MADE's main interface (Figure B1) includes a *Project Explorer* and supporting libraries (i.e., *Failure Concepts*, generic *Palette*, and a *Local Library* of user defined components), relational diagrams (functional and failure), and continuous error notification (*Problem/Connection Viewer*).

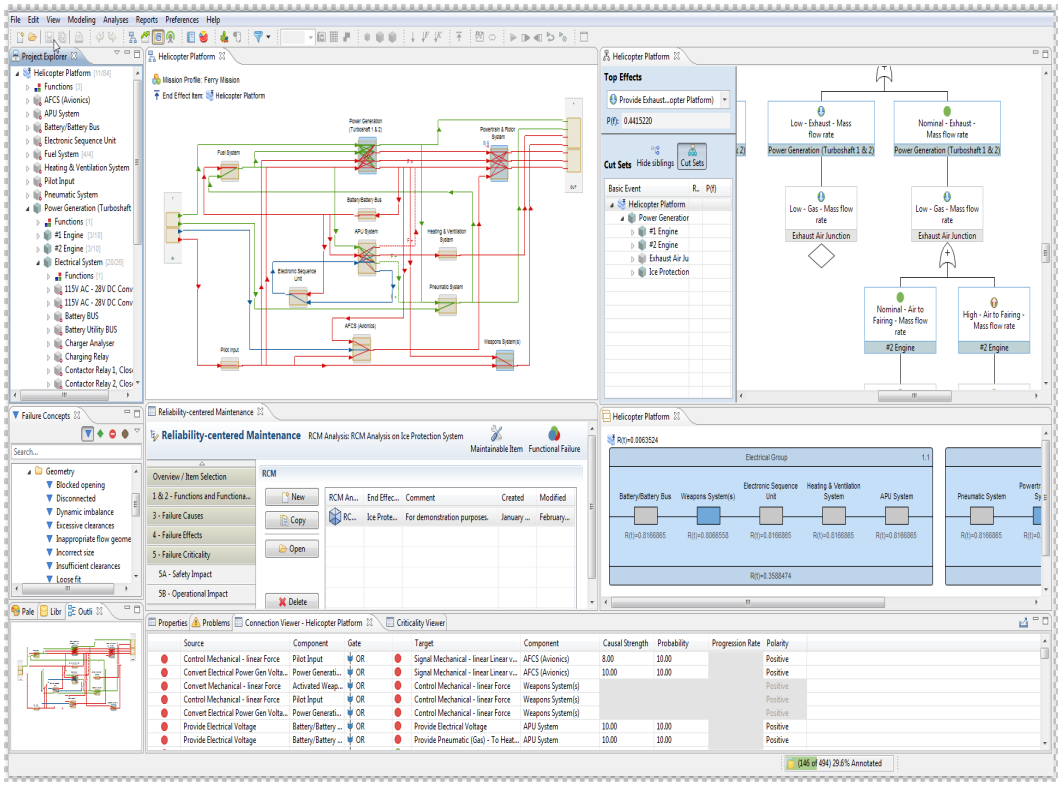

Figure B1: MADE Interface

To model a system in *MADE*, hierarchical structures and characterizations are required to constitute the model to support mission assurance activities and generate risk-based artifacts. Therefore, it is recommended that the Techniques and Tips in the subsequent sections be followed to:

I. Create/Import system hierarchy in *Project Explorer* and define *Mission Profile*.
II. Refine/Enhance *functional block diagrams* with functions, flows, supporting narratives, and documentation for each element.
III. Add/Refine and integrate mission-assurance discipline-specific modeling.
    i. Reliability, Maintainability, and Availability (RMA) Engineering - design/assurance-specific failure, maintenance, and availability characterization parameters (i.e., failure causes/mechanisms, modes, and symptoms, detection methods, compensating provisions, occurrence probability, criticality/severity of the occurrence, and spares)
    ii. System Safety - design/assurance specific hazard category and condition parameters (i.e., causes, controls, consequences, severity, occurrence, and verifications).
    iii. Software Assurance - TBD
    iv. Quality Engineering - TBD
IV. Import requirements and link them to the model for compliance assessment, verification, and tracking.
V. Verify and validate modeling with error adjudication, simulation, and peer consults.
VI. Generate mission-assurance reports/assessments and requirement verifications/compliances.
VII. Iterate and update modeling (using Steps I – VI).

Appendix B : Recommended MADE Modeling Techniques & Tips

I. Create/Import system hierarchy in Project Explorer and define Mission Profile

A. Create a new Project and Profile

1) Use the MADE file menu to open a *New Project* (Figure B2).
2) Use the MADE main toolbar *Profile* option under modeling or right-click on the newly created project in the *Project Explorer* pane to open the *Mission Profile* window and define mission parameters: mission length, mission phases and their length, and phase segments. The *Mission Profile* (Figure B3) can be updated during the modeling process with component duty cycles, numerical values, and functional limits at the component level.
3) Mission Success Criteria should also be established in accordance with mission assumptions and requirements. The *Mission Profile* (Figure B3) can also be updated during the modeling process with individual component's duty cycle, numerical values, and functional limits at the component level.

B. Create/Import[1] system hierarchy

1) Use the *Project Explorer* to create the system- and subsystem-level hierarchy or "skeleton" of the model (Figure B4). This is done by simply right-clicking on the "New Project" created earlier and selecting "New" Subsystem to add each subsystem to be modeled. Note: Parts or Components can also be added using this method, but these are best added automatically during the *Functional Block Diagram* creation so the *Pallette* or user library constructs for failure/flows can be leveraged and customized.
2) Create each *Functional Block Diagram* (FBD) and system hierarchy simultaneously by right clicking on the "New Project" and "New Subsystem" created earlier and opening a blank *System Model*.
3) Add *Components* or *Parts* to each FBD, using either the predefined components from the *Palette* or the MADE/user's *Library* (Figure B5), to populate the block diagrams with a drag-and-drop action, which will also autonomously populate the *Project Explorer* accordingly, or use a custom component/part (created via the Optional Step shown below) from the *Project Explorer*.
4) Custom *Component* or *Part* creation (Optional Step) is done by right clicking the system of interest (parent entry in the *Project Explorer* or *FBD* window) and manually adding a blank element to the model.

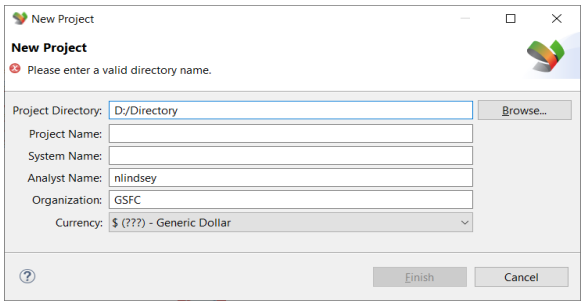

Figure B2: MADE New Project Window

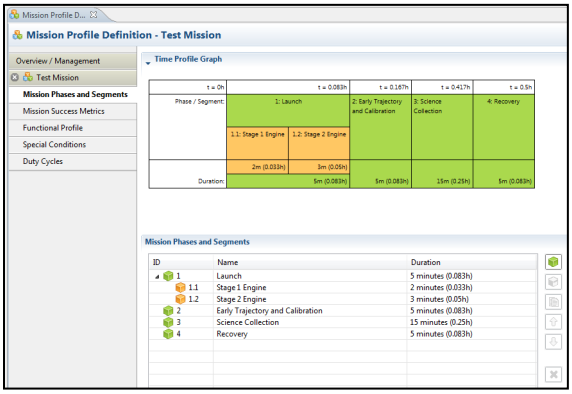

Figure B3: MADE Mission Profile Interface Example: The Sounding Rocket Mission Profile consists of a 30-minute mission with four main phases: Launch, Calibration, Science Collection, and Recovery. The Launch phase has two segments for Engine 1 and Engine 2 burns. The mission success criterion defined for this system is to "Exceed minimum mission success criteria requirement," and a duty cycle of 100% is considered for the nominal operation.

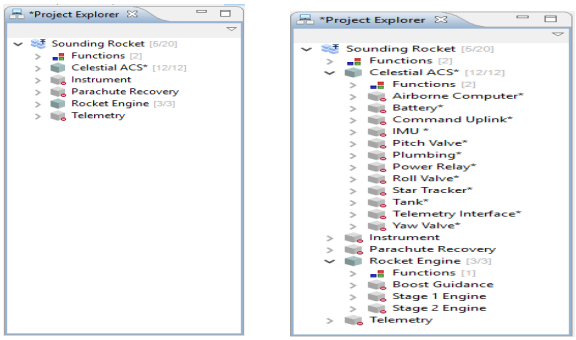

Figure B4: System Hierarchy Skeleton (Image on Left) and Completed Hierarchy with underlying components (Image on Right)

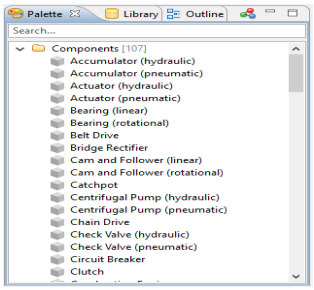

Figure B5: MADE Palette and Library Pane

1 Model-Based Engineering/Digital Engineering (MBE/DE) and Mission Assurance of Complex NASA Systems with Federated Modeling, Lindsey & Sindjui, 2025 (NTRS – 20240015898)

Appendix B : Recommended MADE Modeling Techniques & Tips

II. Refine/enhance functional block diagrams with functions, flows, supporting narratives, and documentation for each element

A. Add *Functions*, *In-Flows,* and *Out-Flows* to elements

1) Use the *Functions* option from the *Modeling* menu or right-click on the item being developed. This action will open the *Functions* and *Flows* window (Figure B6), and the modeler should select the appropriate functions for the item from that window's tabs (Figure B7).
2) Then, the appropriate *In-Flows* and *Out-Flows* should be connected by dragging a line from the in-flow to out-flow to visually define the functionality of that item, as shown in Figure B6.

After the *Functions* and *Flows* are defined for all the components, the FBD will automatically show input and output nodes in the model (Figure B8). These nodes are color-coded by type (Material, Energy, and Data), and the model should have an output or outputs associated with each input.

3) Use the curser to drag and connect the appropriate input/output FBD nodes to have a functionally interconnected model (Figure B8 and A9).
4) It is recommended the modeler pause at this point to adjudicate any errors (See Step V and Figure B61) in advance of further population. The errors reported can be connection errors within in-flows and out-flows or between part/component/subsystem blocks. Once errors are resolved, this is also a good time to validate the model with system designers and/or other discipline engineers to minimize modeling errors.

At this point the model is ready for mission assurance specific modeling, simulations, and requirement allocation/verification, before analyses, reports, and assessments are generated and reported. See each assurance discipline section for specific modeling guidance and recommendations, while maintaining routine multidisciplinary modeling team collaboration, to achieve overall modeling consistency and accuracy, so that accurate and value-needed results are produced.

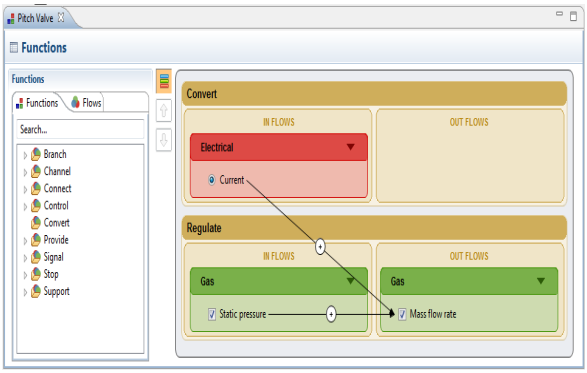

Figure B6: MADE Functional Definition Interface - For example, the Sounding Rocket Pitch Valve's functionality (Figure B7) was represented by the "Convert" and "Regulate" functions. Electrical Current and Gas Static Pressure were selected as inputs (in-flows), and correspondingly they will be processed and yield "Gas Mass Flow Rate" as the component final output (out-flow).

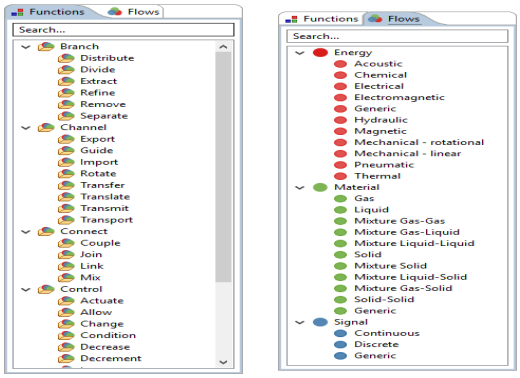

Figure B7: MADE Functions and Flows Tabs

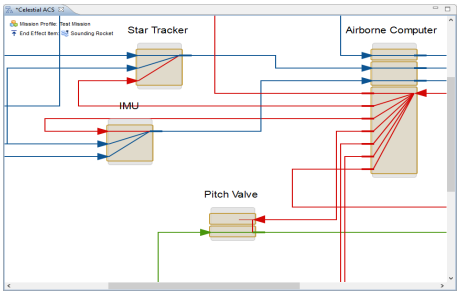

Figure B8: Model blocks input and output nodes

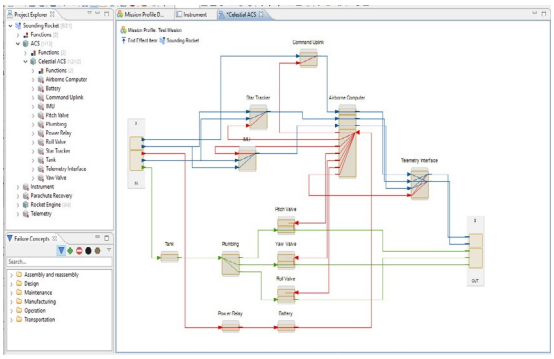

Figure B9: MADE System Hierarchy Interface and Functional Block Diagram Example: Sounding Rocket Celestial ACS

Appendix B : Recommended MADE Modeling Techniques & Tips

III. Add/Refine and integrate mission assurance discipline specific modeling

i. Reliability, Maintainability, and Availability (RMA) Engineering

A. Add failure characterization for FTA, FMECA, and Risk Priority Number (RPN) analysis via *failure diagrams*. Note: The purpose of a *failure diagram* is to visually represent the chain of corresponding *causes, mechanisms, faults*, and *failure conditions* for every *function* and outline possible *symptoms* for failure of that *function*.

1) Open the *failure diagram* by right-clicking on the modeling element in the *Project Explorer* or in its parent FBD and selecting "Failure Diagram" (Figure B10).
2) Populate (or refine) the *failure diagrams* (Figure B11) connected to each element by simply dragging and dropping failure causes, mechanisms, and faults/failure modes from the library of "Failure Concepts" (See Figure B12) into the *failure diagram*. Note: Those items/components that were previously selected from the *Palette* or MADE/User's *Library* already come with a preset *failure diagram* that can be customized to reflect project needs or conditions.
3) Add/Customize optional *Failure Conditions* (behaviors) for each failure mode. Then connect failure modes to the corresponding failure conditions and functions as applicable by dragging the selector from one to the other. Note: If System Safety modeling (see Step III.ii.B) has been completed, there may be linked *Failure Conditions* ( ) present in a failure diagram that will need to be connected to corresponding failure modes and functions.
4) Add/Customize optional *Symptoms* (indications or detectable manifestations) for each failure mode or failure condition. Then connect symptoms to the corresponding failure conditions, failure modes, and functions as applicable.
5) Add *Detectability, Severity, Occurrence,* and *Controllability* qualitative values (ratings) using the "Criticality and Reliability Editor" under the "Analyses" menu. The modeler should set these values for every function loss of an element using the rulers or use the value box (Figure B13) and the *Failure Mode Ratios* option that is used in FTAs only. A value from 1 to 10 should be selected for the high and low options. For example, a severity/criticality of 4 might be set if a board fails due to low voltage; but if it fails due to high voltage (overvoltage), the severity may be set to 7 or 8 because of the

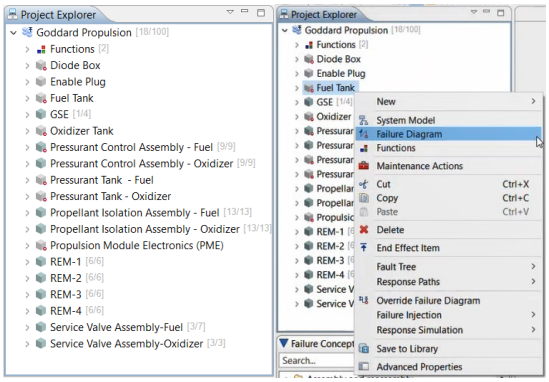

Figure B10: Project Explorer and Failure Diagram Navigation

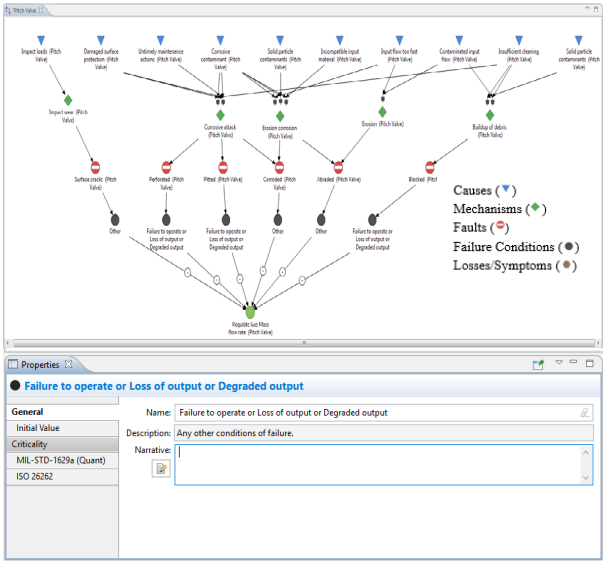

Figure B11: Example Failure Diagram and Properties Section – The failure diagram for the Europa Propulsion System "Fuel Tank", shows the chain of events for each cause (blue triangles), with various mechanisms (green diamonds), that would result in faulty operation (red circles with white lines), which would end up with possible failure conditions (black circles) that effect both the valve's "Store Liquid Volume" function (blue (signal), green (material), or red (energy) circles) and the symptoms of failure (brown hexagons).

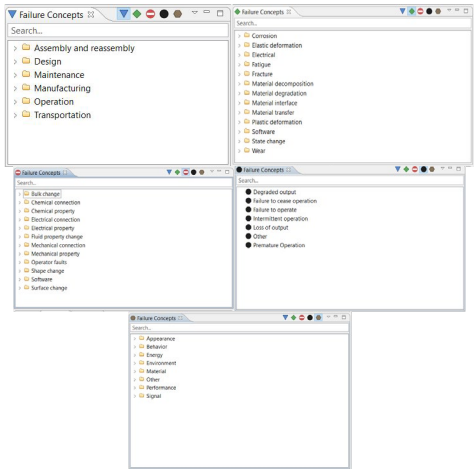

Figure B12: MADE Failure Concepts Tabs

possibility of other hazards, depending on system's sensitivity to the effect, as defined in the RPN Table or mission requirements (See Figure B13). Note: A modeler may want to run a FMECA/FTA report (Step VI.i.A) to fully understand potential propagations/effects prior to making a final *severity value* assessment to ensure accuracy.

6) Add *Severity Narratives* (optional) by editing the narrative properties of each functional loss (low and high) of an element. While optional, it is important that the modeler add natural language *narratives* as appropriate to each functional loss (low and high) to reflect consequences in mission terms for use in RMA reports (See Figure B13). Note: A modeler may want to run a FMECA/FTA report to fully understand potential propagations/effects prior to making final *severity narratives* to ensure accuracy.
7) Add *Compensating Provisions* and *Detection Methods* by right-clicking on each failure mode of interest in the *failure diagrams* and either selecting either the compensation or detection option to open the *Compensating Provisions & Detections Interface* window. When that window (Figure B14) is open, the modeler should select from the list or note a custom *narrative* for compensation and detection.

A *narrative box* is available for every *cause, mechanism, fault, failure condition,* and *symptom*. And the modelers should use these to make their models and outputs specific to the system's purpose.

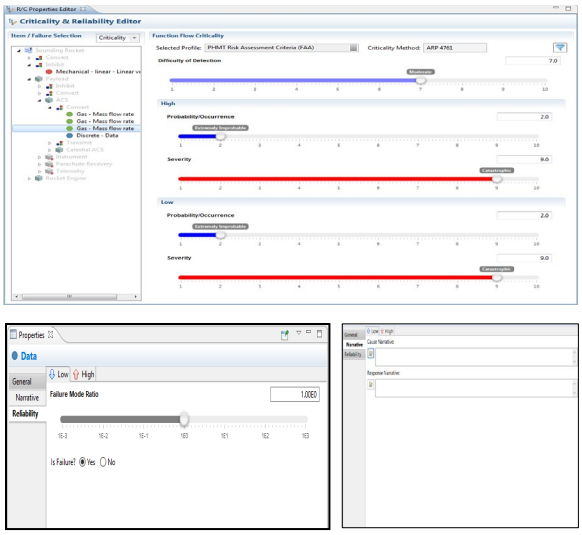

Figure B13: MADE Criticality and Reliability Editor & Properties Window

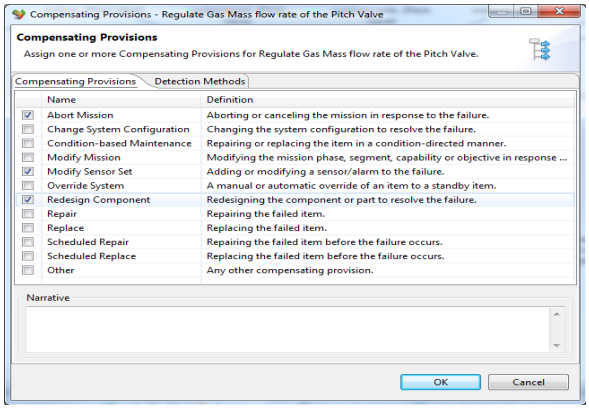

Figure B14: MADE Compensating Provisions & Detections Interface

B. Quantify reliability and availability modeling for Reliability Block Diagram (RBD), Availability Block Diagram (ABD), Reliability Centered Maintainability (RCM) analyses, and FTAs.

1) Add reliability quantification/distribution parameters to the properties of each element of the model as appropriate using the RMA module's *Reliability* interface (Figure B15). Currently, Exponential, Weibull, and Weibull (unit Life) data can be manually entered, or the "Failure Rate Prediction" option under the "Analyses" menu can be used. The latter requires naming of the analysis, selection of the item, and the addition of a complete parts list for the item/component being evaluated to the *Parts List section* of Figure B16. And when the calculation is completed, the modeler should use the "Apply Analysis MTTF to Item(s)" button so that the calculation is automatically applied to the properties of the item being evaluated (Figure B15).

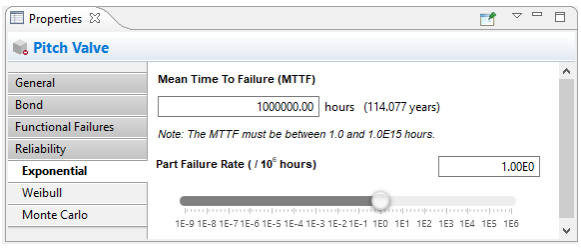

Figure B15: Item properties window for Failure Rate setting

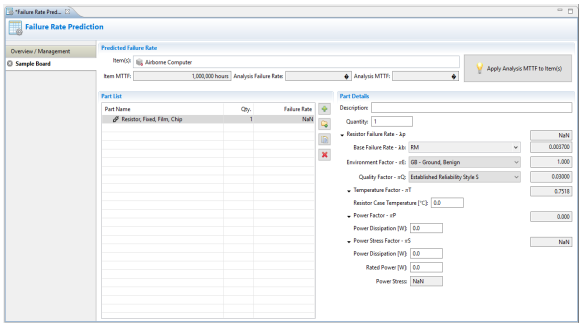

Figure B16: MADE Failure Rate Prediction Window

2) Define/Refine the *Reliability Block Diagrams* (RBDs) attached to each element. The blocks in each RBD are automatically generated along with the system hierarchy and should be organized in appropriate parallel, series, and/or standby groupings to accurately represent the system (Figure B17).
3) Add availability data to the properties of each element of the model as appropriate using the *Availability* option interface available when the RMA module is active (Figure B18). Currently, availability is calculated from MADE by MTTR (hrs.), delay time (hrs.), turnaround time (hrs.), and spares on hand (qty), and these are used in availability (e.g., Ao, Ai) and RCM calculations (i.e., Mean Administrative and Logistical Delay Time).
4) Define/Refine the *Availability Block Diagrams* (ABDs) attached to each element. The blocks in each ABD are automatically generated along with the system hierarchy and should be organized per the system layout (Figure B19) or the modeler's preference with appropriate parallel, series, and/or standby groupings to accurately represent the system.

C. Add Maintainability parameters to support Reliability Centered Maintenance (RCM) and technical life budgets for logistics, maintenance, and limited life analysis (LLA).

1) RCM entries – Will be included when testing of these features is more thoroughly preformed.
2) Add item life limits (estimated life budget) the parent subsystem or system has for the limited life item by selecting that subsystem or system and opening the *Technical Budget* tab (See Figure B19).
   a. Use the green plus sign to open the "Manage Technical Budgets" window.
   b. Use the green plus-sign to add or edit a budget by selecting the *budget type* from the *Quantity* drop-down and entering the budget limit in the *value* field (For example, 30000 shots, 8 lbs., and 92 Watts, as shown in Figure B20). Note: If the *budget type* does not exist, a *Custom Technical Budget* type can be added by using the "Custom Technical Budget" button (See Figure B20).
3) Once a budget is set, the current use or planned use of that budget can be added or edited via its *Technical Property* entries in the *Properties* window (Figure B21).
   a. Use the green plus sign to open the *Manage Technical Budgets* window.
   b. Use the green plus sign to add a *budget type* and enter the *value* in that field; or use the "pencil icon" to edit in the *value* field.

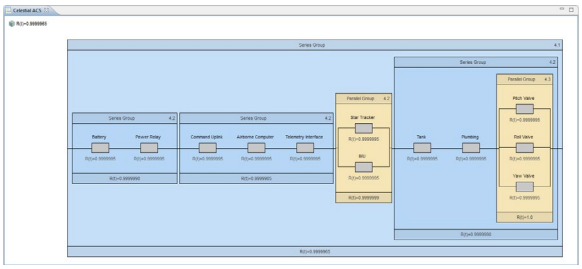

Figure B17: MADE Reliablity Block Diagram (RBD)

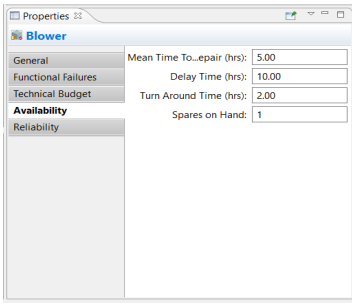

Figure B18: Item properties window for Availability setting

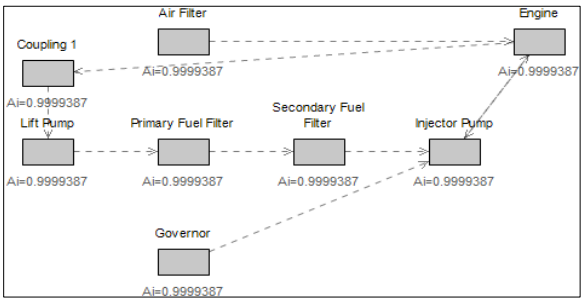

Figure B19: MADE Availablity Block Diagram (ABD)

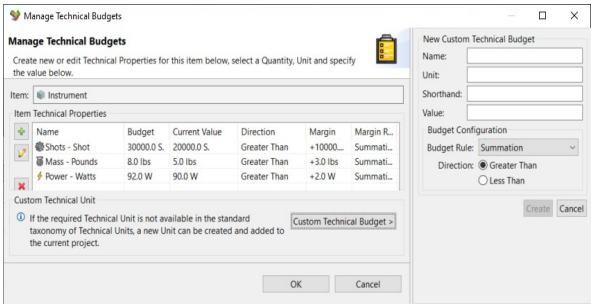

Figure B20: Establish budget in Manage Technical Budgets Window

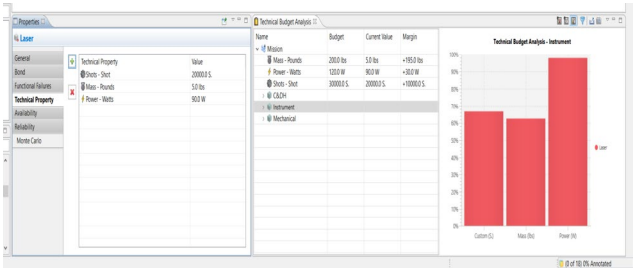

Figure B21: Establish budget in Manage Technical Budgets Window

4) Once budgets and use-values are entered, the margins can be viewed (tabularly or graphically, Figure B21) by using the "Technical Budget Analysis" option under the "Analyses" menu and clicking on the desired system, subsystem, or item.

At this point the model is ready for final error adjudication and validation with peers, system designers, other discipline engineers, and simulations (See Step V.B and V.C) to minimize modeling or assessment errors. After that, the model is ready to be used to generate the variety of RMA analyses and reports (Step VI.i.A-C) that can be used for requirement verifications and compliance assessments.

ii. System Safety Engineering

A. Add hazard characterizations for Hazard Analysis via the MADE *Functional Diagram* for the system. The hazards that need to be represented depend upon the design and Safety mission requirements and are therefore manually entered by the user until a hazard library or reference functional diagram is constructed.

1) Open the *Functional Diagram* by right-clicking on the system and selecting "Functional Diagram."
2) Add a *Function* for each NASA hazard category (Figure B22) by right-clicking on the diagram or using the "New Function" button.
3) Add the hazards (or hazard titles) to each category by right-clicking on the *Function* and opening the *Failure Conditions* window and adding a *Functional Failure* for each hazard. These are represented by a Venn Diagram Symbol as shown here: and in Figure B23.
4) Postulate the hazard causes for each hazard by adding a *Failure Condition* (represented by the ● symbol) for each by right-clicking on the *Functional Failure* (or hazard) or using the *Failure Conditions* window's left-side menu symbol ●. Note: Right-clicking on the window will add the *Failure Condition* to the last *Functional Failure* (or hazard) in the window.
5) Characterize each *Functional Failure* (or hazard) with a consequence (via the *Failure Effect* field, Figure B23). Note: A *Failure Condition Summary* for the highest severity *Failure Condition* (or hazard cause), Verification Summary, and Logical Links associated *Functional Failure* (or hazard) will be shown for reference.
6) Characterize each *Failure Condition* (or hazard cause) with a description of how the cause produces the hazard (via the *Narrative* field, Figure B24), consequence (via the *Failure*

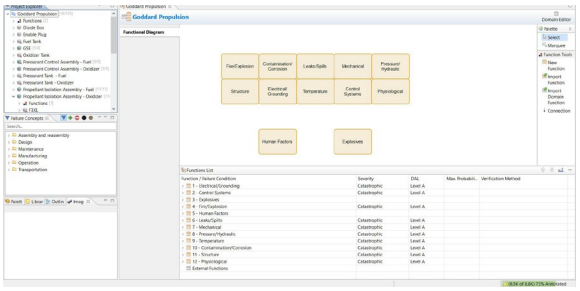

Figure B22: Example Functional Diagram of Twelve NASA Hazard Categories

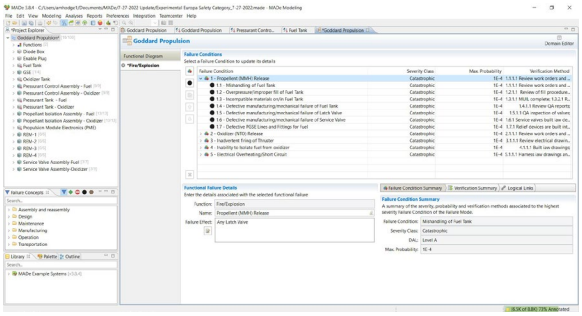

Figure B23: Example Failure Conditions

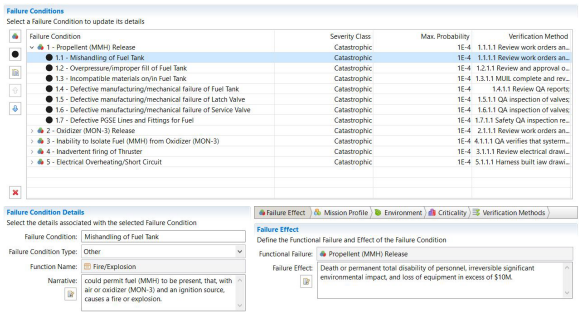

Figure B24: Example Hazard Cause Characterization Details
Note: Failure Condition Type should be set to "Other" since following these are hazard causes, not failure conditions.

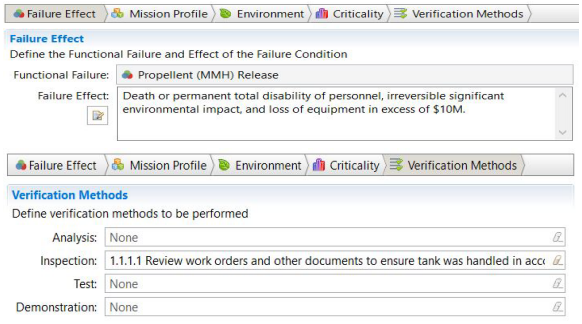

Figure B24: Example Failure Effect and Verifications

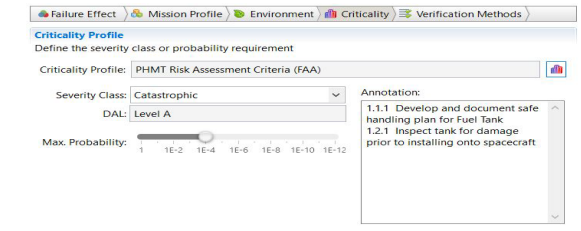

Figure B25: Example Criticality/Likelihood

Effect field, Figure B23), potential occurrence period (via the *Mission Profile* field), enabling environment (if any, via the *Environment & Emergency Configuration* fields), criticality/ limitations (via *Severity Class, DAL,* and *Max Probability* fields), controls (via the *Criticality Annotation* field), and verifications (via the method fields of *Analysis, Inspection, Test, and Demonstration*). See Figures A24-26.

Note: MADE uses each hazard cause's *Max Probability* (Figure B25) as a limit only and reports/displays the common/minimum value as the hazard's probability limit under its *Max Probability* field. This not the likelihood of the hazard cause that will be computed by the model or defined by other analysis or reference and applied to the *Hazard Tree*(s) by editing the event(s) associated with the cause.

7) It is recommended the modeler pause at this point to adjudicate any errors (See Step V.A and Figure B51) in advance of further population. Once errors are resolved, this is also a good time to validate the model with system designers and/or other discipline engineers to minimize modeling errors.

Note: The hazards and their characterization can be viewed or edited by right-clicking on the hazard category and selecting the "Failure Conditions" any time.

B. Integrate the hazard causes with each element's failure modeling, or *failure diagram* (optional for hazard analysis but recommended for overall risk assessment).

1) Begin linking each hazard category, aka *Function*, to its related item by right-clicking the "Functional Failure Hazard Category,*"* and opening *Functional Model Linking* window, (Figure B28) and selecting "Link to System Model."
2) Select the hazard category and item you want to link together and select "Create Link." Once a hazard category is linked, the hazard category will appear in the "Linked Function List" for each component and all the category's associated a hazard causes [*Linked Failure Condition* ( )] will appear in the item's *failure diagram*.
3) Connect (as described in Step III.i.A.3) applicable hazard causes or *Linked Failure Condition* to the corresponding *failure conditions* and functions, as applicable, in the item's *failure diagram* and delete any non-applicable or repeated hazard causes (Figure B29). Note: The same hazard cause may be present in multiple hazard categories, however only one hazard cause of the same name should be included in the *failure diagram*, but all applicable categories should be represented by at least one cause. The categories associated

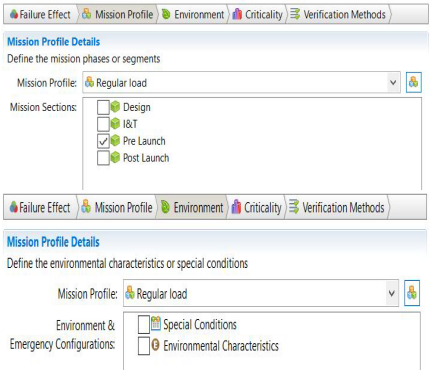

Figure B26: Example Mission Profile and Environment

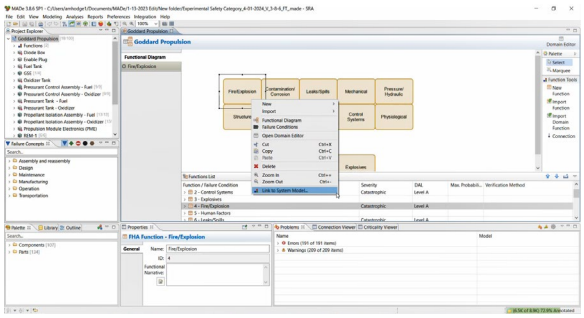

Figure B27: Linking Window

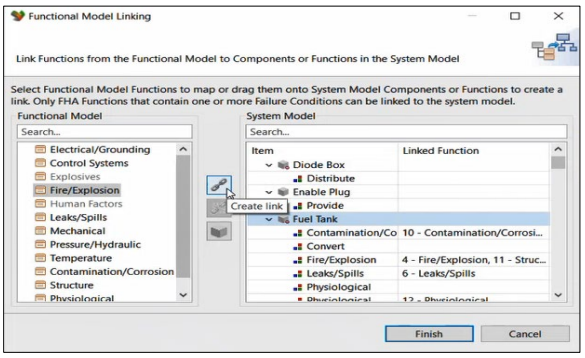

Figure B28: Linking Window

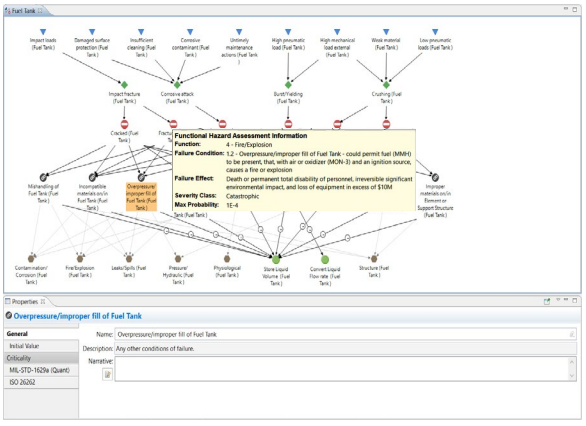

Figure B29: Example Linked Failure Diagram

with each cause can be viewed by hovering over the linked hazard causes [*Linked Failure Condition* ( )] and observing the resultant pop-up (Figure B29).

4) Add MADE *Symptoms* (represented by the symbol – Figure B29) to each *failure diagram* for each applicable linked hazard category and connect these to all the hazard causes, *Linked Failure Conditions,* that are associated with that hazard category, even if the *Linked Condition* is from another category. This is essential, since the HA Report needs to indicate all the hazard causes associated with each hazard category (even if their links were deleted as repeated *Conditions*).

Figure B30: Software Failure Diagram (TBD)

At this point, the model is ready for final error adjudication and validation with peers, system designers, other discipline engineers, and simulations (See Step V.B and V.C) to minimize modeling or assessment errors. After that, the model is ready to be used to generate the variety of System Safety analyses and reports (Step VI.ii.A-D) that can be used for requirement verifications and compliance assessments.

iii. Software Assurance – TBD

Figure B31-39: TBD for SWA

iv. Quality Engineering – TBD

Figure B40-49: TBD for QE

IV. Import requirements and link them to the model for compliance assessment, verification, and tracking

A. Import project or mission assurance requirements

1) Using the MADE add-on called the *Requirements Verification Tool (*RVT*)*, which allows the user to add requirements to the model.
   a. Outside the model, the desired requirements will need to be stored in a CSV/Excel format as shown in Figure B50.
   b. Once the requirements are prepared, the user will select "Requirements" under the Modeling menu to open the *Requirements Window* (Figure B51).
   c. Click the "Import Requirements" icon ( ) to open the *Requirement File Import Wizard* (Figure B51).
   d. Create/use a *Requirement Configuration* that corresponds to the requirements file and cross-references to the MADE properties (Figure B52) by selecting from the drop-down or clicking the "Add New" button to open the *Preferences* window. On the *Preferences* window, select the green plus-sign and enter a name, description (optional), worksheet name, number of header rows (to be ignored), and file-column to MADE property-mapping entries for each requirement field/column.
   e. Browse to the requirements file to be imported, select it, and click "Next," "Import all Requirement," and then the "Finish" button on the *Requirement File Import Wizard* to import without linking (recommended for easier re-use, but you can also link during importing (See Step IV.B.4)). Upon import completion, the requirements will display in the *RVT Requirements* window and requirement text will be in the *description* field (Figure B53).

2) Add initial NE (Not Evaluated) *Classifications* for non-SMA-related requirements or (I) for non-requirement information/titles. MADE learns and adds *Classifications* to the drop-down list as they are added.

Note: If a NASA-STD (such as the NASA-STD-8719.24 vol. 3, which is needed for System Safety) is ingested into MADE, it can be shared with other/future MADE models without re-ingesting from scratch by exporting the requirements at this point using the export button ( ). This will export all the requirements in the model to an Excel file. If there is a desire to import individual documents, then it is recommended that multiple ingest-export-delete requirements sequences be used to create individual exports. Then importing any of the exports to a new model will be done using a *Requirement Configuration* that is effectively a MADE-to-MADE correlation (that can be reused for any export) and only steps IV.A.1 c-e above.

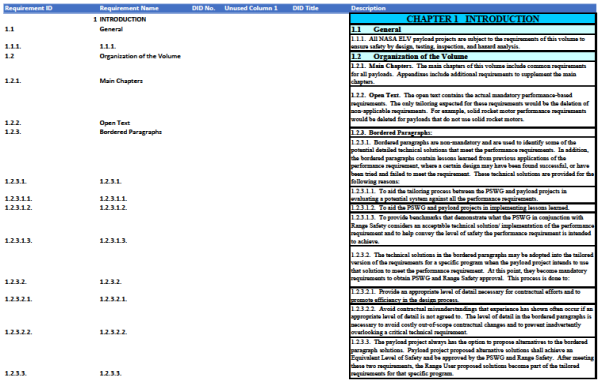

Figure B50: Example of Requirement Import Excel File: NASA-STD-8718.24 Vol 3 (double Click to open pdf)

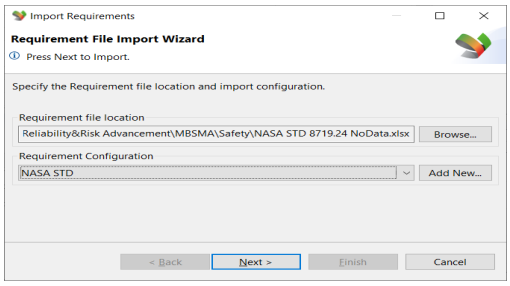

Figure B51: RVT Requirement File Import Wizard

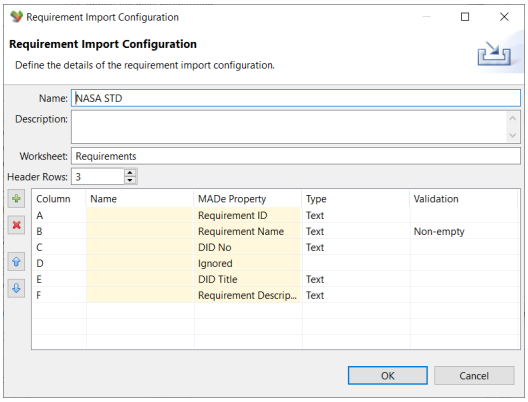

Figure B52: Requirement Import Configuration
Note: MADE properties for requirements are selectable from the drop-down for that field.

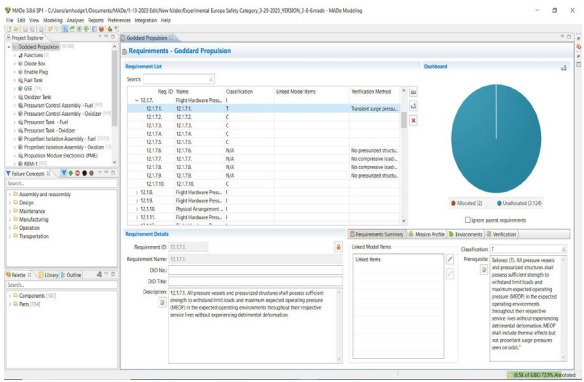

Figure B53: RVT Requirements Window

B. Characterize requirements for the system being modeled and Link them to modeling elements

1) Select the requirement in the *RVT Requirements* window (Figure B53) and Add/Edit the *Classification* of 'N/A' to those requirements that are Not Applicable to the system being modelled.
2) Select the applicability of each requirement by opening the *Mission Profile* (Figure B54) and *Environments* (Figure B55) tabs of the *Requirement Details* section of the *RVT Requirements* window and activating the appropriate checkboxes.
3) Enter the *Verification Method* planned for each requirement by opening the *Verifications* tab (Figure B56) of the *Requirement Details* section of the *RVT Requirements* window and entering text or using the drop-down option (viewable by pressing the space bar) to add one or more methods.
4) Link (or allocate) requirements to modeling elements by selecting the requirement in the *RVT Requirements* window, and (while viewing the *Requirements Summary* tab (Figure B53) of the *Requirement Details* section of the *RVT Requirements* window) use the link button ( ) to specify the items to link to the requirement by selecting them in the *Link Model Items* window (Figure 57). All linkages will be displayed in the *Requirement List* and the *Requirements Summary* tab (Figure B58) of the *Requirement Details* section of the *RVT Requirements* window but are only editable in the *Requirements Summary* tab. Currently, each requirement will need to be individually linked. Parent-requirement linkage does not currently automatically apply to child requirements and vice versa.

Note: Requirement updates for synchronization are currently accomplished by importing the revised requirements (steps IV.A 1-2), which replaces those already present (by matching identifiers) with selective updated imported field data (care should be taken to import only fields that need changes; other fields will remain as-is unless imported) and creates new requirement instances for any added requirement (since new requirements can only be added via importing at this time). Updates can also be accomplished manually but requirement deletions can only be done manually.

C. Annotate requirement with compliance/verification assessment/attachments and tailoring required

1) Select the requirement in the *RVT Requirements* window, and (while viewing the *Requirements Summary* tab of the *Requirement Details* section of the RVT window) use the *Classification* field (Figure B58) to update *Classifications* with compliance/verification assessments and annotate those updates as necessary using the annotation process shown

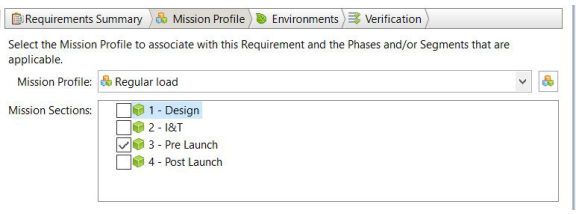

Figure B54: Example RVT Mission Profile Selection

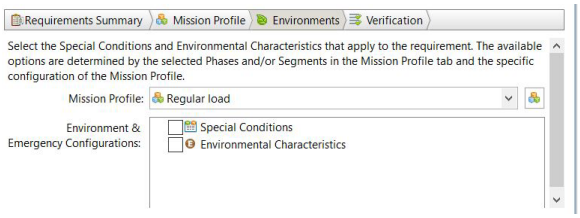

Figure B55: RVT Environments Category

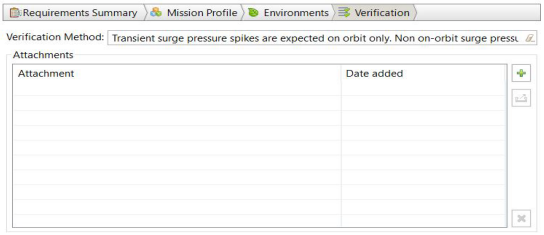

Figure B56: RVT Verification Input Tab

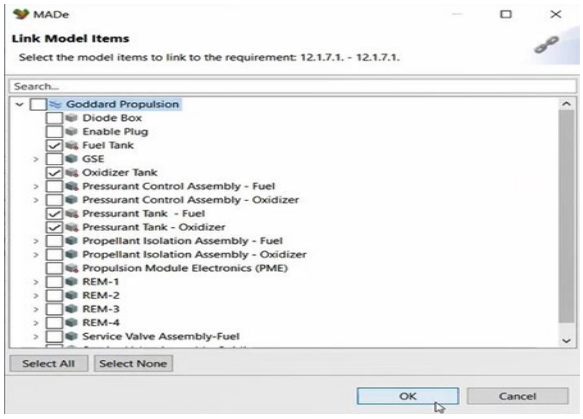

Figure B57: RVT Model Linking Window

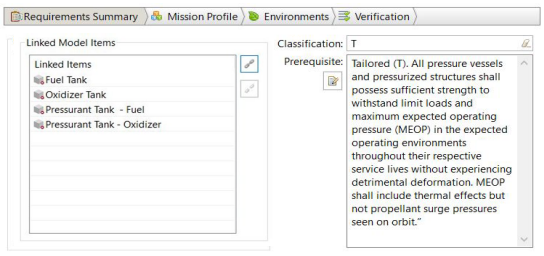

Figure B58: RVT Classification Window

Note that Requirement number 12.1.7.1 has been evaluated to show an example of a Tailored (T) Requirement.

in Step IV.C.2 (below). Note: This is a drop-down and free text field, so it is highly advised to use common terms/ definitions across a project consistently. For example, NASA uses 'C' for Compliant, 'T' for Tailored, 'NC' Not Compliant (or verification failure), 'V' for Verified, and 'UV' for Unverified (or verification pending/planned but not completed) or a combination (e.g., 'T/V', 'C/UV'); not the MADE provided categories: *Allocation, Derived, Design, Functional, Operational, Performance,* and *N/A.* MADE learns and adds *Classifications* to the drop-down list as they are added.

2) If the *Tailored* classification is used (as is needed for NASA System Safety), the revised text should be put in the *Prerequisite* field (Figure 58), and the tailoring rationale for the change should be put in the *Annotation* for the *Classification* change event by opening the "Annotations" editor under the "Modeling" menu and locating this change event, double clicking on it, entering the rationale in the narrative textbox, and clicking "OK" (Figure B59). The resulting rationales will be displayed in the *Annotations* window's table and can be edited/updated using the same process and accepted by editing and adding a comment ("Tailoring Rationale") and clicking "OK."

3) Attach verification-supporting artifacts using the green-plus sign button on the *Verifications* tab (Figures A56 and 60) of the *Requirement Details* section of the *RVT Requirements* window to select/attach the appropriate file(s).

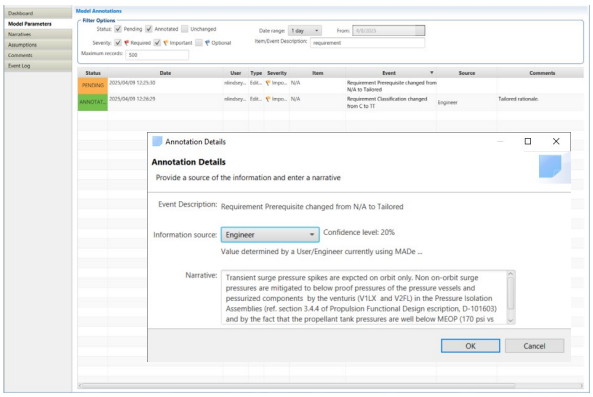

Figure B59: Annotations Window

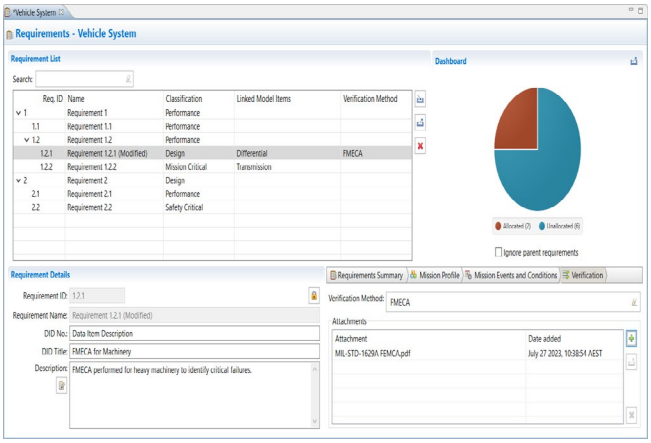

Figure B60: Requirements Verification Tab With Attachment

V. Verify and validate modeling with error adjudication, simulation, and peer consults

A. Adjudicate errors to ensure model is fully functional

1) In the instantaneous and live model *Problems* tab (Figure B61) a modeler can view the errors and warnings that need to be addressed. The problems reported can be connection errors within *in-flows* and *out-flows* or between part/ component/subsystem blocks.
2) Double-click on the error or warning to have MADE take you directly to the source of the issue.
3) Adjust the associated modeling appropriately to fully resolve issue and the error/warning will disappear. If the modeling change you make changes or partially addresses the issue, the error/warning will be removed and replaced by a new finding.

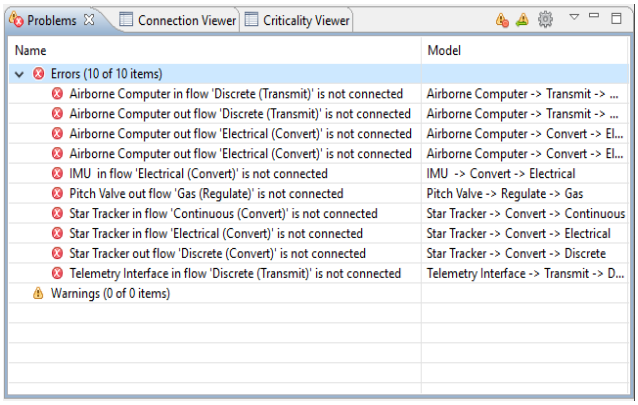

Figure B61: MADE Problem Reporting Window

For Example: the Sounding Rocket Problem Pane, Figure B10, shows that 10 nodes, including 5 out-flows and 5 in-flows are missing connections. This problem reporting section can also be used to verify if the right numbers and types of flows have been modeled during earlier modeling stages.

B. Verify failure propagation and fault tolerance (inhibit sufficiency verification) rationality and accuracy with simulations.

1) Create a *Fuzzy Cognitive Map (*FCM) to define and control the simulation of each failure/fault/ inhibit test case.

a. Select "FCM Threshold Preferences" from the *Preferences* menu to open the *FCM Analysis Preferences* window and set the *FMC Threshold Type*.

b. Select *Threshold Type* and enter *threshold* or *Min/Max value*s where applicable for the simulation. For example: "No Threshold" was selected in the Europa Propulsion due to complexity and varied outcomes that were expected (Figure B62).

Note: Thresholds control the expected behavior of the system in MADE: *Bivalent Threshold* is used when either a high or low state (scalable into values of 0 or 1) response is expected from only a low complexity system; *Trivalent Threshold* is used when three scalable states (low (-1), medium (0), and high (1)) are expected from a low complexity system; *Bivalent Sigmoid Threshold* is used for any responses expected for a smaller system that has a feedback connection; *Trivalent Sigmoid Threshold* is used for a smaller system that has physical response; and *No Threshold* is used for a larger complex system mode with any response state and with or without feedback connection.

c. Set/Adjust FCM Simulation parameters: *Perturbed Value, Perturbed Response Margin, and Step Limit* (optional) until the desired simulation completion is achieved. For example, on Europa Propulsion the maximum and minimum limit values were set to 10 and -10, respectively. This ensured that the response values calculated were not constrained by the threshold

2) Configure simulation test case by selecting the component(s) that the scenario will propagate from by right-clicking on each and selecting the *Failure Injection* option to choose the flow(s) that will be perturbed and its condition (e.g., up-arrow (sets *Perturb by value* to 1), down-arrow (sets *Perturb by value* to -1), and *Perturb by Value* for *No Threshold* FCM). The result will be that simulation will be enabled and the modeling space now mentions "Functional Perturbations" with the indication as to the type of perturbation(s) that have been configured the *Propagation Table* will appear at the bottom of the MADE window.

Note: In MADE the up and down arrows indicate that a component has changed its state, and the modeler can use either direction to indicate a particular state (e.g., on/off, open/closed, failed/operating, present/ absent, flowing/blocked). Therefore, it is recommended that the modeling team agree on a definition (e.g., NASA uses up for open or on and down for closed or off) to avoid potential confusion and garbled simulation-responses.

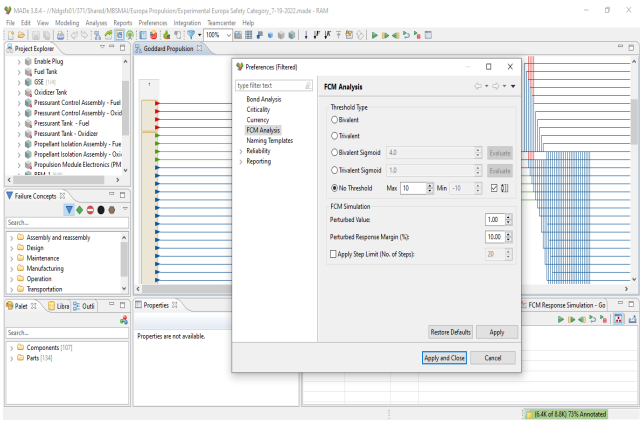

Figure B62: FCM Analysis Preferences

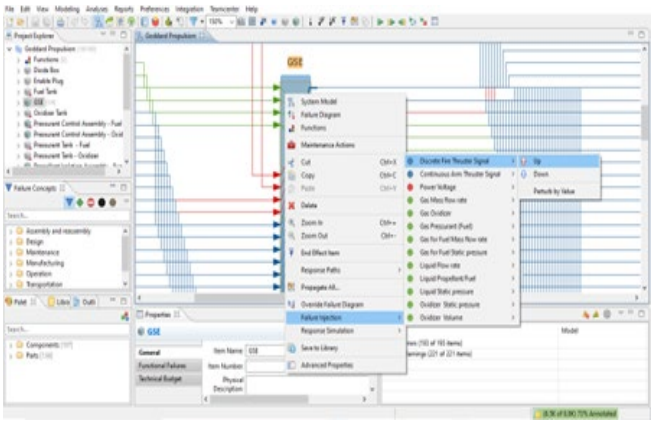

Figure B63: Failure Injection Configuration

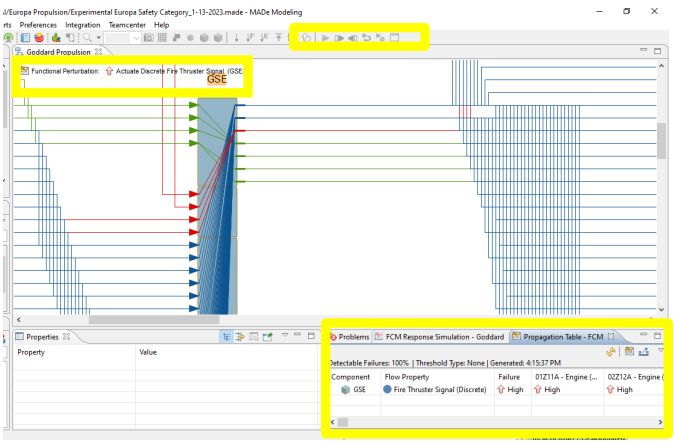

Figure B64: Functional Block Diagram Ready for Simulation

3) Execute the simulation by left-clicking on a *functional block diagram* space and then clicking the green "Play" button desired (for Step to Equilibrium (run to conclusion), Step forward one Step, or Step Backward one Step), located just above the modeling space in the MADE tool bar (Figure B64).
4) Watch (capture/record) simulation steps in the functional diagram space [Figure B65 - propagations (up, side-to-side, and down arrows) will appear, and the diagram may change as well until the next step or equilibrium is reached] and review results (Step V.C) to verify failure propagation is rational and fault tolerance/inhibits are sufficient.
5) If results are not rational, then adjust simulation parameters (Step V.B.1) to better match the complexity of the system or response possibilities but do not restrict impacts or propagations unnecessarily; or test case conditions (Step V.B.2), including changing the *Perturb by Value* to adjust the degree of the failure based on the range set for the FCM (e.g., if a valve is leaking but not open, the *Perturb by Value* could be adjusted from 1 to 0.5) in the *No Threshold* case; or *Polarity* to invert the effect of the injection at a downstream element (+/- on Figure B66); or lower the *Causal Strength* (Figure B66) on any receiving element's incoming flow (*In-flow* line) that should passively or nominally respond to the failure injection so that the propagation will go further in the system. For example, this would be appropriate to do on a material-flow or monitoring device that should pass on the anomalous condition but not be failed by the failure injection. See also MADE FCM user guide via help.

Note: Before executing any simulation, it is imperative that all the functional flows i.e.,, blue, red, green flow lines) leaving the selected component and continuing downstream are connected with no break in the chain of flow. An unexpected break in the chain of flow will prematurely stop the simulation.

C. Validate model accuracy by reviewing simulation results and mission assurance reports (Step VI) independently and with system designers and/or other discipline engineers.

1) Review simulation results in the system's *functional block diagrams*,
2) Review all perturbations to the system from each simulation(s) in the *step table* (Figure B67). The *step table* displays each successive response of system element until equilibrium is reached.
3) Review the effects of simulation(s) on each element of the system in the *propagation*

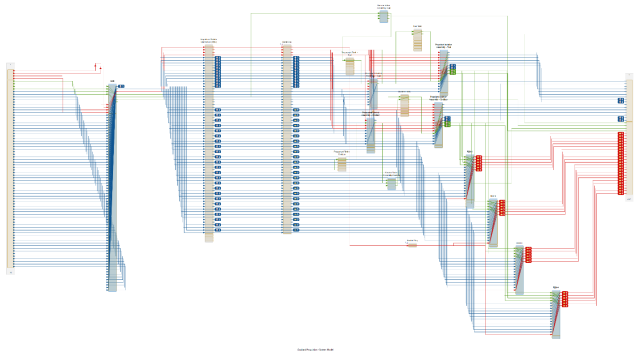

Figure B65: System/Simulation Presentation

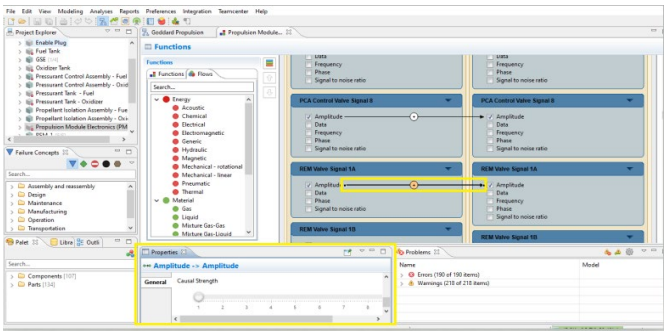

Figure B66

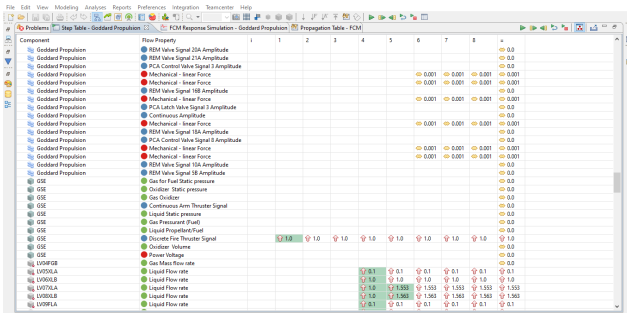

Figure B67: Simulation Step Table

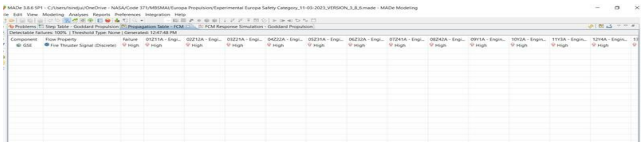

Figure B68: Propagation Table (partial view)

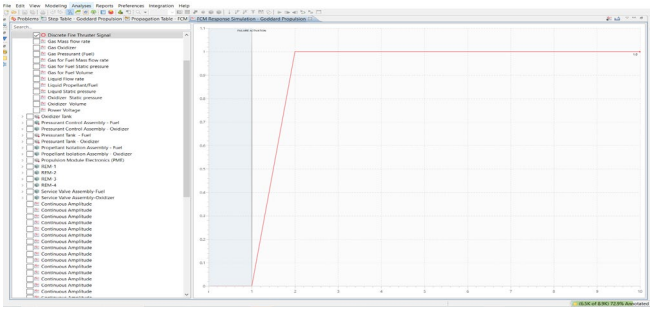

Figure B69: FCM Response Graph (for discrete fire)

table (Figure B68). This table shows the resultant effects of each item in a system and their effects on corresponding items in the system based on their functional connection.

4) Review simulation results using the *FCM response* tab to see a graphical representation (Figure B69) of the *step table.*

5) Review/Discuss mission assurance reports (Step VI) to validate modeling accuracy and results.

VI. Generate mission assurance reports/assessments and requirement verifications/compliances

In MADE there are several report categories (i.e., Criticality, Data Quality Analysis, Maintainability, Mission Profile Definition, Reliability & Availability, Reliability-Centered Maintenance (RCM), and Prognostics and Health Management) that allow the user to report on different analyses conducted in many formats and output types (including PDF and Excel - which is recommended for Federation[2] and sharing). More information about each report is displayed in the *Report Wizard* window *description field* (Figure B70).

i. Reliability Report(s)

A. *FMEA/FMECA:* Use the *Report Wizard* to select the desired failure-modes report format, under the "Criticality" category, and output type. For example, NASA/GSFC uses the "FMECA (MIL-STD-1629A, Custom)" option to generate their desired report (Figure B71).

B. RBD/Predictions: Use the *Report Wizard* to select the "Reliability Block Diagram" report format, under the "Reliability and Availability" category, and output type. This report (Figure B72) includes failure rates, MTTR, grouping logic, and reliability/availability prediction information.

C. FTA/FTs: The Fault Tree Analysis (FTA) report in MADE requires that an FTA (either hardware, functional, hazard, or custom-event-related) be previously built (Steps VI.C.1-4 below). FTA reports can then be generated using the "Fault Tree Analysis (FTA) Report" format, under the "Safety" category in the *Report Wizard* and selecting a previously built FTA; or selecting "Fault Tree" underneath the "Analyses" main menu and viewing a previously built FTA and taking screenshots of the analysis of interest to use in other documents/analyses or in a user-created FTA report.

1) To build an FTA, enable the *Safety and Risk Assessment module* by clicking that module's button (Figure B63); otherwise, the *Fault Tree Builder* will not be accessible.

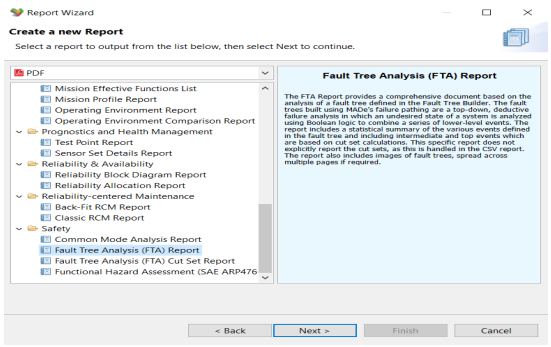

Figure B70: Report Wizard

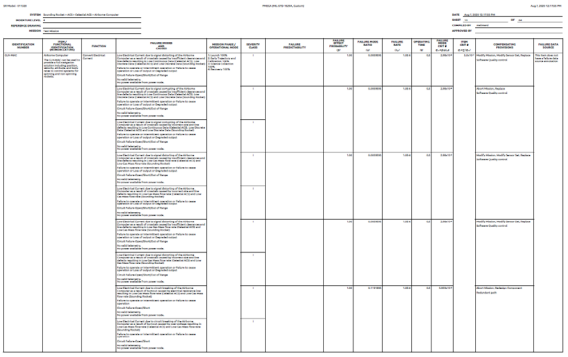

Figure B71: Sample FMECA Output

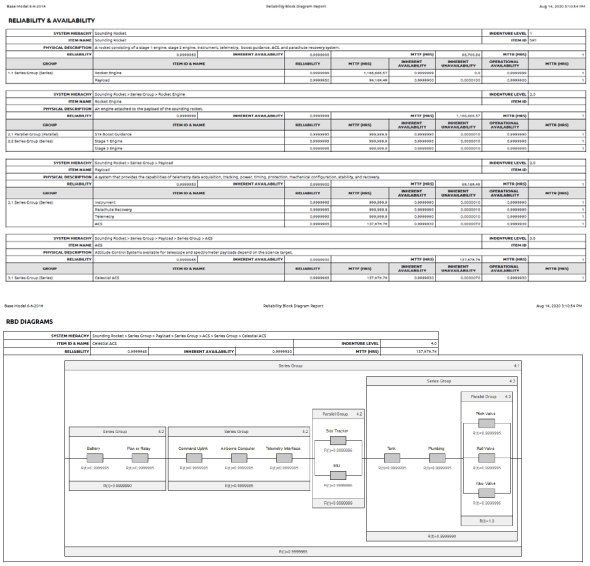

Figure B72: Sample RBD Output

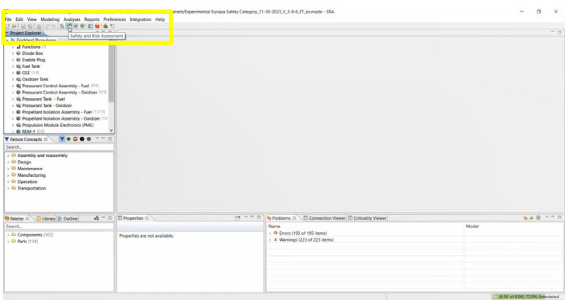

Figure B73: Safety and Risk Assessment Module

2 Model-Based Engineering/Digital Engineering (MBE/DE) and Mission Assurance of Complex NASA Systems with Federated Modeling, Lindsey & Sindjui, 2025 (NTRS – 20240015898

2) Open *Fault Tree Builder* from the "Analyses – Fault Tree" menu (Figure B74) and create an FTA instance by clicking the green plus-sign and entering a *Name* (editable later), *Description* (optional - editable later), *Fault Tree Type*, and *Top Event*.

Note: Model-Based/Default *Top Event* availability varies depending on FT-type selected. For Hardware FTs, the top events are limited to system elements (hardware or software); for User-Defined FTs, the top events can be system elements, *failure conditions* (these are used for hazard cause trees (see Step VI.ii.D)), *symptoms,* or *custom* defined*;* and for Model-based Analysis FTs, top events can be system elements or *symptoms* (these are used for hazard trees (see Step VI.ii.D).

3) Specify *Propagation Type, Depth of Analyses, Level of Analyses, and Preferred Gate Type* if "Model-based Analysis" is selected.

4) Click OK to save and generate the FT. And MADE will return FTs, as shown in Figures A65 - A67, which can be captured (using the camera button) and saved/printed.

After any FTA is built, it can be captured or output in an FTA report at any time as a PDF or an Excel spreadsheet (recommended for Federation[3] and sharing).

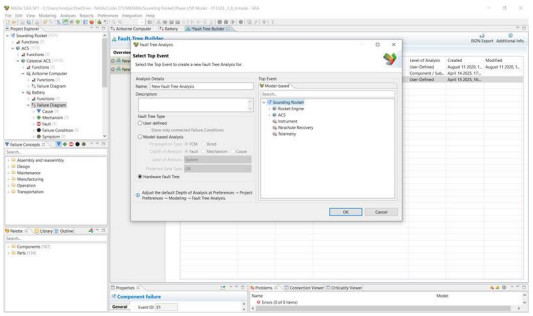

Figure B74: Fault Tree Builder

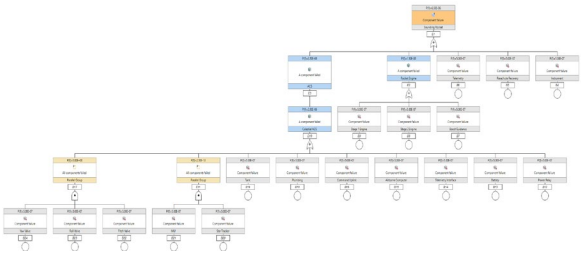

Figure B75: Sample Hardware Fault Tree Output

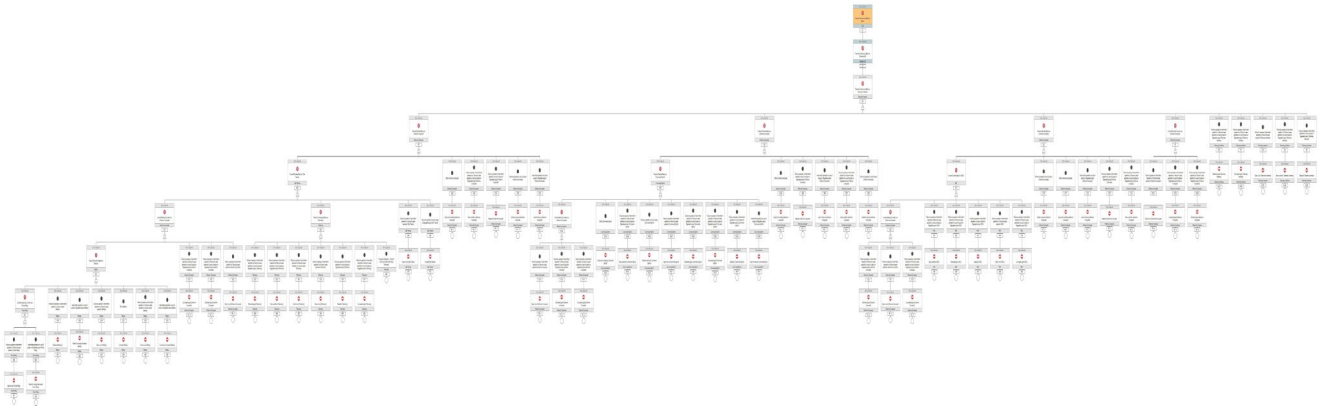

Figure B66: Sample Functional Fault Tree Output (embedded pdf) –
This FT reports the failure of a specific function and may be used to assist with scenario-based PRA analyses.

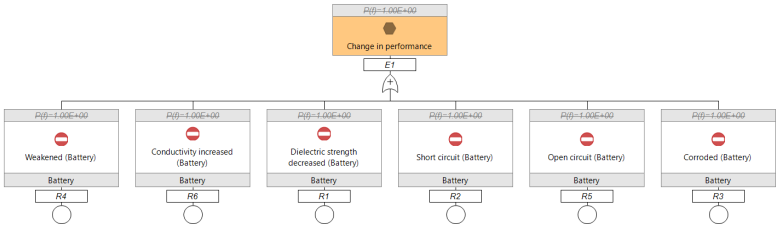

Figure B67: Sample Symptom Fault Tree Output –
This FT can report a specific event and may be used to assist with scenario-based PRA analyses.

3 Model-Based Engineering/Digital Engineering (MBE/DE) and Mission Assurance of Complex NASA Systems with Federated Modeling, Lindsey & Sindjui, 2025 (NTRS – 20240015898)

ii. Safety Report(s)

A) Hazard Assessment Reports (HAs/OHAs/OHSAs): Use the "Report Wizard" to select the "Functional Hazard Assessment (SAE ARP4761)" report format, under the "Safety" category and select the system for the Functional Hazard Assessment (FHA) and *Mission Profile*, *Duration*, and *Phases & Segments*. The resultant PDF or Excel report (Figure B68) includes in a tabular format the Hazard Category (*Function & Function Name*), Hazard (*Functional Failure*), Hazard Cause (*Failure Condition*), Hazard Description (*Narrative*), Severity Consequence (*Failure Effect*), Severity (*Severity Classification*), Probability of Occurrence (*Probability Objective*), Controls (*Supporting Material*), and Verifications (*Verification Methods*).

B) Safety Requirements Compliance Checklist (SRCC): Open the MADE add-on called the *Requirements Verification Tool (*RVT*)* by clicking *Requirements* under the *Modeling* menu and use the export requirement button ( ) to generate a draft SRCC in Excel. Note: requirement tailoring annotations are not yet included, but the tailored text will be shown.

C) Safety Data Package (SDP): To create an SDP from a MADE model, combine three documents: the *Hazard Analysis Report* (Step VI.ii.A), MADE System Information Report (by using the *Report Wizard* and selecting *System Information Report* (SIR) under the *General* category), and a Ground Operations Plan created outside of MADE (but can be attached to the model). Note: The SIR does not include any modeling attachments (such as system images), but the SDP must include images of the system. So, significant exports from the model and post-processing of these will be needed to produce a complete SDP.

D) Hazard Trees (HTs): An HT report in MADE requires that an FT be built using the same methods as Model-based Analysis FTs (Steps VI.C.1-4) but with only a hazard symptom as the top-level event (Figure B71) and setting *Propagation* set to *FCM*, *Analysis Depth* to *Cause* (to ensure mechanisms are included), and *Preferred Gate Type* to *OR* (or other model specific option - *AND* or *K of N*). After any HT is built (Figure B72), it can be edited to refine/define non-model-derived hazard-cause likelihood or captured/output in an FTA report at any time as a PDF or an Excel spreadsheet output (recommended for Federation[4] and sharing).

iii. Software Assurance – TBD

iv. Quality Engineering – TBD

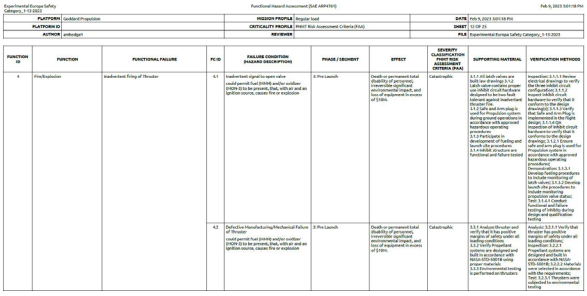

Figure B68: Sample of Hazard Assessment (HA) Report for Fire/Explosion

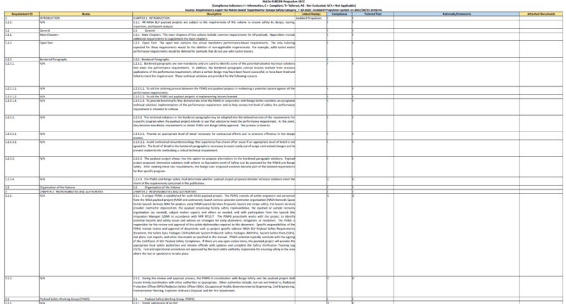

Figure B69: Sample of Safety Requirements Compliance Checklst with titles post-processed

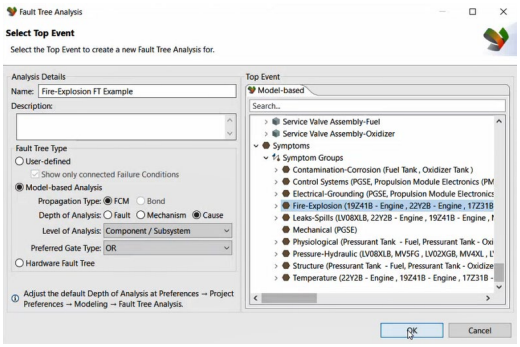

Figure B70: Hazard Tree Top-Event Selection

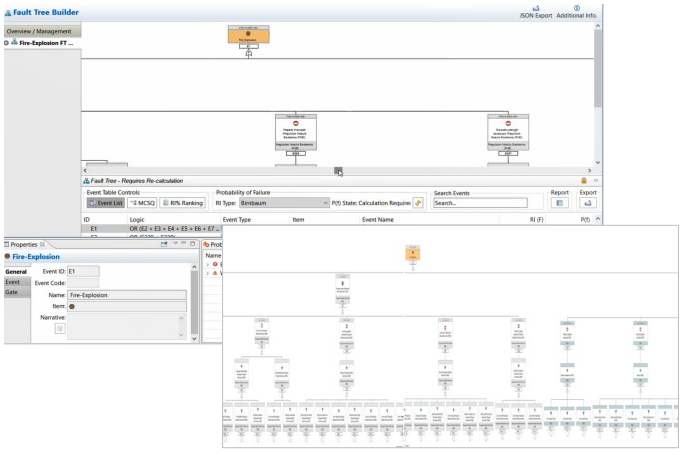

Figure B71: Hazard Tree Example (all hazard cause proabailities are model-based in this instance)