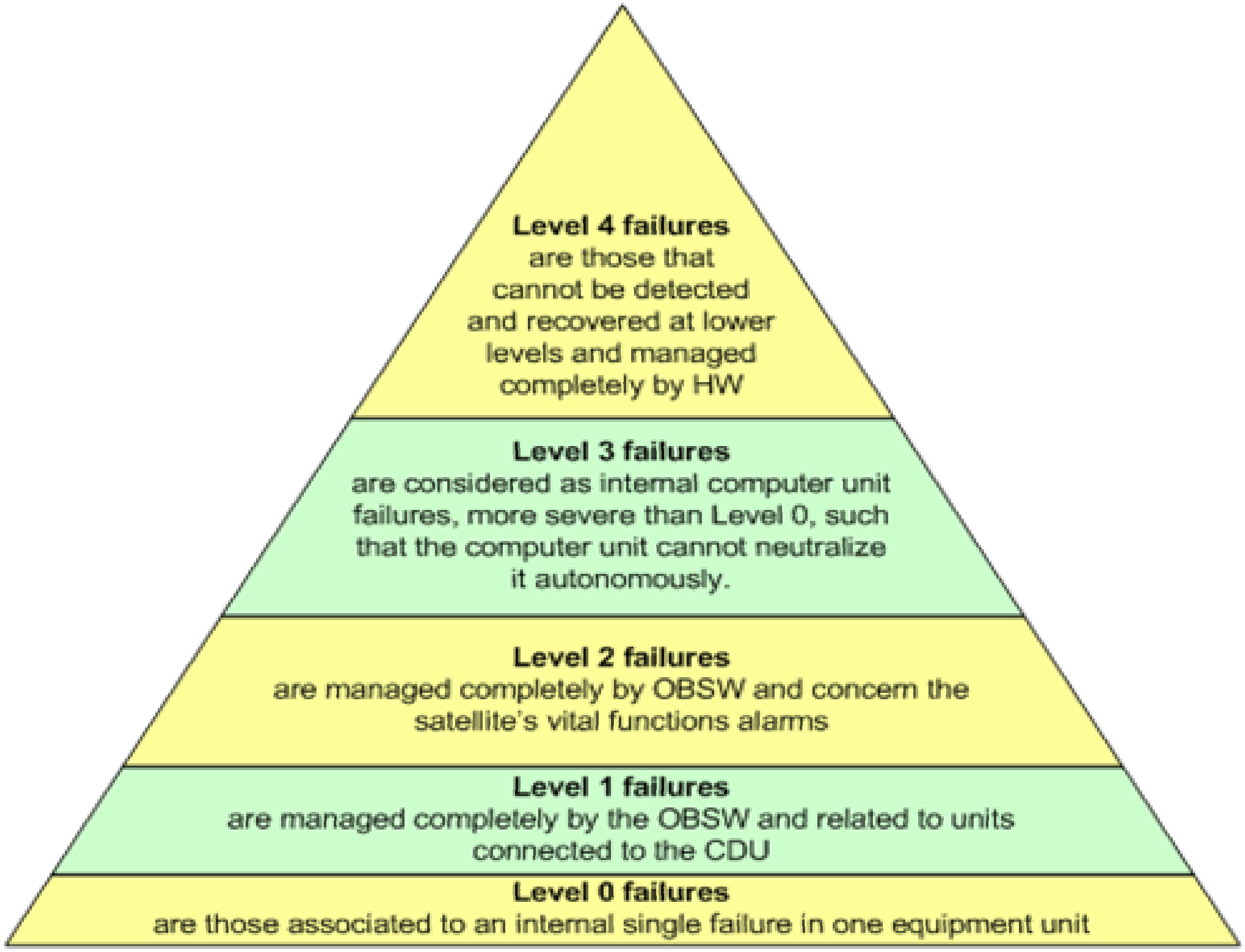
FDIR Concept & Prior Art - 2

- **FDIR scope: Space System as a whole**

- **FDIR functions organized in a hierarchical breakdown (see figure in the next slide from GalileoSat FOC Project)**

- **Additional considerations (and subjects for further discussion):**
  - Level 2: failures related to complex and mutual-related issues, detected by the OBC processor which collects and analyses information coming from different units and subsystems; recovery performed by the On-Board Computer (OBC) processor
  - Level 3: failures related to the OBC, which are neither detectable nor neutralizable by the OBC processor (e.g. EDAC modules, processor watchdog)
  - Level 4: failures identified by ad-hoc surveillance systems, fully independent from the OBC, both allocated within the satellite platform (e.g. Power Alarm and Thermal Limit detectors) and/or the Ground Control Segment (e.g. TM data off-line analysis).

FDIR Concept & Prior Art - 3



Requirements of the proposed FDIR methodology - 1

- **Uncertain/stochastic information processing**
- **Direction of the reasoning not fixed in advance**
  - i.e. deduced via the collection of the information
- **Supporting Recovery Decisions**
- **FDIR focused on unit/SS models**
  - Models built from the expert knowledge and not only from sample/quality data
  - Interaction between the different models (divide & rule strategy)
    - top-down strategy to conceive S/S-level FDIR models
    - and bottom-up strategy to build up the system-level FDIR model

Requirements of the proposed FDIR methodology - 2

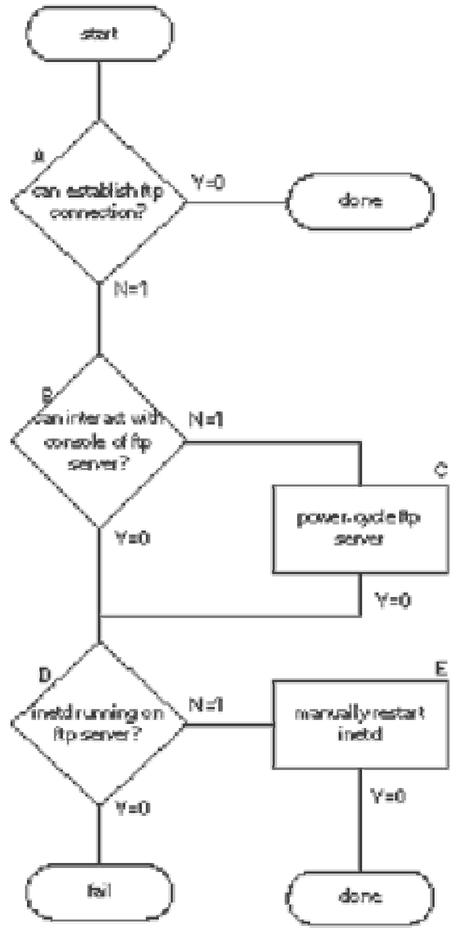
- **Flexibility and Generality**
  - Adaptation to different contexts and the acquired experience
  - OO paradigm: model inheritance and composition
- **Robustness**
- **Efficiency (Computational Resources)**
- **To support both system-level FDIR conception and the implementation of the related diagnostic procedures**
  - System & subsystem engineers to adopt the same models and the same terminology
  - Model Composition & Integration

FDIR & Probabilistic Reasoning using Bayesian Networks

Bayesian Networks & Troubleshooting - 1

Thinking about FDIR ...

- FDIR is generally based on procedures
- This offers a deterministic and tested approach to fault detection, isolation and recovery
- But…
  - Alternative solutions are not explored
  - It is rigid to unexpected scenarios
  - Support to ground recovery is not flexible



Bayesian Networks & Troubleshooting - 2

Bayesian Approach to Fault Identification and Recovery

- The path to recover a fault can be driven by probabilistic reasoning, taking into account the trade-off between costs and repairing probability.
- As far as new evidence is collected, some observations and actions become more appropriate than others.
- This can lead to a more flexible and adaptive approach to FDIR.

$$P(H/E) = \frac{P(E/H)\ P(H)}{P(E)}$$

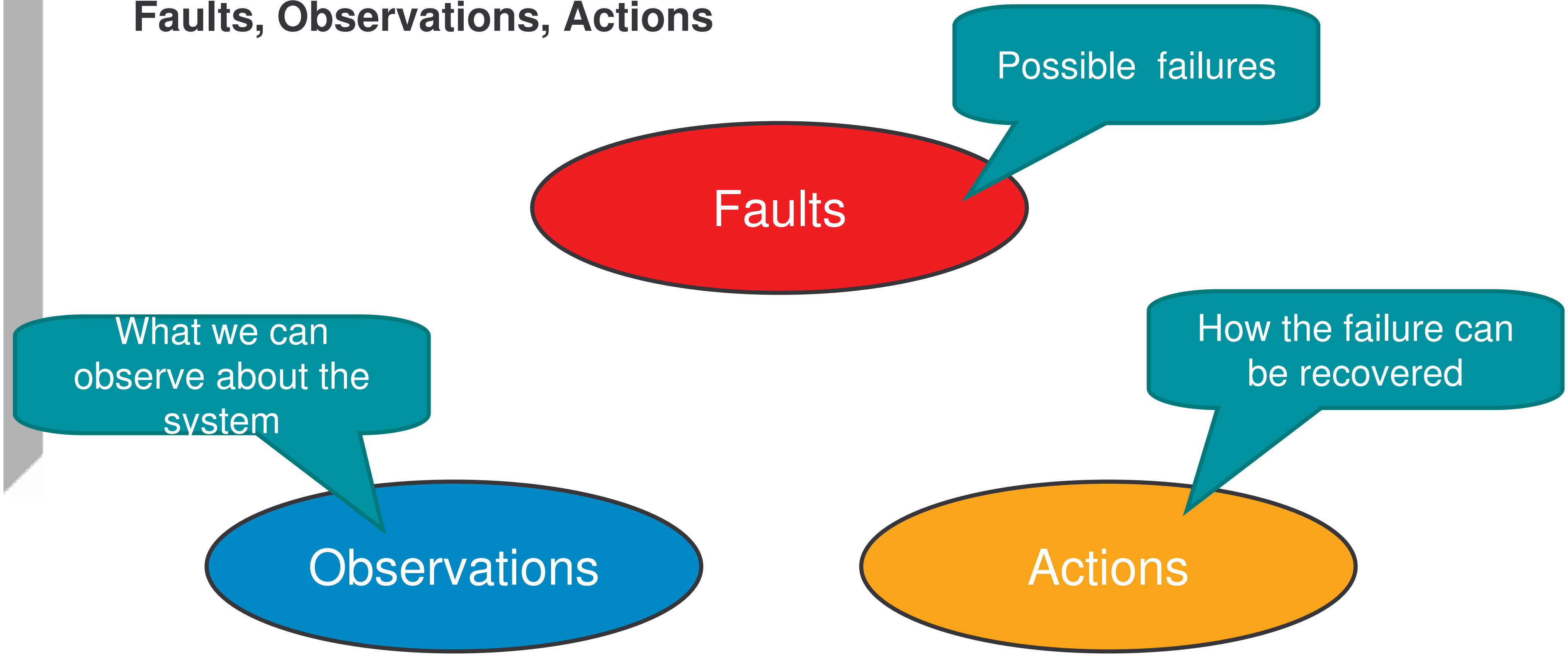
where $P(H)$ = the previous or *a priori* probability that the hypothesis is true

$P(E)$ = the probability that an event will occur

$P(E/H)$ = the probability that the event will occur given that the hypothesis is true

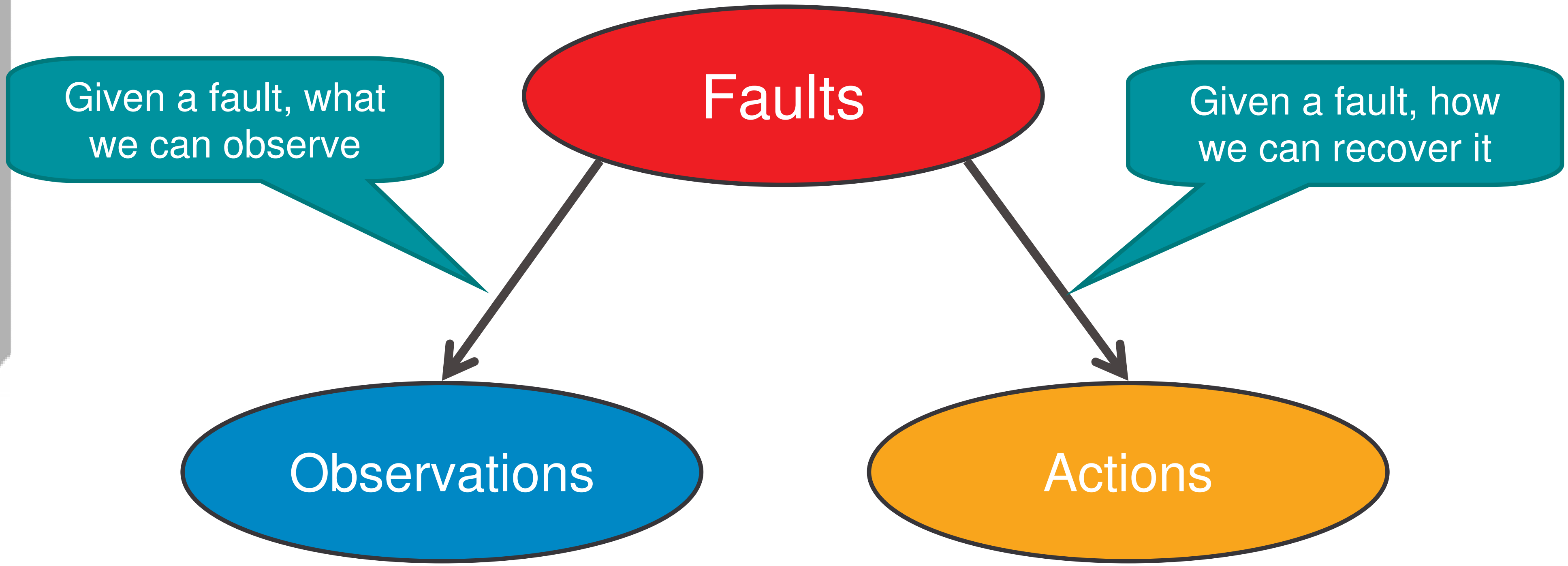
Bayesian Networks & Troubleshooting - 3

Faults, Observations, Actions



Bayesian Networks & Troubleshooting - 4

Faults, Observations, Actions



NOTICE: Relationships are not deterministic, but probabilistically determined.

Bayesian Networks & Troubleshooting - 5

Bayesian Troubleshooting

- Relationship between Faults, Actions and Observations
  - Bayesian Network Structure
- Cause-effects are not deterministic
  - Conditional probabilities
- Actions and Observations have an associated cost (e.g. time)
  - Decision utility

Bayesian Networks & Troubleshooting - 6

Decision Utility

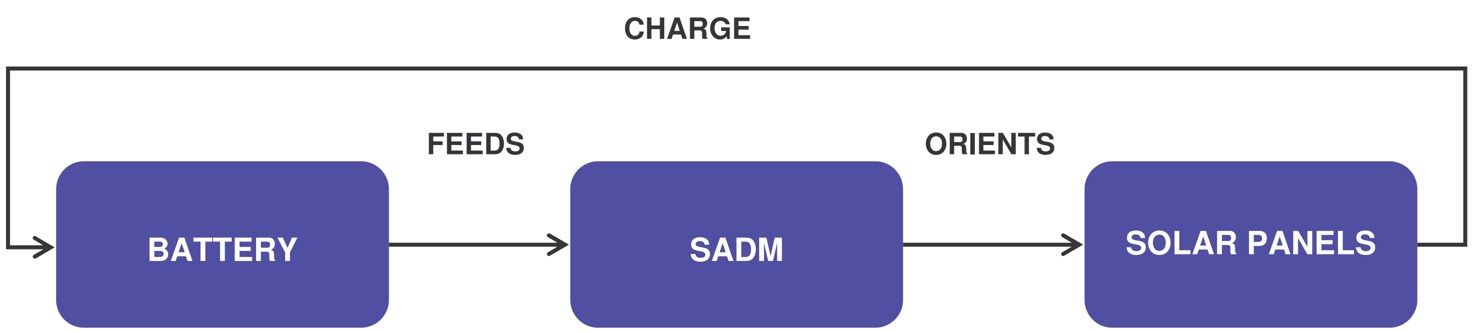
- Optimality objective:
  - **To minimize the cost of recovery (e.g. time)**

- Cost Sources:
  - **Expected Cost of Repairing (ECR)** is the cost we expect to have for recovering the failure
  - **Expected Cost of Observation (ECO)** is the repairing cost we expect after an observation is collected

An illustrative example - 1

- FDIR model integrating the following S/Ss
  - Battery
  - SADM
  - Solar Panels

- S/S models interactions:



- Battery low
- Different levels

- RPS not found
- Potentiometer broken
- Step counter broken
- Step counter not calibrated

- Orientation not correct
- Degraded solar arrays

An illustrative example - 2

Integrated Model: Fault Tree

- Generic Issue (100.00 %)
  - battery (25.00 %)
    - battery low (50.00 %)
    - different levels (50.00 %)
  - solar arrays (25.00 %)
    - orientation not correct (50.00 %)
    - degraded solar arrays (50.00 %)
  - SADM (40.00 %)
    - RPS not found (33.33 %)
      - step counter broken (50.00 %)
      - step counter not calibrated (50.00 %)
    - power is not sufficient (33.33 %)
    - potentiometer broken (33.33 %)
  - links (10.00 %)

An illustrative example - 3

Integrated Model: Observations

- VBAT1 = VBAT2
  - **YES**
  - NO
- battery-SADM-panels connection
  - **correct link**
  - NO SADM-BAT link
  - NO PANELS-SADM link
  - NO BAT-PANELS link
- solar arrays position
  - **able to obtain the maximum voltage**
  - unable to obtain the maximum voltage
- VBAT > VBAT LOW
  - **YES**
  - NO
- step counter impulse and potentiometers voltage
  - **consistent values**
  - step counter shows a mismatch with both potentiometers and both potentiometers provide a consistent position
  - step counter shows a mismatch with both potentiometers and there is a mismatch between the 2 potentiometers
  - step counter does not match with a single potentiometer voltage
- state of potentiometers, step counter and RPS
  - **consistent values**
  - potentiometer not to RPS and step counter reseted
  - potentiometer to RPS and step counter not
  - RPS not found after 360°
- solar arrays voltage
  - **nominal voltage**
  - not nominal voltage - with eclipse
  - not nominal voltage - without eclipse
- step counter adjustments
  - **anyone**
  - 1
  - 2
  - 3

An illustrative example - 4

SADM Model: Repairing Actions

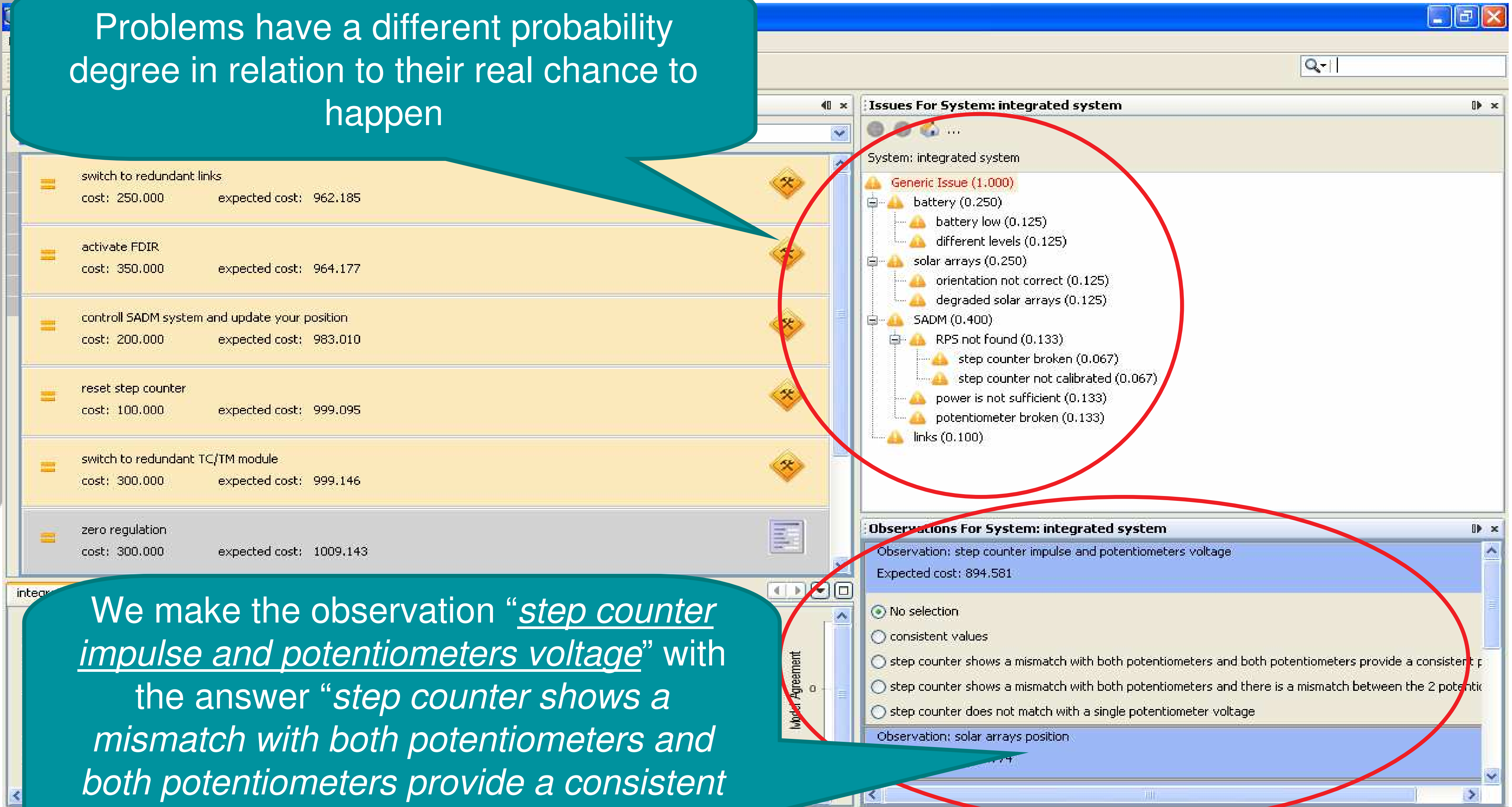
- switch to redundant links
- update the land
- switch to redundant TC/TM module
- switch to redundant step counter
- reset step counter
- activate FDIR
- controll SADM system and update your position

An illustrative example - 5

- Once the model is fully specified, it can be delivered (on board?) in order to drive the B-FDIR component in solving a problem, once a failure becomes or it is likely evident …
- For example, sometime, somewhere in the space, let us suppose the step counter stops working properly…
- We don't know where the problem is… but Huston we know there is a problem!!
- Let's discover and solve it!!

An illustrative example – 6

Problems have a different probability degree in relation to their real chance to happen



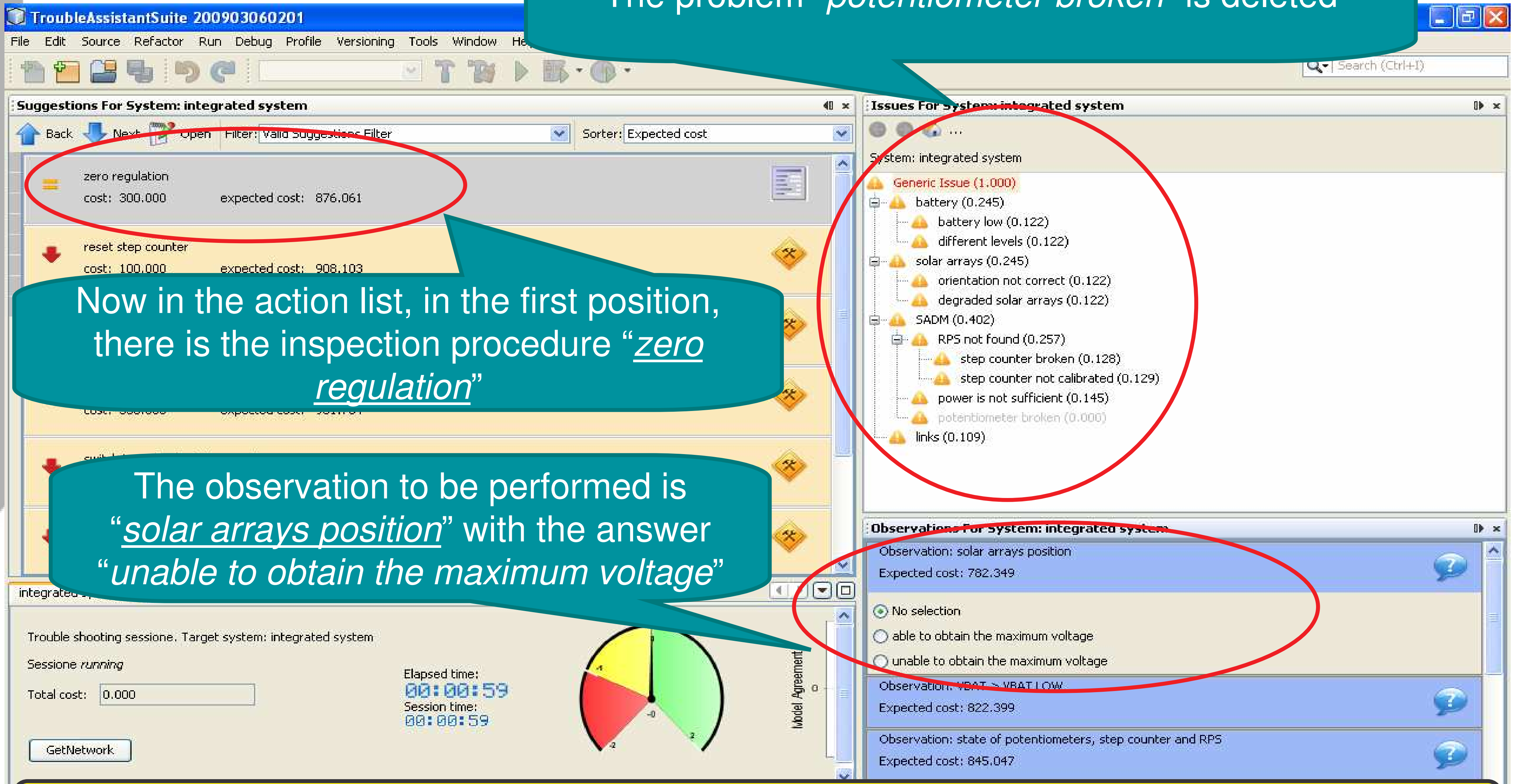
We make the observation "*step counter impulse and potentiometers voltage*" with the answer "*step counter shows a mismatch with both potentiometers and both potentiometers provide a consistent position*"

An illustrative exam

The problem “*potentiometer broken*” is deleted

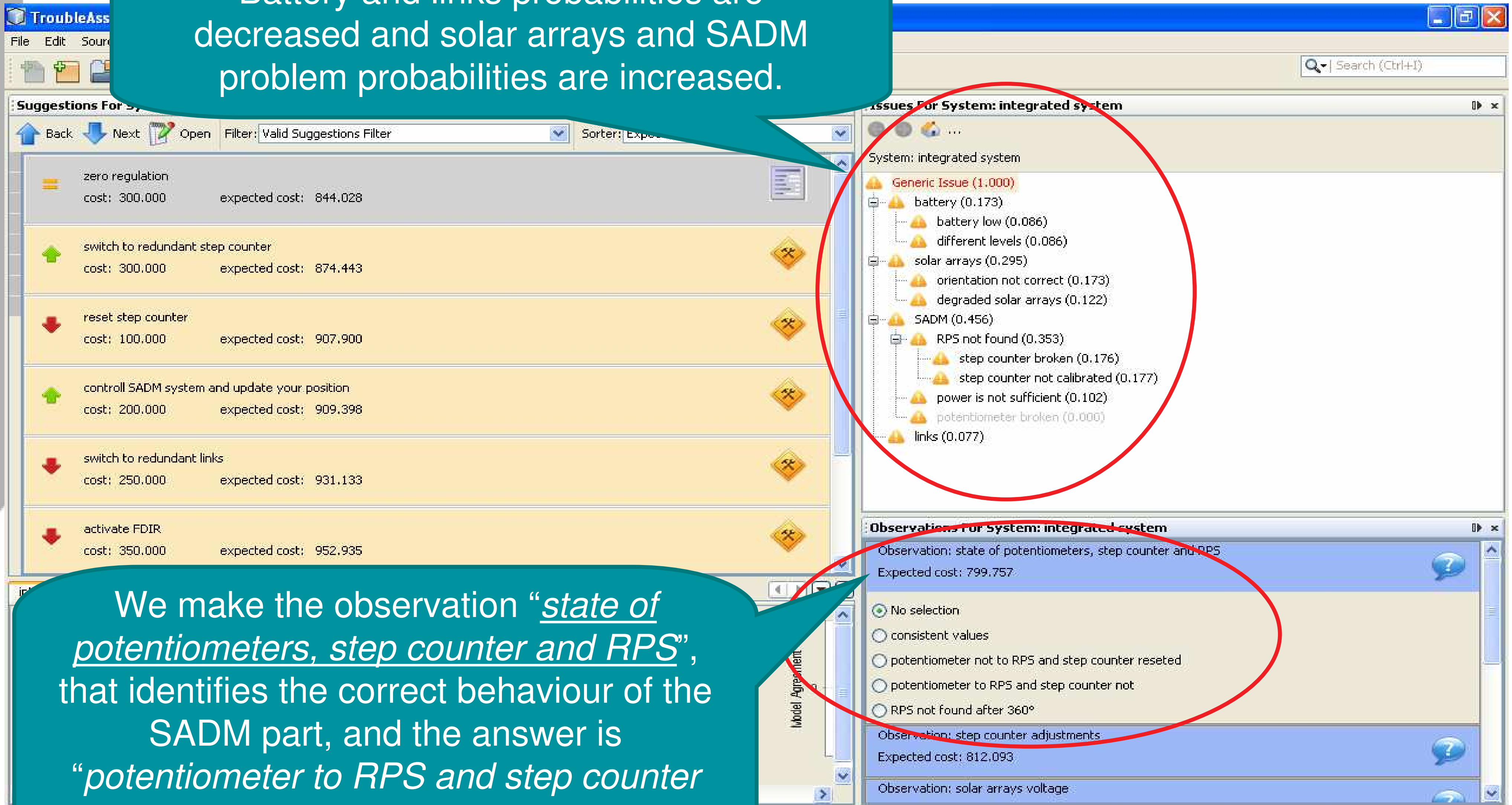
Now in the action list, in the first position, there is the inspection procedure “*zero regulation*”

The observation to be performed is “*solar arrays position*” with the answer “*unable to obtain the maximum voltage*”



INSPECTION PROCEDURE is a particular kind of preliminary investigation that includes actions and observations: in this case the inspection procedure can exclude or focus on the SADM issues.

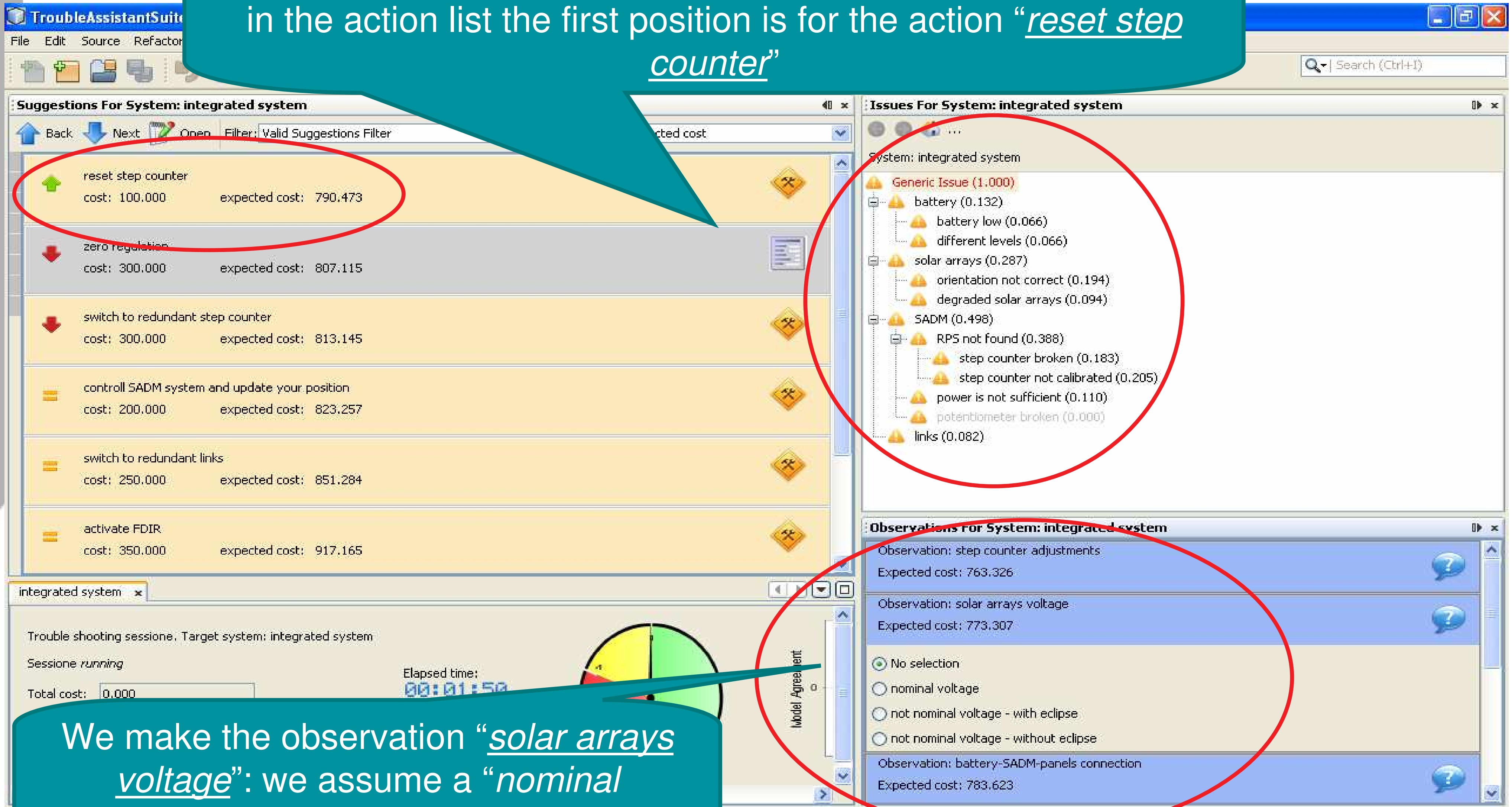
Battery and links probabilities are decreased and solar arrays and SADM problem probabilities are increased.



We make the observation "*__state of potentiometers, step counter and RPS__*", that identifies the correct behaviour of the SADM part, and the answer is "*potentiometer to RPS and step counter not*".

An

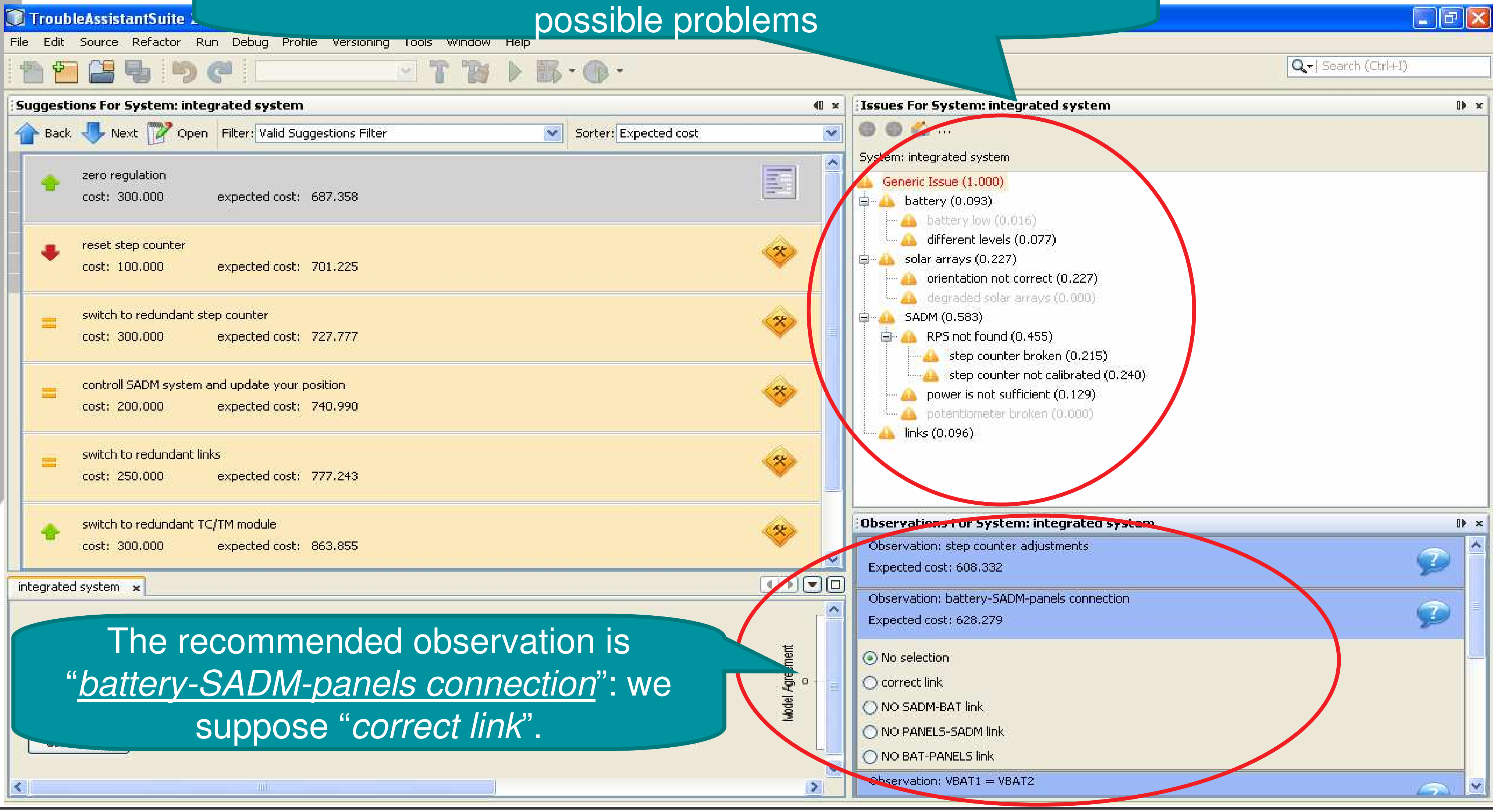
Problem probability is changed and, as we have increased the probability of the problem "*step counter not calibrated*", in the action list the first position is for the action "*reset step counter*"



We make the observation "*solar arrays voltage*": we assume a "*nominal voltage*"

An i

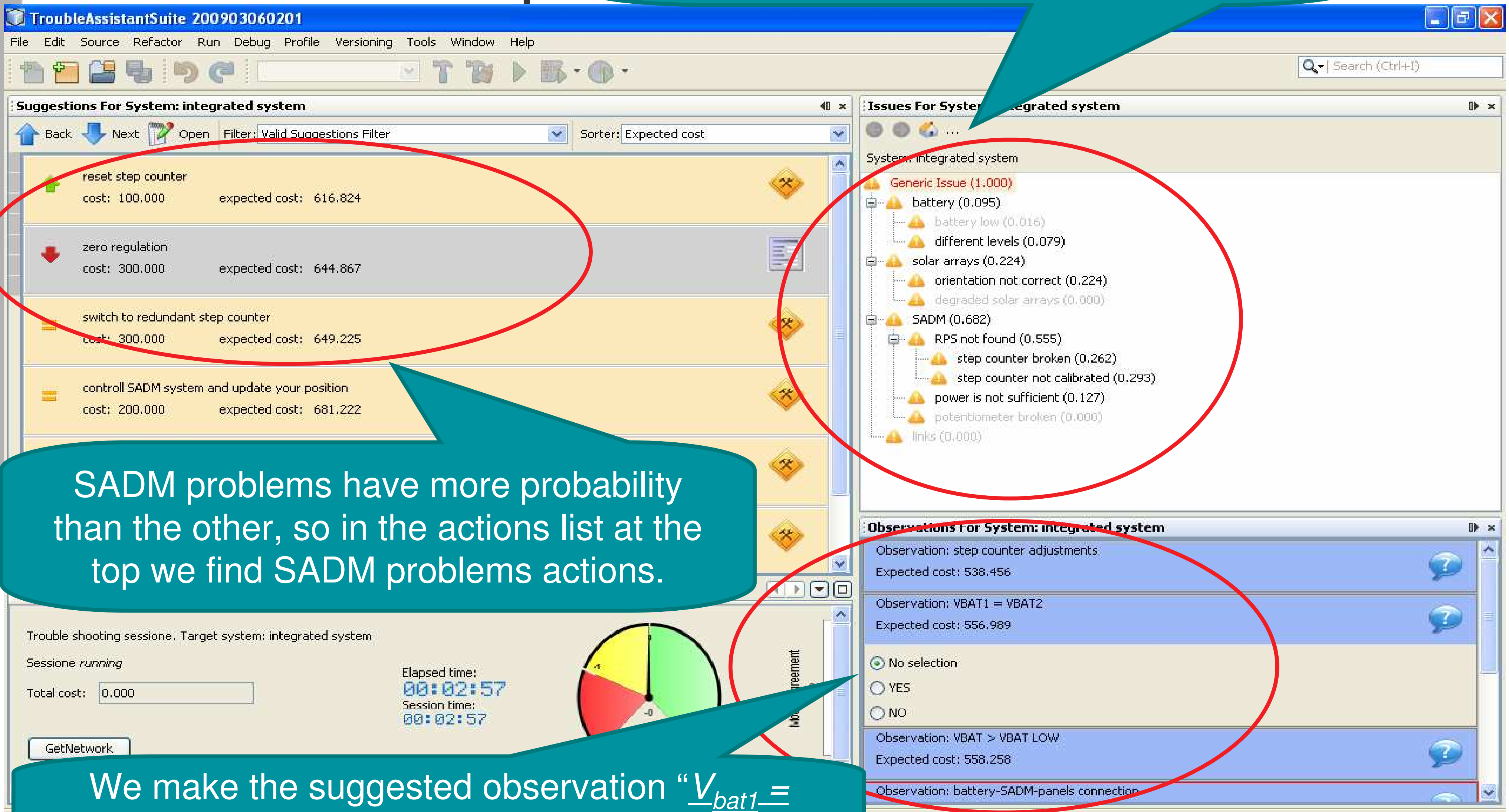
Having gathered “*nominal voltage*”, we exclude “*battery low*” e “*degraded solar arrays*” from the possible problems



The recommended observation is “battery-SADM-panels connection”: we suppose “*correct link*”.

An illustrative example

"L*ink*" problems were eliminated

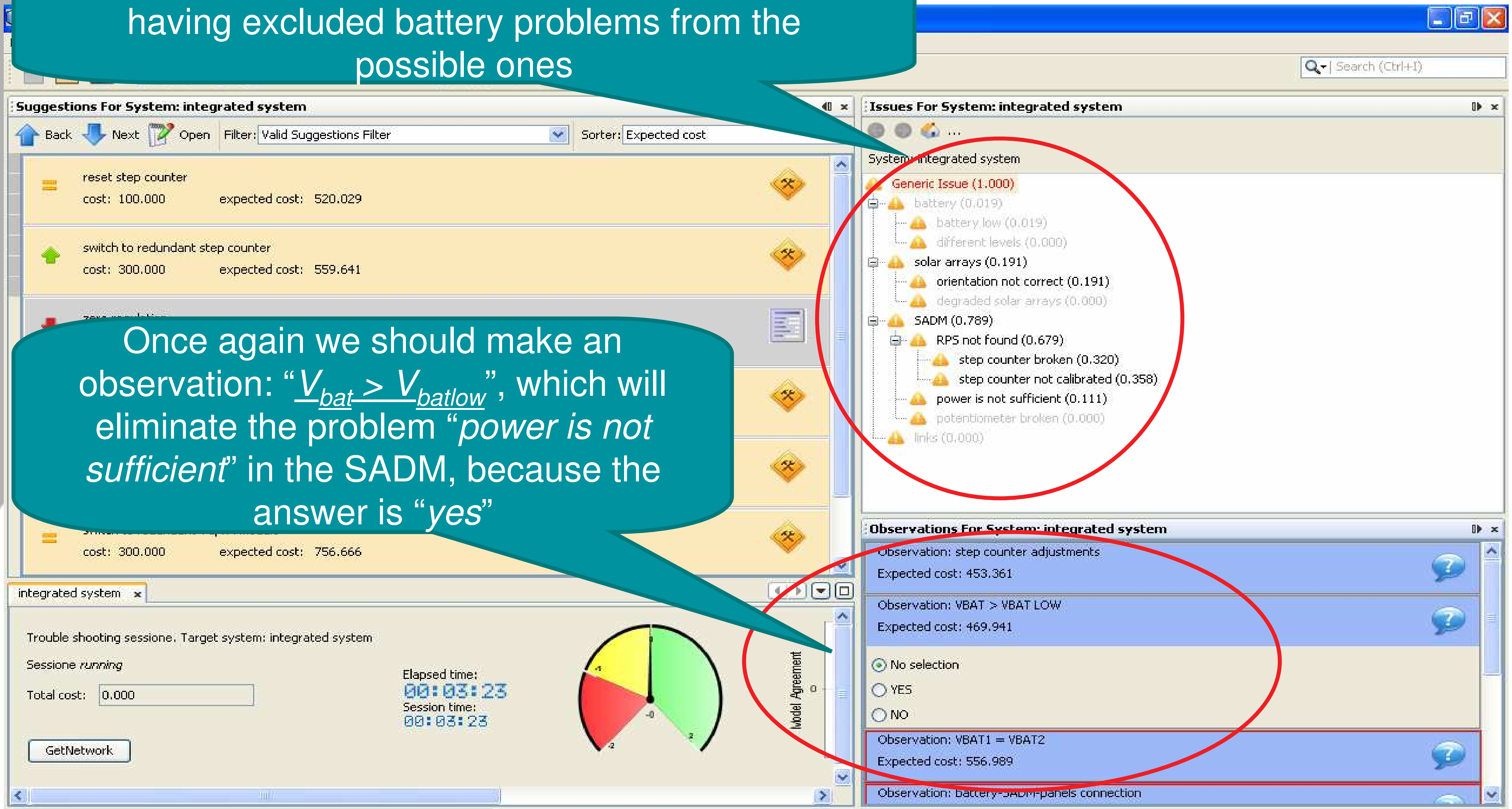


SADM problems have more probability than the other, so in the actions list at the top we find SADM problems actions.

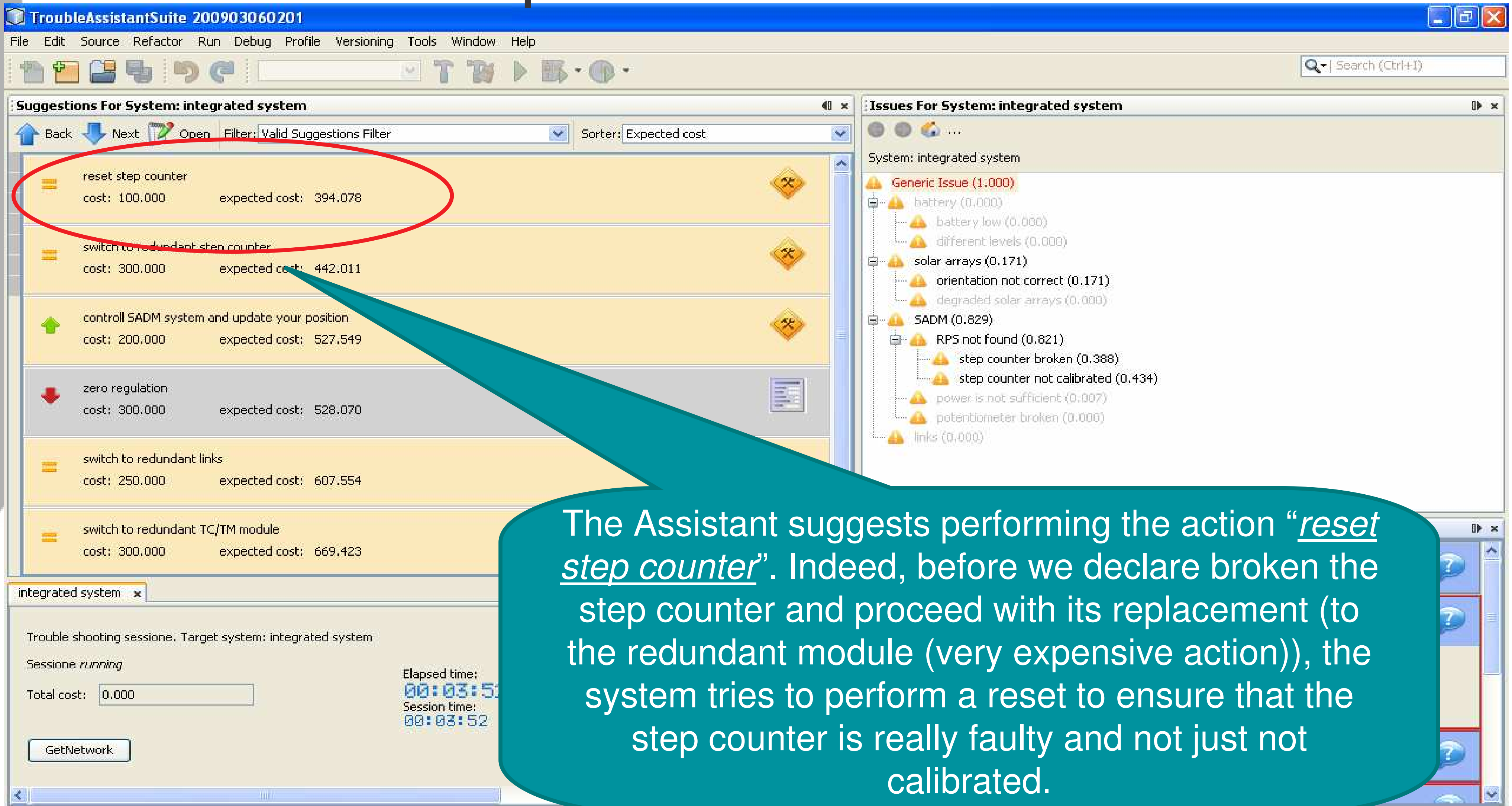
We make the suggested observation "$V_{bat1} = V_{bat2}$" with the answer "*yes*".

The problem “*different levels*” is eliminated as well, having excluded battery problems from the possible ones

Once again we should make an observation: “$\underline{V_{bat} > V_{batlow}}$”, which will eliminate the problem “*power is not sufficient*” in the SADM, because the answer is “*yes*”

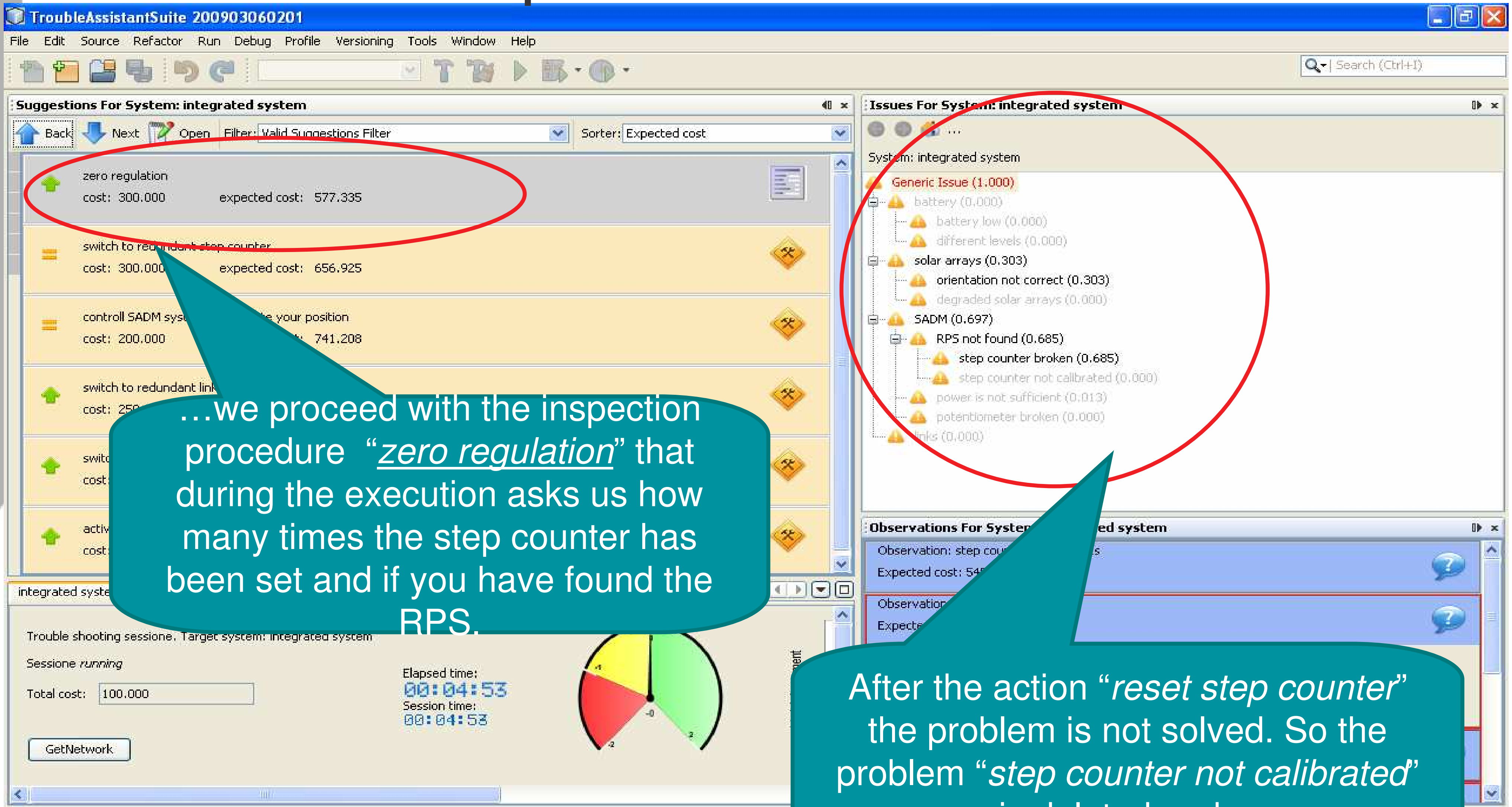


An illustrative example - 13



The Assistant suggests performing the action "*reset step counter*". Indeed, before we declare broken the step counter and proceed with its replacement (to the redundant module (very expensive action)), the system tries to perform a reset to ensure that the step counter is really faulty and not just not calibrated.

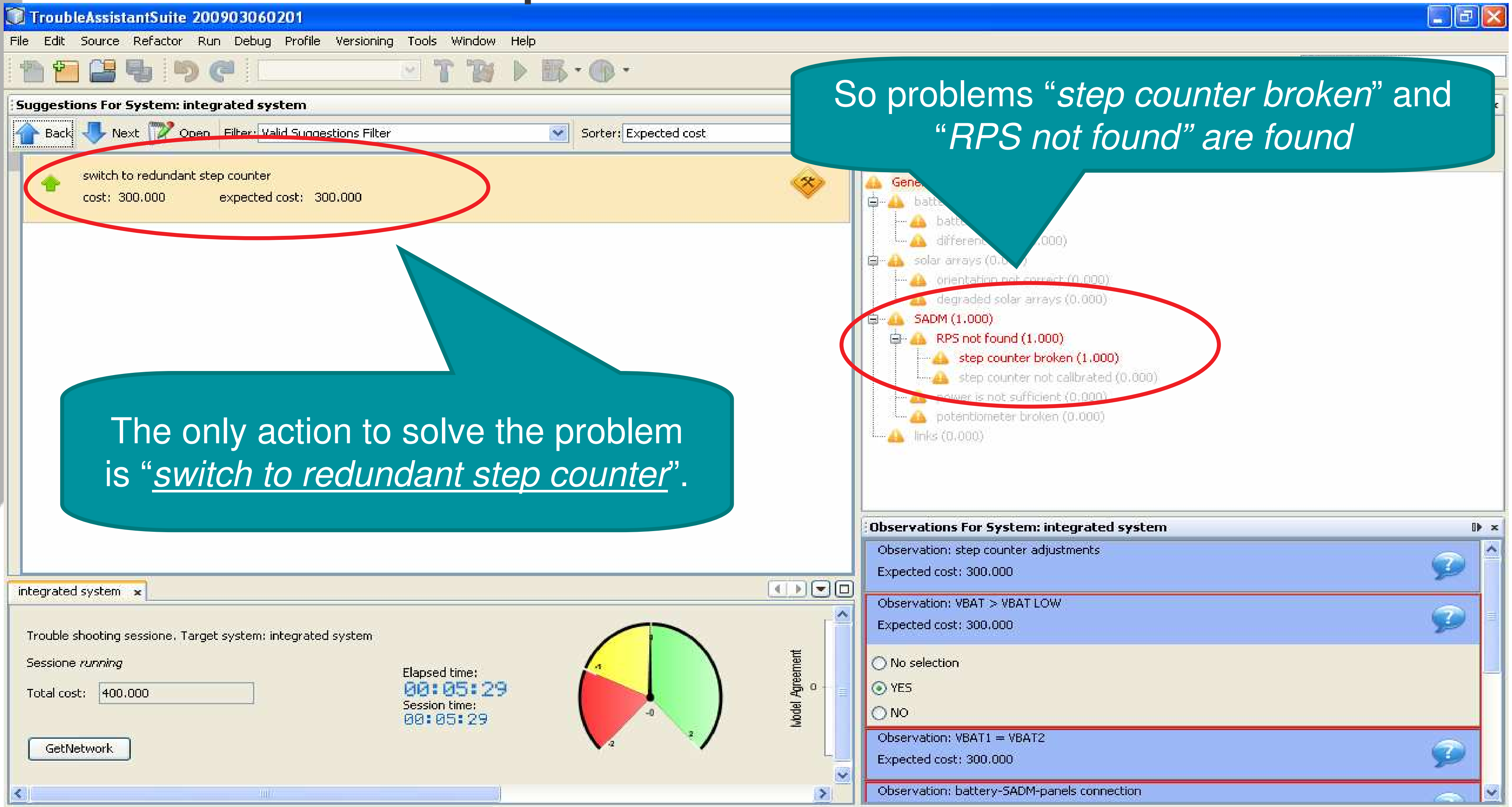
An illustrative example - 14



…we proceed with the inspection procedure "*zero regulation*" that during the execution asks us how many times the step counter has been set and if you have found the RPS.

After the action "*reset step counter*" the problem is not solved. So the problem "*step counter not calibrated*" is deleted and…

An illustrative example - 15



How Bayesian approach can meet FDIR requirements - 1

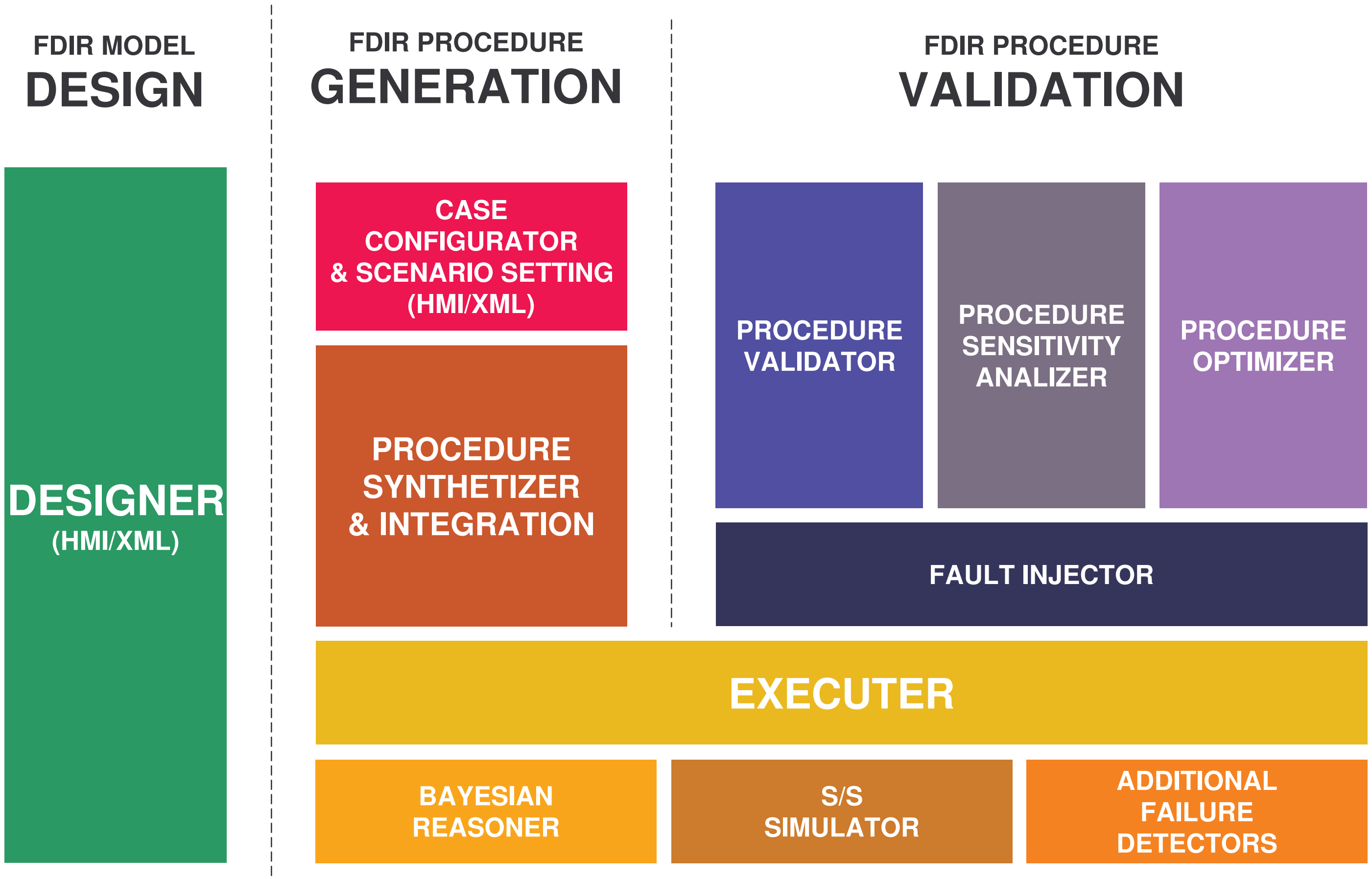
- **Uncertain/stochastic information processing**
  - Bayesian networks are deeply-investigated, robust and well-defined methods to handle uncertainty in causal models
- **Direction of the reasoning not fixed in advance**
  - Bayesian based FDIR is able to address a problem dynamically by choosing most appropriate actions as far as new evidence regarding the failure is being collected
- **Supporting Recovery Decisions**
  - Recovery can be performed automatically or by interacting with humans, supporting them in selecting best options.
- **FDIR focused on unit/SS models**
  - Bayesian models can be organized, instanced and integrated by sub-models in order to cope with the system structure and complexity

How Bayesian approach can meet FDIR requirements - 2

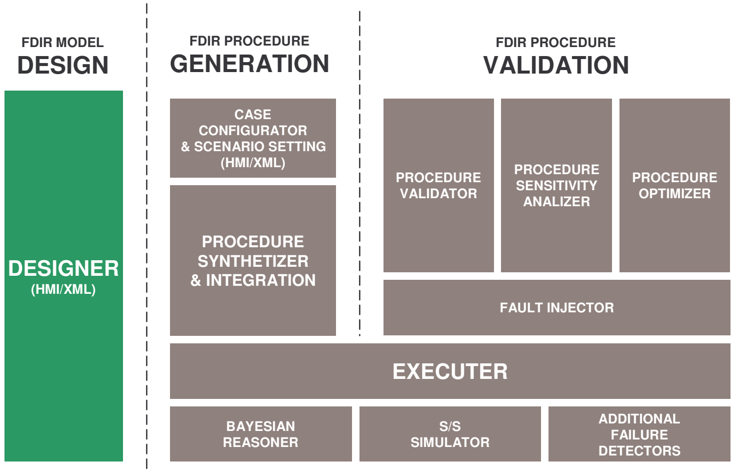
- **Flexibility and Generality**
  - Algorithms are coded only once, with their behaviour controlled via model parameters, thus reducing validation and maintainability activities
  - Models can be composed and/or extended.
- **Robustness**
  - Models do not strictly depend on parameters, as we are interested only in the order actions are sorted and suggested (i.e. outcomes do not fluctuate radically as consequence of minor changes in the Bayesian Network)
- **Efficiency (Computational Resources)**
  - Efficient algorithms to deal with Bayesian inference
  - But... attention should be paid to on-board implementations.
- **To support both system-level FDIR conception and the implementation of the related diagnostic procedures**
  - Bayesian models at S/S level can be easily integrated in larger models at system level and reused for FDIR procedure implementation.

Future Developments

Bayesian Based FDIR C.A.S(ystem).E. - 1



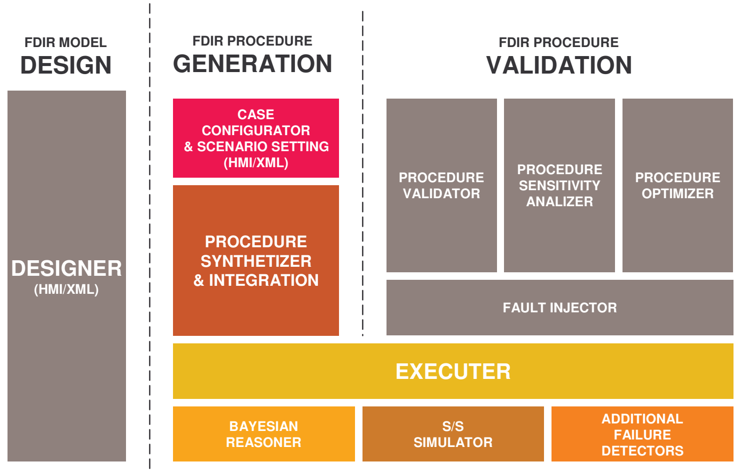
Bayesian Based FDIR C.A.S(ystem).E. - 2



S/S FDIR Model Structure & Parameter definition:

- failures, observations, and action
- failure occurrences
- observation/failure correlation indices
- action & observation costs
- action/failure repairing indices

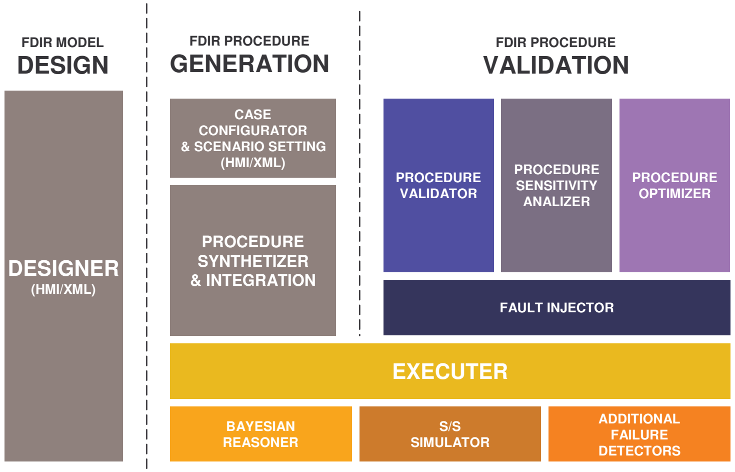
Bayesian Based FDIR C.A.S(ystem).E. - 3



FDIR Procedure Generation:

- FDIR Models are used as inputs
- FDIR Models translation into Bayesian Networks (Executer)
- Scenario and CASE Configurations (from the user)
- Scenario Execution (Executer, with the support of the Bayesian Reasoner, S/S Simulator & Additional Failure Detectors)
- Procedures are built and integrated
- Procedures made available on target support
- Additional Failure Detectors: limit monitoring, correlation tests, S/S & unit built-in test, Kalman filtering.

Bayesian Based FDIR C.A.S(ystem).E. - 4



FDIR Procedure Validation:

Failure injection to :

- validate the generated procedures (Procedure Validation)
- stress the procedure robustness (Procedure Sensitivity Analizer)
- tune and optimize the FDIR model Parameters (Procedure Optimizer)

Bayesian Based FDIR SW Building Blocks

- **FDIR SW Building Blocks: Core (Algorithms) & declarative I/F**
- **Algorithms configurable via tables**
  - decisional trees, scenarios, inputs and sub-systems
- **Applicability to levels 2 (OSW) and 4 (Ground Segment)**
- **OSW/GS FDIR centralised or distributed approach**
- **Reusability, deployment, reconfigurability, portability without carrying out a time-consuming V&V process (the core does not change)**
  - Model/Configuration Table V&V
- **Operational benefits and FDIR adaptability to the lessons learnt during the phase E.**
  - Patching in PM/SGM RAM of the only FDIR configuration tables

Project Partnership

- **OHB System AG** (B. Brünjes)
  - Satellite System Integrator, Software Design and Development for On-board Systems, Test Facilities, Simulators, and Mission Control Centres
- **CGS SPA** (M. Tipaldi)
  - Small Satellite Manufacturer, Space Systems Design, ISS facility, EO Applications, Ground Segments, Software Engineering
- **University of Sannio** (L. Troiano)
  - ranked top 20 for teaching and research quality in Italy, excellence in research regarding Computational Intelligence, Software Engineering, Software Architectures, Security, Process modelling
- **Intelligentia s.r.l.** (D. De Pasquale & M.C. Vitelli)
  - spin-off company of University of Sannio focused on Computational and Artificial Intelligence