MODEL BASED FUNCTIONAL SAFETY FOR COMPLEX SOFTWARE INTENSIVE SYSTEMS

BARRY HENDRIX
APT RESEARCH, INC.

37TH ISSC
2019

SAFETY ENGINEERING
SEAC
& ANALYSIS CENTER

Safety Engineering and Analysis Center
A Division of A-P-T Research, Inc.
4950 Research Drive, Huntsville, AL 35805
256.327.3373 | www.apt-research.com

NDIA and INCOSE Visions

- “Model-Based Engineering (MBE): An approach to engineering that uses models as an integral part of the technical baseline that includes the requirements, analysis, design, implementation, and verification of a capability, system, and/or product throughout the acquisition life cycle.”

  Final Report, Model-Based Engineering Subcommittee, NDIA, Feb. 2011

“Model-based systems engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases.”

INCOSE SE Vision 2020 (INCOSE-TP-2004-004-02, Sep 2007)

Model Based - Modern Systems Mandate

- DoD 2018 Digital Engineering Strategy signed by Michael D, Griffin Under Sec of Engineering focuses on incorporating 5 modern pillars.
  - 1. Formalize Development Integration and Use of Modeling for Capability, Use Functionality and Behavior
  - 2. Authoritative Source of Truth - Technology, Facts, Objective Evidence
  - 3. Incorporate Technology Innovation
  - 4. Establish Infrastructure and Environments
  - 5. Transform Culture and Workforce

Evolving DoD Model Based Processes focused on Safety

- Architecture Central Virtual Integration Process (ACVIP) based on SAE International Aerospace Standard AS5506C Architecture Analysis and Design Safety Language (AADL).

- SAE1005 Model Based Functional Safety for Complex Software Intensive Systems - Draft Guidelines 2019 (under G-48 Review)

- Model Based methods and techniques along with AGILE software development and other modern and evolving methods, such as ACVIP, have been mandated by DoD, NASA, and the FAA.
- The newly created Army Futures Command in Austin, TX and Academia - Carnegie-Mellon University (CMU) Software Engineering Institute (SEI) and a consortium of the largest aerospace companies are researching and developing modern methods that are key to meeting the DoD Engineering Strategy. Models are central to the mandates. Safety processes with models must be developed to keep up with the new systems engineering models.

Model-Based System Engineering

- MBSE started in early 1990s and Model-Based Development (MBD) became popular in mid-1990s
- MBSE has been endorsed by International Council of System Engineering (INCOSE) since 2005.
- Most of the model-based paradigms are in complex and software-intensive systems development – many are safety-critical
- "Model-based systems engineering (MBSE) is the formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases." INCOSE SE Vision 2020 (INCOSE-TP-2004-004-02, Sep 2007)
- DoD 2018 Digital Engineering Strategy promulgates and mandates

System Model

- System model – A structured representation that focuses on the overall system requirements, and depicts behavior, structure, properties, and interconnections, including the many software capabilities and functionality.
- A safety-critical and software intensive system can be depicted better with models than by narratives and English prose.
- System decomposition and behavior can be part of the model
  - Models can determine what must be done to meet the requirements
  - Functional behavior and activity models in software can be depicted
  - State and mode models can show behavioral differences
  - Safety-critical functional flow can be depicted in a variety of ways making it easier to interpret vs. levels of abstraction

Model Centric

- The Past: in English prose documents with complete sentences, “shall” statements, lots of pages.
  - Specifications
  - Interface requirements
  - System design
  - System verification documents

- Current and future: Model Centric
  - Modern software tools
  - Established modeling concepts and standards
  - Modeling languages
  - Better depiction of system functionality and behavior

Models - Safety Advantages

- System modeling (software intensive) can address:
  - Systems integration
  - System requirements
  - Functionality depicted
    - Functional models
    - Behavioral model

- Safety models
  - Safety-critical functions (SCF)
  - Safety-significant hazards
  - SCF sequences and functional flow diagrams
  - Safety features, safety devices depicted
  - Safety functional threads, logic flow
  - Safety performance (intended behavior verified in models)
  - Special safety tests in models (to prove correct behavior)

Functional Safety Benefits

- Functional Hazard Analysis (FHA) can be depicted beyond just templates
- Safety Behavioral diagrams can elaborate on safety attributes
- Functional Flow Diagrams can show safety aspects
- Systems integration and SOS can depict safety aspects
- Explicit System and Safety Requirements can be part of the model
- Functionality Depicted in Functional Models and Behavioral Model
- Safety Models with Safety-critical functions (SCF)
- Safety-significant hazards can be depicted clearly
- SCF sequences and functional flow diagrams can depict SCFs
- Safety Features, Safety devices depicted
- Safety functional decomposition, functional threads, logic flow
- Safety performance models can show intended behavior and verified in models
- Special Safety Tests in Models can help to prove correct behavior
- Complex Artificial Intelligence (AI) functions can be depicted

Some Large Model Based Programs

- System safety and software safety are highly involved in many modern and complex systems that use various model-based engineering processes, methods, and tools.
  - B-787 and newer aircraft use DO-331 Model-Based Development and Verification Supplement to DO-178C
  - FAA Ground Station Software uses DO-278A with DO-331 Supplement
  - One DoD example: Joint Services F-35 Vehicle Management Computers (VMC) since 2001.
  - Another DoD example: U.S. Army Multi-Mission launcher for IFPC currently using model-based software in recent years.
  - 2018 Digital Strategy Number One Objective is Using Models in Engineering

Concepts and Precepts of Functional Safety

- Two Domains exists in a System
  - Physical
  - Functional
- Two Domains for Safety
  - Physical (hardware)
  - Functional (more software involvement in behavior)
- Physical can be described in hardware, length, weight, type material
- Functional is how a system behaves or what it does (functionality)
- A system hardware configuration is accepted using a Physical Configuration Audit (PCA)
- A system software configuration is part of a Functional Configuration Audit

Modern View

- A Modern Software Intensive System must be viewed Functionally - not from a pure hardware or software perspective. Rather as a highly integrated system
  - The Physical System
  - The Software System
  - The Computer System
    - Functionality Provided (video, communications, command and control, sensor data, movement of surfaces, fire control solutions, autonomous control /robotics, precision weapons firing, much more)
    - System-of -systems (SOS) interoperability may involve highly integrated and complex safety critical multi-functionality.
      - SCFs must be identified early through SE functional analyses (FHAs, Software Safety Analyses)

Functional Safety Modeling Principles

- Functional Safety Models can help precisely document Objective Safety Evidence
  - SCFs in the software instruction set
  - Safety models of the complex system architecture,
  - Functions, behavior, credible failure and faults
  - FHAs focus on failure conditions that cause the hazards
  - Functional causal factors that affect safety behavior are identified
  - How a system reacts (safely or hazardously) to situations and conditions
    - Knowing helps safety analysts fill in the uncertainty gaps and escapes of the undocumented safety details of the system behavior.

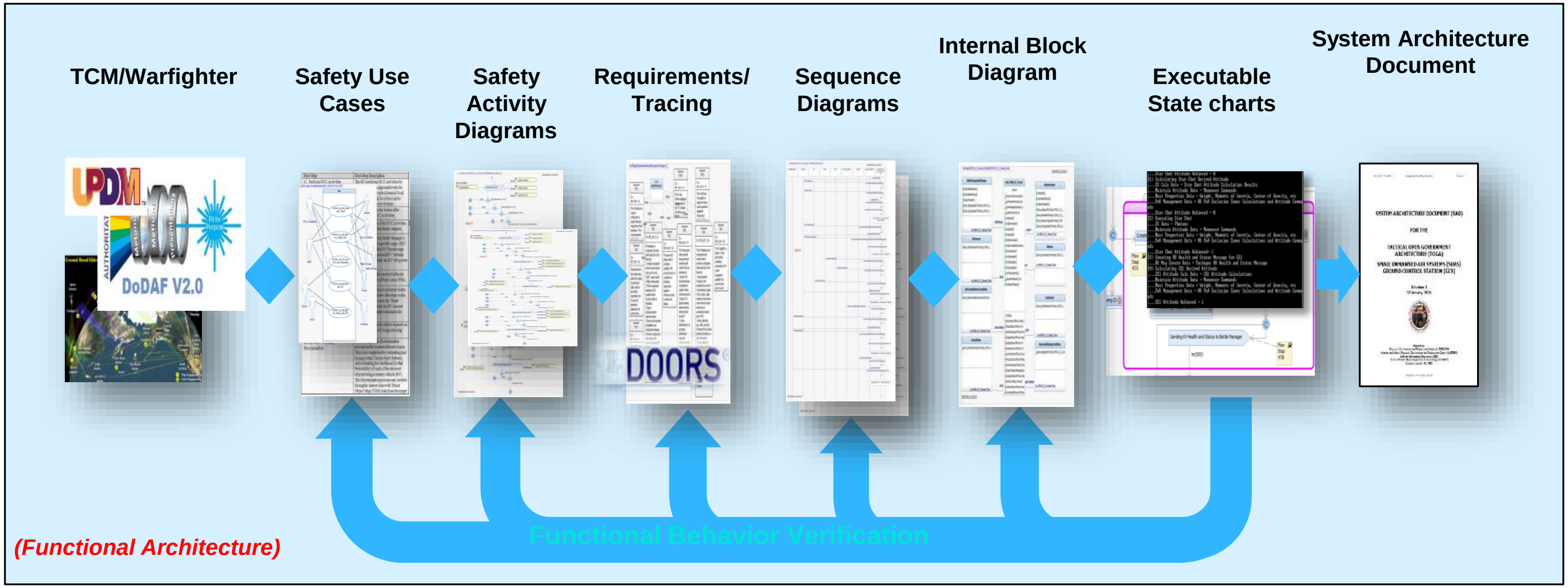
MBSE and Software System Safety

- INCOSE, DoD, and NASA endorse MBSE as the norm on some complex and safety-critical systems
- DoD acquisition in place for models with software intensive with complex software functionality
- MBSE has engineering development advantages for software-intense integrated systems
- MBSE used for interoperability and optimization to enhance performance in future battlefield scenarios.
- Model-based development has emerged over the past two decade as one solid solution proven to dovetail well with software engineering/software system safety goals and objectives

MBSE Output Useful for a SAR or Safety Case

Example of MBSE in place today by Army Aviation and Missile Research, Development and Engineering Center (AMRDEC) Software Engineering Directorate CoMBAT Team Process

DoDAF MBSE framework allows better behavioral allocations to be handed off to software developers (and software safety). Software safety can have inclusive inputs at use case (beginning of the process).



TCM/Warfighter

Safety Use Cases

Safety Activity Diagrams

Requirements/ Tracing

Sequence Diagrams

Internal Block Diagram

Executable State charts

System Architecture Document

Functional Behavior Verification

(Functional Architecture)

Functional Safety Modeling Candidates

- The following can be modeled for better program safety documentation:
  - Identify, Analyze, Reduce, and Accept (IARA) safety process per ANSI/GEIA-STD-0010-2009.
  - Functional Hazard Analyses per MIL-STD-882E Task 208. FHA, Use Cases, SCF, Functional behavior can be modeled using safety activity diagrams
  - Functional Hazard Assessments per SAE ARP4761. FHA, Use Cases, SCF, Functional behavior can be modeled using safety activity diagrams
  - Functional Design Assurance Level (F-DAL) per SAE ARP4754A.
  - Software Safety Analyses (6) per IEEE Std 1228-1994
  - Preliminary System Safety Assessment (PSSA) per SAE ARP4761. May contain preliminary models.
  - System Safety Assessment (SSA) per SAE ARPs. May document final safety models.
  - Safety Assessment Report (SAR) per MIL-STD-882E. May document final safety models as objective evidence.
  - Safety Case per United Kingdom Ministry of Defense (MOD) Standards and other industries. To support claims and to refute arguments

Model Based Functional Safety Guidance (1)

- Determine what the system and software must do to meet the safety requirements
- Identify the transformations of safety-significant inputs to outputs (functional/activity models)
- Model the possible credible behavioral differences (state/modes/situations/conditions models)
- Model the incoming requests for services (safety-critical message)
- Safety integration models (system-of-systems) should graphically depict the entire integrated system model, including interoperable systems and subsystems, from the system architecture, redundancies, safeguards, and safety features.
- High granularity safety models may also show top-level events and/or top-level mishaps and how the system prevents, eliminates, or controls the functions from failure conditions, severe hazards, hazardous behavior, unintended consequences, and risks.
- Depict a model of how the system architecture and functions achieve the required safe behavior under the given constraints. This could be showing parallel, independent and redundant systems, forms of autonomy, engineered safety features and other physical or functional hazard mitigation.
- Model any hazard or combination of failure conditions and resultant hazard and precisely how they are reduced, mitigated, closed and accepted as a manageable risk.
- Integrating existing hazard analysis methods, techniques, such as FHAs, fault trees, event trees, software safety analyses as inputs to the functional safety model showing as much of the safety behavior as possible.
- Functional modeling of software safety verification activities should ensure SCF and software behave as intended and produces the expected outcome

Model Based Functional Safety Guidance (2)

- Functional Models can help ensure SCFs do not allow undocumented, undefined, and unintended actions or consequences
- Functional Models can Functional Models of SCFs can help ensure functions and behavior perform at the intended time allocation and within its defined sequence
- help ensure SCFs behave as expected in normal/nominal scenarios, situations, and conditions
- Models can help ensure SCFs and all functions within software behave as predicted, specified as expected in off-nominal fault/failure conditions with the innovative use of fault insertion techniques of failure modes effect testing (FMETS), failure immunity testing (FIT).
- Modeling Functional Threads Analysis is tracing the safety-significant functional threads from the SRS to each CSCI to the CSC to the CSU. In some languages this could be class or type for Java language - to the software code or test script. MBFS should model complete functional threads traceability.
- Structural coverage, decision coverage, path coverage and object code branch testing can be modeled to show meeting software quality assurance and design assurance objectives of DO-178C with level A and B contributing to safety-critical software.
- Functional Design Assurance Levels (FDAL) can be modeled to show compliance with SAE ARP 4754A.
- Functions within software should have requirements and allocated test cases to continuously monitor SCFs and detect faults/failures of safety-significance.
- Functions within software should be able to monitor safety critical functions, handle detected faults in a safe manner to prevent hazards from manifesting, such as setting the system to a known safe state. These are often modeled.
- Functional failure conditions should be detected to alert operators and systems by annunciating fault/failures and take autonomous actions to set to a safe state without immediate operator interaction. These are often modeled.
- Autonomous systems are ideal candidates for modeling and most large contractors construct functional behavioral models to prove system functionality works with minimal operator intervention.

It is all about modeling behavior of safety aspects and attributes

- OMG SysML is one of many popular ways in activity diagrams to decompose Use Case, Capability requirements into FUNCTIONAL BEHAVIOR
  - FUNCTIONAL **Decomposition** and BEHAVIOR are KEY to understanding the SAFETY ATTRIBUTES OF A SYSTEM
  - MODELING FUNCTIONAL FAILURE CONDITIONS CAN HELP DETERMINE HAZARDS
  - Documenting Safety Behavior in Models – Beyond Hazard Analysis Templates – can show explicit behavior in normal and off nominal environments

Model-Based Tools

- DO-330 Tools is a supplement to DO-178C. It does not endorse any specific tool.
- Some common industry MBSE tools
  - Rhapsody (Trademark) provides a consistent design model that is also tied to requirements
  - Rhapsody is one of many Enterprise Tools supportive of UML/SysML tools.
  - Safety attributes can be added throughout model-based tools and linked to DOORS (to Rhapsody) or other safety requirements for full traceability from USE CASES, to DOORS, TO MODELING TOOLS and links to safety actions and produced artifacts.
  - Many tools available with excellent capability to help augment safety case with graphical and highly visual flow representation of safety aspects and attributes vs. just words.
  - Magic Draw Rhapsody, SIMULINK, MATLAB, many more excellent tools
  - SCADE, ANSYS Medini are popular safety-critical tools

Some System Safety Advantages of Model-Based Development

- If so designed, MBSE can show the “big safety picture” and explicit safety functions, safeguards, safety features with easy-to-interpret sequence flow diagrams and behavioral flow diagrams of safety-critical functions.
- MBSE improves engineering collaboration, teaming, and communications across domains – same core representation – for safety documentation
- System engineering, software engineering, and safety engineering processes and actual FUNCTIONS and normal/failure CONDITIONS can be visualized vs. word interpretation that can be vague and ambiguous
- Proposed changes (safety changes) can be evaluated
- More consistent safety documentation and traceability improve technical integrity
- Already validated auto-code generation using the tools to perform them can be better analyzed in a model-based setting. A plus for safety.

Tools are Proprietary, Copyrights

- Many Tools – Research them online using BING or GOOGLE
- Many copyright diagrams are online.
- OMG SysML is one of many popular ways in activity diagrams to decompose Use Case, Capability requirements into FUNCTIONAL BEHAVIOR
  - FUNCTIONAL **Decomposition** and BEHAVIOR are KEY to understanding the SAFETY ATTRIBUTES OF A SYSTEM
  - MODELING FUNCTIONAL FAILURE CONDITIONS CAN HELP DETERMINE HAZARDS
  - Documenting Safety Behavior in Models – Beyond Hazard Analysis Templates – can show explicit behavior in normal and off nominal environments

DO-331 Model Based for Aviation

DO-331 is titled “Model Based Development and Verification Supplement to DO-178C and DO-278A”

- DO-178C rewrite completed in 2012 with model-based development and verification
  - DO-331 promulgated as a separate DO since new methods required guidance
- Most modern airborne aviation systems, avionics, flight controls, engine controller software for DO-178C level A and B software (safety-critical functions) have various types of model-based development, model-based architecture using many processes, methods, and techniques.
- Aviation Safety & Airworthiness with subsets of system safety, software safety, software design assurance all require integration into the model-based environment to ensure requirements are met.

DO-178C / DO-331 Objectives

Objectives for DO-178C suite of documents, including the supplements:

- Promote safe implementation of aeronautical software
- Provide clear and consistent ties with the systems and safety processes
- Address emerging software trends and technologies
- Implement an approach that can change with the technology
- Industry-accepted guidance for satisfying airworthiness requirements for avionics equipment

Purpose of DO 331 in Software Safety Area

- Industry-accepted guidance for satisfying airworthiness requirements for avionics equipment
  - To provide guidelines for software to comply with:
    - Proof of no intended function
    - Proof of performance in an avionics LRU installation
  - To provide agreed criteria consistent with civil certification authorities
  - By treaty agreement, this applies to NATO nations and any other countries recognizing this set of guidelines for aviation software

- Results Needed
  - Agreed criteria for airworthiness certification requirements for software that doesn’t differ from one person or certification authority to another
  - Allows for recognition of an aircraft model capability by air traffic control for airspace access and interoperability

Model Based Software - Aviation

DO-331 Model-Based Development and Verification Supplement to DO-178C and DO-278A

- **Objectives for DO-178C suite of documents, including the Supplement DO-331.**
  - Promote safe implementation of aeronautical software
  - Provide clear and consistent ties with the systems and safety processes
  - Address emerging software trends and technologies
  - Implement an approach that can change with the technology
  - Industry-accepted guidance for satisfying airworthiness requirements for avionics equipment

Model-Driven Architecture

Model-Driven Architecture (MDA)

- System and software architecture enhances software safety
- Standardized approach from Object Management Group (OMG) based on consortium from aerospace, healthcare, high technology, government.
- Meshes well with complex domain engineering and software-intensive systems
- Structure and convention is vital to safety-critical systems and software
  - Object-oriented technology
  - Uniform Modeling Language (UML), XML, SySML, others
  - Platform Specific Models (PSM)
  - Robust Real Time Operating Systems (RTOS) for safety certification
    - Middleware layer between application software and RTOS
    - Deterministic behavior in the safety domain
    - Kernels, wrappers, integrity protocols contribute to safety and security

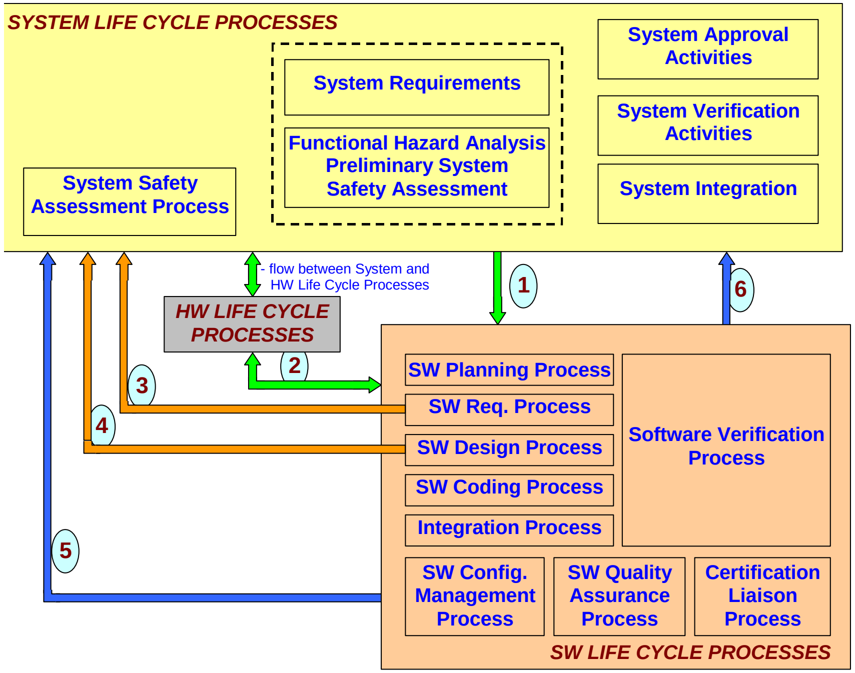
MDA and Safety Models

- Model-Driven Architectures (MDA) and various computer platform independent models, platform-specific models, or dual or multi-core models should be qualified for the intended operating environment.
- Safety models can show platforms and systems capable to retain SCFs (no loss of function) by hosting safety-critical applications with various forms of redundancy.
- Parallel and independent redundant systems can be modeled to show mitigation of hazards, lower probability of total failure, design safety features to mitigate safety risks.

Information flow between system and SW life cycle processes

SAFETY ENGINEERING SEAC & ANALYSIS CENTER

Context for use of DO-331 MBD


SYSTEM LIFE CYCLE PROCESSES
System Approval Activities
System Requirements
System Verification Activities
Functional Hazard Analysis Preliminary System Safety Assessment
System Safety Assessment Process
System Integration
flow between System and HW Life Cycle Processes
1
6
HW LIFE CYCLE PROCESSES
2
3
4
5
SW Planning Process
SW Req. Process
SW Design Process
SW Coding Process
Integration Process
Software Verification Process
SW Config. Management Process
SW Quality Assurance Process
Certification Liaison Process
SW LIFE CYCLE PROCESSES

ARP4754A

DO254

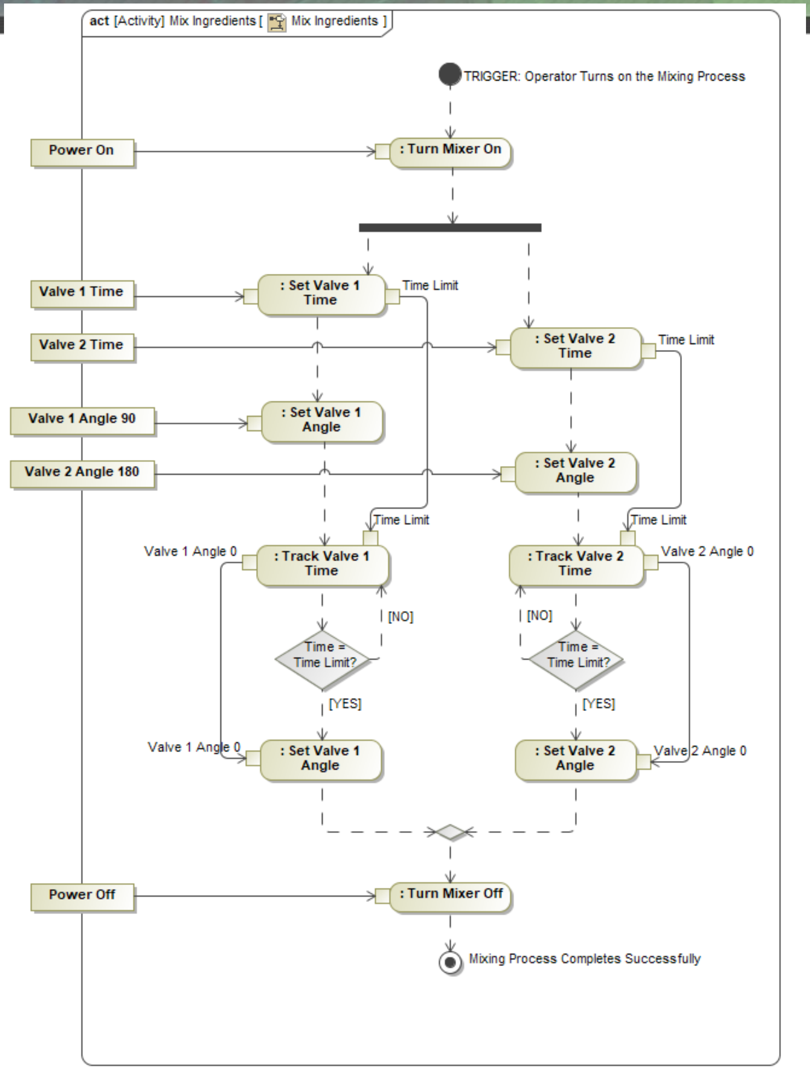
DO178C

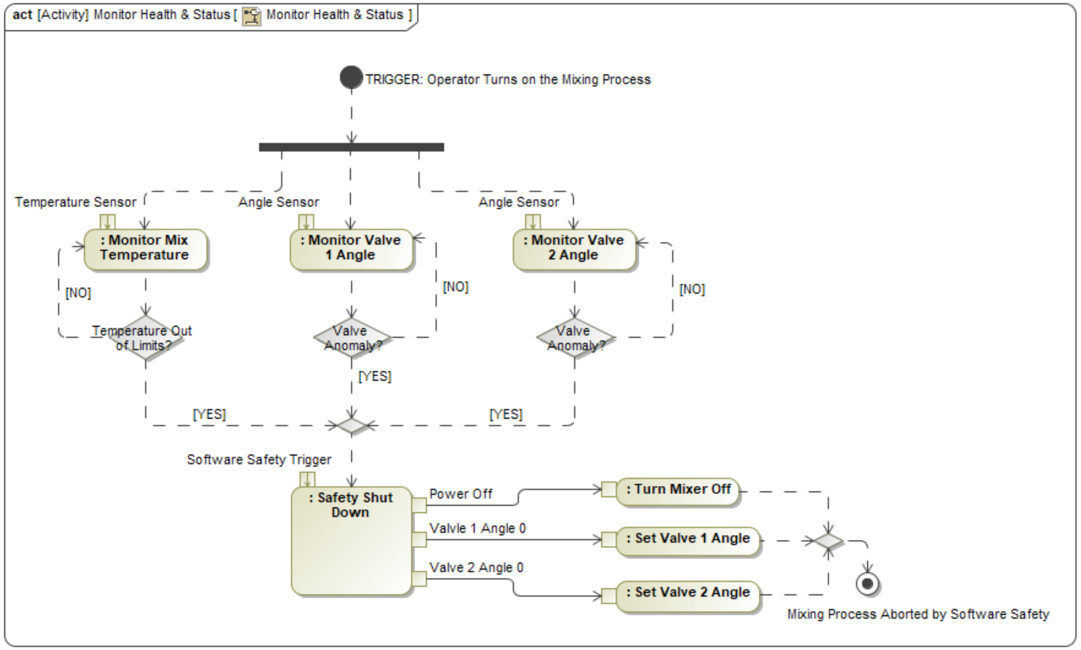
DO-331* is used with DO-178C

*It adds and modifies DO-178C objectives for MBD aspects

Functional Safety Activity Diagram

- Activity Diagrams depicting functions using SYSML and common tool set
- The following safety activity diagram using SYSML depicts functions of a simple mixing function and how such can be modeled in detail. The SYSML models shows the chemical mixing process to determine timing, valve angle, mixture, temperature, and limits.
- The second activity diagram shows how software is used to monitor health and status to mitigate a hazard by detecting an out-of-limits temperature, and/or valve anomaly to shut down the process. Any safety function, simple or complex can be modeled using SYSML.
- Such models should be included in safety documentation as convincing, objective evidence the function is well understood and designed to mitigate hazards and risks. Activity Diagrams are valuable to show exactly how a SCF behaves under normal or abnormal failure conditions, but SYSML diagrams can show virtually any safety activity and are highly encouraged in the mix of safety tasks to convincing evidence and proof of safety concept or safety features.
- Source: Pie Valley Consulting LLC.


act [Activity] Mix Ingredients [Mix Ingredients]
TRIGGER: Operator Turns on the Mixing Process
Power On
: Turn Mixer On
Valve 1 Time
: Set Valve 1 Time
Time Limit
Valve 2 Time
: Set Valve 2 Time
Time Limit
Valve 1 Angle 90
: Set Valve 1 Angle
Valve 2 Angle 180
: Set Valve 2 Angle
Time Limit
Time Limit
Valve 1 Angle 0
: Track Valve 1 Time
: Track Valve 2 Time
Valve 2 Angle 0
[NO]
[NO]
Time = Time Limit?
Time = Time Limit?
[YES]
[YES]
Valve 1 Angle 0
: Set Valve 1 Angle
: Set Valve 2 Angle
Valve 2 Angle 0
Power Off
: Turn Mixer Off
Mixing Process Completes Successfully


act [Activity] Monitor Health & Status [Monitor Health & Status]
TRIGGER: Operator Turns on the Mixing Process
Temperature Sensor
Angle Sensor
Angle Sensor
: Monitor Mix Temperature
: Monitor Valve 1 Angle
: Monitor Valve 2 Angle
[NO]
[NO]
[NO]
Temperature Out of Limits?
Valve Anomaly?
Valve Anomaly?
[YES]
[YES]
[YES]
Software Safety Trigger
: Safety Shut Down
Power Off
Valvle 1 Angle 0
Valve 2 Angle 0
: Turn Mixer Off
: Set Valve 1 Angle
: Set Valve 2 Angle
Mixing Process Aborted by Software Safety

Summary

- MBSE and software engineering processes using methods and tools are here to stay and growing with proven value
- INCOSE and DoD mandating on more complex programs
- DO-178C and RTCA/DO-331 requiring MBD for software quality, software safety, software design assurance, airworthiness.
- DoD, NASA, FAA, and many contractors are migrating towards model-based programs
- Challenge is like all other evolving cultural changes
  - Understanding
  - Seeing value and buying into a proven process

System Safety and Software Safety must plug into the model and be part of the multi-domain process

Leading-Edge Endorsements for MBSE

SAFETY ENGINEERING SEAC & ANALYSIS CENTER

Department of Defense

https://sercuarc.org/wp-content/uploads/2018/06/Digital-Engineering-Strategy_Approved.pdf

A-P-T Research, Inc. Model Based Workshop

https://www.apt-research.com/MBSESSS/Agenda.pdf

Model Based Safety Analysis by NASA and NASA Langley

http://shemesh.larc.nasa.gov/fm/papers/Model-BasedSafetyAnalysis.pdf

SAE1005 Model-Based Functional Safety for Complex Software Intensive Systems – Draft 2019 in Review by G-48