INL/EXT-20-59904

Light Water Reactor Sustainability Program

Assessment of Verification and Validation Status - EMRALD and HUNTER

September 2020

U.S. Department of Energy

Office of Nuclear Energy

INL/EXT-20-59904

Assessment of verification and validation status – EMRALD and HUNTER

Yong-Joon Choi

September 2020

Prepared for the
U.S. Department of Energy
Office of Nuclear Energy

EXECUTIVE SUMMARY

The United States nuclear industry is facing a strong challenge to ensure maximum safety while enhancing economic benefit. Safety is a key parameter to all aspects related to light water reactor (LWR) nuclear power plants (NPPs), especially cost savings. Since the goal is to extend the lifetimes of these NPPs, the traditional deterministic safety concept may not guarantee a current economic asset. The Light Water Reactor Sustainability (LWRS) Program has been promoting a wide range of research and development (R&D) in this field to maximize the safety, economics, and performance of these NPPs through improved scientific understanding.

One of the best practices to achieve this goal is to identify and optimize safety margins, which can lead to cost reduction. To do this, under the LWRS framework, the Risk-Informed Systems Analysis (RISA) Pathway will focus on the optimization of safety margin and minimization of uncertainties to ensure both safety and economics at the highest level. The RISA Pathway will provide enhanced capabilities for analyzing and characterizing LWR systems performance by developing and demonstrating methods, tools, and data to enable risk-informed margins management (RIMM).

The goals of the RISA Pathway are twofold: (1) deploy the risk-informed tools and methods that enable better representation of safety margins and factors that contribute to cost and safety; and (2) conduct advanced risk assessment applications with industry to support margin management strategies that enable more cost-effective plant operation. The tools and methods provided by the RISA Pathway will support effective margin management for both active and passive safety systems, structures, and components (SSC) of an NPP.

The tools and methods used in the RISA Pathway should have high confidence and highest technical maturity for and implementation to industry at its current setting. They should also have a capability to support risk-informed decision making for both probabilistic and deterministic elements of safety. The RISA Pathway will, therefore, perform a comprehensive assessment of verification and validation (V&V) status of RISA Toolkit to enhance credibility RISA Toolkit which be used by industry.

This report summarizes assessment technical maturity of RISA dynamic PRA code EMRALD and dynamic HRA framework HUNTER including, V&V status, specific information of the tool such as capability and features, quality assurance program, developer/independent V&V record, separation/integral tests history, user documents, and feedback.

CONTENTS

FIGURES

TABLES

ACRONYMS

3D	three-dimensional
AOO	Anticipated Operation Occurrence
ATF	Accident Tolerant Fuel
BEPU	Best Estimate Plus Uncertainty
BWR	Boiling Water Reactor
DBA	Design Basis Accident
DNC	Dynamic Natural Convection
DOE	U.S. Department of Energy
EPRI	Electric Power Research Institute
ER	Equipment Reliability
ESFAS	Engineered Safety Feature Actuation System
FLEX	Diverse and Flexible Coping Strategy
FY	Fiscal Year
GOTHIC	Generation of Thermal-Hydraulic Information for Containment
HRA	Human Reliability Analysis
HEP	Human Error Probability
I&C	Instrumentation and Controls
INL	Idaho National Laboratory
LOCA	Loss Of Coolant Accident
LWR	Light Water Reactor
LWRS	Light Water Reactor Sustainability
MAAP	Modular Accident Analysis Program
MOOSE	Multiphysics Object-Oriented Simulation Environment
MP-BEPU	Multi-Physics Best Estimate Plus Uncertainty
NEI	Nuclear Energy Institute
NEUP	Nuclear Energy University Program
NPP	Nuclear Power Plant
NRC	U.S. Nuclear Regulatory Commission
PDF	Probability Density Function
PRA	Probabilistic Risk Assessment
PSF	Performance Shaping Factor
PWR	Pressurized Water Reactor
R&D	Research and Development

RAVEN	Risk Analysis in a Virtual Environment
RCIC	Reactor Core Isolation Cooling
RD&D	Research, Development, and Demonstration
RIA	Reactivity Initiated Accident
RIMM	Risk-Informed Margin Management
RI-MP-BEPU	Risk-Informed Multi-Physics Best Estimate Plus Uncertainty
RISA	Risk-Informed Systems Analysis
RISC	Risk-Informed Safety Categorization
RPS	Reactor Protection System
SLR	Second License Renewal
SNL	Sandia National Laboratories
SSC	System, Structure, and Component
TTEXOB	Terry Turbine Expanded Operating Band

1. INTRODUCTION

1.1 Background

The risk-informed tools and methods, so-called RISA toolkit, to be used in industry needs an appropriate level of verification, validation, and uncertainty characterization to give maximum credibility for safe nuclear power plant operation. A unique focus of our assessment is on the maturity and usability of a specific tool. Since the tools and methods being used in RISA Pathway needs to have highest technical maturity and could be used in industry immediately, the comprehensive assessment of verification and validation (V&V) status level is one of most important tasks for successful industry deployment of risk-informed tools and methods [1].

Most of tools have developer and/or user V&V program to improve quality of certain tool. However even many of V&V programs are reporting some of the tools still need to confirm its technical maturity based on philosophy of the risk-informed, thus, application for Probabilistic Risk Assessment (PRA). The RISA Pathway will therefore use the following goals to assure RISA toolkit quality:

- Define requirements based on risk-informed concept
- Investigate and review development and V&V status for technical maturity assessment.
- Identify technical gap and propose improvement to meet RISA toolkit requirements

The first part of the work was to collect and summarizes available information of selected RISA Toolkit. This includes list of documents and accessibility of the V&V records for each version of the software, if necessary. The selected toolkit was then evaluated its maturity level. Based on risk-informed concept, capability and/or applicability of PRA method will be the main requirement as RISA toolkit. The high-level requirements were set based on the Requirement Traceability Matrix (RTM) to capture the requirements from user and developer of the project or software. The importance of each requirement were evaluated by Phenomena Identification and Ranking Technology (PIRT) method which gives systematic way of gathering information from experts on a specific subject, and ranking the importance of the information, in order to meet some decision-making objective to determine what has highest priority for research on that subject. Finally, Technology Readiness Level (TRL) was used to measure level of maturity. Developed by National Aeronautics and Space Administration (NASA), the TRL is a method for estimating the maturity of the technologies during the development and acquisition phase of certain technology. A total of nine levels are set for RISA Pathway from low level (1~4) to high (5~9), which represents from research and development (R&D) status to ready for industrial use. Figure 1-1 shows an example of TRL increase through the RISA pathway pilot project.

The industry application pilot demonstration projects (pilot project in-short) are main features of the RISA Pathway to give clear vision on risk-informed margin recovery to U.S. nuclear industry and decision makers. The pilot projects focus on specific scope of phenomena, components, and simulation capabilities needed to address the given issue area. As a part of these applications, refinement of the associated methods and tools would continue at a reduced level of effort compared to the effort associated with the RISA toolkit development. However, not all tools are suitable for use in RISA pilot project, thus, certain tool may need additional development. The RISA Pathway will therefore identify technical gap of a target toolkit and will propose additional development to meet requirements as a RISA toolkit.

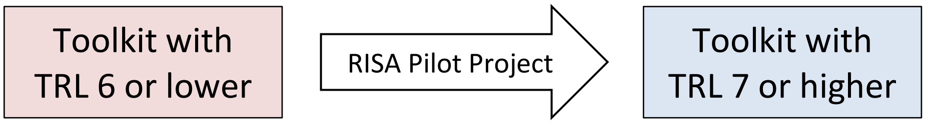


Figure 1-1 Schematic diagram of TRL improvement in RISA pathway

As the development and capabilities of the RISA toolkit progresses, the pathway will collaborate with industry to determine how to transition the RISA toolkit to a user-supported community of practice. The assessment of V&V status project will support smooth industry deployment of RISA toolkit including planning for lifecycle software management issues such as training, software quality assurance, and development support.

In FY-2019, technical maturity was assessed for RISA thermal-hydraulics tool RELAP5-3D and multi-purpose PRA tool RAVEN [2]. Both of tools were found sufficient level of technical maturities and proposed potential needs to be used in RISA pathway and future industrial deployment. The outcome of this report supported developer and user to understand current level of technical maturity and clarifies the need of additional development of technical gap for industrial deployment. As a consequence, the best-estimate plus uncertainties (BEPU) capabilities are under development for RELAP5-3D.

1.2 RISA Industry Pilot Demonstration Project

As of September 2020, a total of eight "Pilot Demonstration Projects" are conducting in the RISA Pathway. The "Pilot Demonstration Project" (pilot project in short) was proposed in May 2018 through the Idaho National Laboratory (INL) organized special workshop with delegations from major U.S. nuclear utilities to discuss the issues related to NPP safety margins and economics. Table 1 shows how each pilot demonstration project relates to each RISA research and development (R&D) focus area. These are the most relevant industry topics that can potentially impact plant operations in a significant way making them interesting and relevant applications for the RISA toolkit. The RISA Pathway will continue to communicate with various U.S. nuclear industries to collect issues and develop additional pilot demonstration projects.

Table 1-1 Pilot demonstration projects related to RISA R&D focus areas.

RD&D Focus Areas	Pilot Demonstration Projects
Enhanced resilient NPP concepts	RISA-Enhanced Resilient Plant Systems Enhanced Operation Strategies for System Components.
Cost and risk categorization applications	Risk-Informed Asset Management Plant Health Management.
Margin recovery and operation cost reduction	Enhanced Fire Probabilistic Risk Assessment (PRA) Modernization of Design Basis Accidents Analysis with Application on Fuel Burnup Extension Digital Instrumentation and Control (I&C) Risk Assessment Plant Reload Process Optimization

1.3 List of RISA toolkit

Based on current and potential RISA pilot projects, Figure 1-2 shows the list of computational software proposed to be used, and will deployed to the industry for risk-informed margin management. Most of tools are completely developed and currently used in various industries. However, some tools are still under development and may need additional V&V activities.

A brief description of potential RISA toolkits are as follows [1].

- **BISON:** BISON is a finite element-based nuclear fuel performance code applicable to a variety of fuel forms including LWR fuel rods, tristructural isotropic particle fuel and metallic rod and plate fuel. It is an advanced fuel performance code being developed at Idaho National Laboratory

(INL) and offers distinctive advantages over FRAPCON/FRAPTRAN, such as three-dimensional (3D) simulation capability, etc. BISON solves the fully-coupled equations of thermomechanics and species diffusion, for either one-dimensional (1D) spherical, two-dimensional (2D) axisymmetric, or 3D geometries.

- **CFAST:** Developed by National Institute of Standards and Technology (NIST), the Consolidated Model of Fire and Smoke Transport (CFAST) is a computer program that fire investigators, safety officials, engineers, architects, and builders can use to simulate the impact of past or potential fires and smoke in a specific building environment. CFAST is a two-zone fire model used to calculate the evolving distribution of smoke, fire gases, and temperature throughout compartments of a building during a fire. The CFAST package includes NIST's Smokeview program, which visualizes with colored, 3D animations, the results of the CFAST simulation of a specific fire's temperatures, various gas concentrations and growth and movement of smoke layers across multi-room structures.

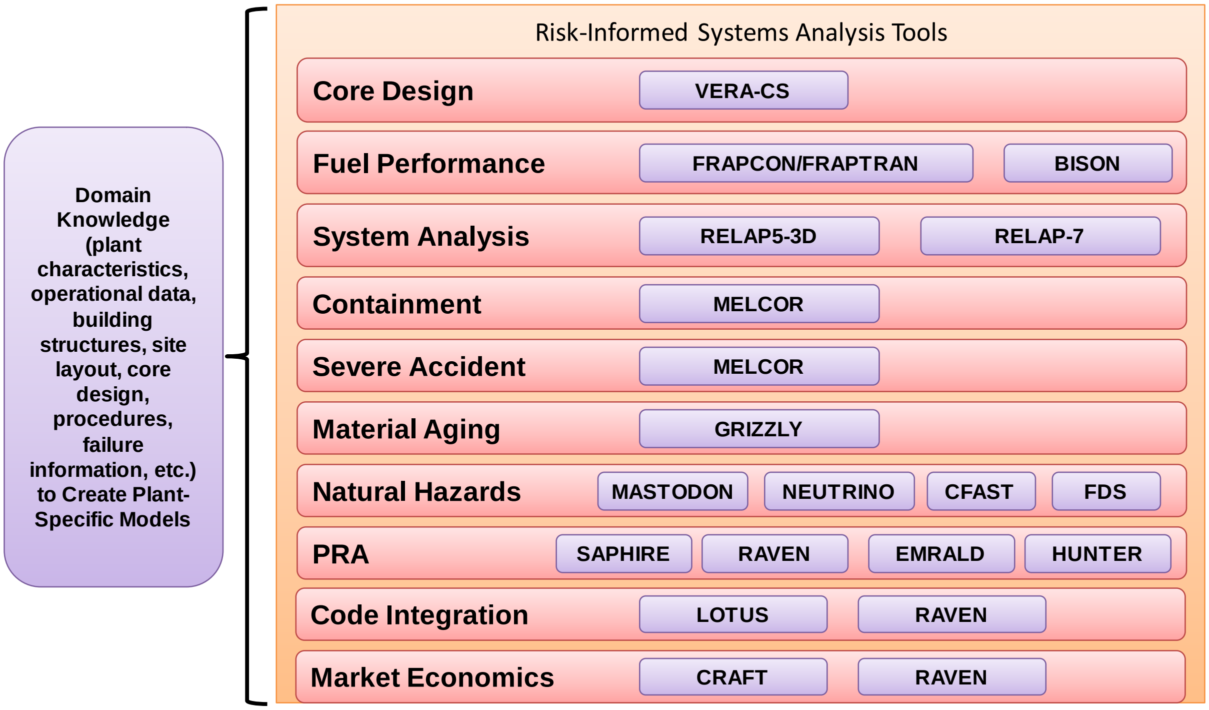


Figure 1-2 Current software modules used to perform RISA-specific analyses.

- **CRAFT:** CRAFT is a stochastic analysis framework that has been designed to evaluate the risk of SSCs of complex systems, including NPPs. The risk is evaluated from both a financial and a safety perspective by explicitly considering aging of SSCs and their impact on the overall plant risk. CRAFT applications range from plant asset management to plant risk diagnostics and prognostic.
- **EMRALD:** EMRALD is a state-based discrete event simulation tool that can calculate system failure probabilities, couple multiple simulations, and perform dynamic PRA. A key part of the EMRALD tool is to develop an object-oriented model that is flexible enough to support the varied dynamic simulation models (e.g., fails to operate, fails on demand). By having a state-based approach, it can integrate different hazards into a single comprehensive model. For example, a single model can include fire-, flooding-, transient-, and seismic-initiating events.

Each of these events becomes a trigger into the state-based approach that tells the model to make a transition based upon the specific initiator.

- **FDS:** NIST developed the computational fluid dynamics Fire Dynamics Simulator (FDS) code to perform computational fire modeling and simulation. The code has been extensively validated for the types of fire scenarios encountered in both standard buildings, as well as nuclear environments. FDS facilitates the simulation of combustion, including fire migration, of an arbitrary number of materials in geometrically complex environments.
- **FRAPCON/FRAPTRAN:** FRAPCON/FRATRAN is a suite of codes developed by Pacific Northwest National Laboratory (PNNL) for the NRC for the purposes of performing fuel performance analyses under steady state (FRAPCON) and transient (FRAPTRAN) conditions. FRAPCON is a computer code that calculates the steady-state response of LWR fuel rods. The code calculates the temperature, pressure, and deformation of a fuel rod as functions of time-dependent fuel rod power and coolant boundary conditions. FRAPTRAN calculates the transient performance of LWR fuel rods during reactor transients and hypothetical accidents such as LOCAs, anticipated transients without scram, and reactivity-initiated accidents. FRAPTRAN calculates the temperature and deformation history of a fuel rod as a function of time-dependent fuel rod power and coolant boundary conditions.
- **GRIZZLY:** GRIZZLY is a simulation code being developed to simulate the progression of aging mechanisms and SSCs in LWRs and to assess their ability to safely perform their intended engineering functions after being subjected to aging. GRIZZLY is ultimately planned to have capabilities for modeling a variety of structures, but current development is focused on reactor pressure vessels (RPVs) and concrete structures because of the essential functions and extreme difficulty of mitigating degradation or replacement of those components. For RPVs, GRIZZLY has a modern and flexible architecture for multidimensional engineering fracture mechanics analysis, which allows it to compute the probability of fracture in the presence of a population of pre-existing flaws that can serve as fracture initiation sites under a given transient event. It also has a set of models being developed to predict microstructure evolution under irradiation, which will be used to provide improved predictive models of embrittlement that can be applied for long-term operation scenarios. For concrete structures, GRIZZLY has coupled physics models to predict expansive mechanisms, including alkali-silica reaction and radiation-induced volumetric expansion, and their effects on the mechanical response of the structure, including fracture and damage.
- **HUNTER:** HUNTER is a flexible hybrid approach that functions as a framework for dynamic modeling, including a simplified model of human cognition—a virtual operator—that produces relevant outputs, such as the human error probability (HEP), time spent on task, or task decisions based on relevant plant evolutions. HUNTER is the human reliability analysis counterpart to the RAVEN framework used for dynamic PRA. Although both RAVEN and HUNTER are under various stages of development, there has been a successfully integrated and implemented RAVEN-HUNTER initial demonstration. The demonstration centers on a station blackout scenario, using complexity as the sole virtual operator performance-shaping factor (PSF). The implementation of RAVEN-HUNTER can be readily scaled to other nuclear power plant scenarios of interest and will include additional PSFs in the future.
- **MASTODON:** MASTODON is a tool that will have the capability to perform stochastic Non-linear Soil-Structure Interaction (NLSSI) in a risk framework coupled with virtual NPP. These NLSSI simulations will include structural dynamics, time integration, dynamic porous media flow, hysteretic nonlinear soil constitutive models (i.e., elasticity, yield functions, plastic flow directions, and hardening softening laws), hysteretic nonlinear structural constitutive models, and geometric nonlinearities at the foundation (i.e., gapping and sliding) [1].

- **MELCOR:** MELCOR is a computational code developed by Sandia National Laboratories (SNL) for the NRC, DOE, and the International Cooperative Severe Accident Research Program (CSARP). MELCOR simulates the response of LWRs during severe accidents. Given a set of initiating events and operator actions, MELCOR predicts the plant's response as the accident progresses. MELCOR also includes containment transient analysis capabilities to model thermal hydraulic phenomena (within a lumped-parameter framework) for existing containment designs for boiling water reactors (BWRs) and PWRs.
- **Neutrino:** Neutrino is a mesh-free, smooth particle hydrodynamics-based solver developed by Centroid Lab, which also uses advanced boundary handling and adaptive time stepping. Neutrino is an accurate fluid solver and is being used to simulate coastal inundation, river flooding, and other flooding scenarios. Neutrino code can model friction and adhesion between solid/fluid boundaries and various adhesive hydrodynamic forces between fluid/fluid particles [5].
- **RAVEN:** RAVEN is a flexible and multi-purpose uncertainty quantification, regression analysis, probabilistic risk assessment, data analysis and model optimization framework, designed to perform parametric and stochastic analyses based on the response of complex systems codes. It can communicate directly with the system codes described above and below (e.g., RELAP5-3D, Neutrino, BISON, MAAP, etc.), which are currently used to perform plant safety analyses. Depending on the tasks to be accomplished and on the probabilistic characterization of the problem, RAVEN perturbs (e.g., Monte-Carlo, Latin hypercube, reliability surface search) the response of the system under consideration by altering its own parameters. The data generated by the sampling process is analyzed using classical statistical and more advanced data mining approaches. RAVEN also manages the parallel dispatching (i.e. both on desktop/workstation and large High-Performance Computing machines) of the software representing the physical model. RAVEN heavily relies on artificial intelligence algorithms to construct surrogate models of complex physical systems in order to perform uncertainty quantification, reliability analysis (limit state surface) and parametric studies.
- **RELAP5-3D:** The RELAP5-3D code has been developed for best-estimate transient simulation of LWR coolant systems during postulated accidents. Specific applications of the code have included simulations of transients in LWR systems, such as LOCA, Anticipated Transients without Scram (ATWS), and operational transients, such as loss of feed water, loss of offsite power, station blackout, and turbine trip. RELAP5-3D, the latest in the series of RELAP5 codes, is a highly generic code that, in addition to calculating the behavior of the reactor coolant system during a transient, can be used for simulation of a wide variety of hydraulic and thermal transients in both nuclear and nonnuclear systems involving mixtures of vapor, liquid, non-condensable gases and nonvolatile solutes.
- **RELAP-7:** The RELAP-7 code is the next generation nuclear reactor system safety analysis code being developed at INL. The code is based on INL's modern scientific software development framework, Multi-Physics Object Oriented Simulation Environment (MOOSE). The overall design goal of RELAP-7 is to take advantage of the previous thirty years of advancements in computer architecture, software design, numerical integration methods, and physical models. The result will be a reactor system analysis capability that retains and improves upon RELAP5-3D's capabilities and extends the analysis capability for all reactor system simulation scenarios.
- **SAPHIRE:** SAPHIRE is a software application developed for performing a complete PRA using a personal computer (PC) running the Microsoft Windows operating system. SAPHIRE enables users to supply basic event data, create and solve fault and event trees, perform uncertainty analyses and generate reports. For NPP PRAs, SAPHIRE can be used to model a plant's response to initiating events, quantify core damage frequencies, and identify important contributors to core damage (Level 1 PRA). The program can also be used to evaluate containment failure and release

models for severe accident conditions given that core damage has occurred (Level 2 PRA). In addition, SAPHIRE can be used to analyze both internal and external events and, in a limited manner, to quantify the frequency of release consequences (Level 3 PRA).

- **VERA-CS:** VERA-CS is a core simulator tool developed by the Consortium on Advanced Simulation of LWRs (CASL) and includes coupled neutronics, thermal-hydraulics, and fuel temperature components with an isotopic depletion capability. The neutronics capability is based on MPACT, a 3D whole core transport code. The thermal-hydraulics and fuel temperature models are provided by the COBRA-TF (CTF) subchannel code. The isotopic depletion is performed using the ORIGEN code system.

The industry codes such as CAFTA (i.e., a PRA tool), Modular Accident Analysis Program (MAAP) (i.e., a systems analysis tool), and GOTHIC (i.e., a containment response tool) will be used in the pilot demonstration projects:

- **CAFTA**: Developed by the Electric Power Research Institute (EPRI), CAFTA is an integrated tool to perform PRA, incorporating linking event tree/fault tree methodology. The code is a fault tree analysis tool, utilizing the full power of today's PCs and providing the ability to effectively model and analyze complex systems. As fault tree analysis assumes a greater importance in many industries, the need to develop models in a logical and efficient manner has increased. CAFTA code addresses this need by providing a set of interactive editors, databases, and model evaluation tools. This interactive environment promotes the smooth flow of information throughout the model development, quantification, and results interpretation process.
- **MAAP**: MAAP is a fast-running computer code that simulates the response of LWR and heavy water reactor moderated NPPs for both current and Advanced Light Water Reactor designs. It can simulate both LOCA and non-LOCA transients for PRA applications as well as severe accident sequences, including actions taken as part of the Severe Accident Management Guidelines.
- **GOTHIC**: GOTHIC is a versatile, general-purpose thermal-hydraulics software package, which solves the conservation equations for mass, momentum, and energy for multi-component, multi-phase compressible flow in lumped parameter and/or multi-dimensional (i.e., 1D, 2D, or full 3D) geometries. The ability to combine these different nodalization options in a single model allows GOTHIC to provide computationally efficient solutions for multi-scale applications.

1.4 RISA Toolkit Deployment Plan

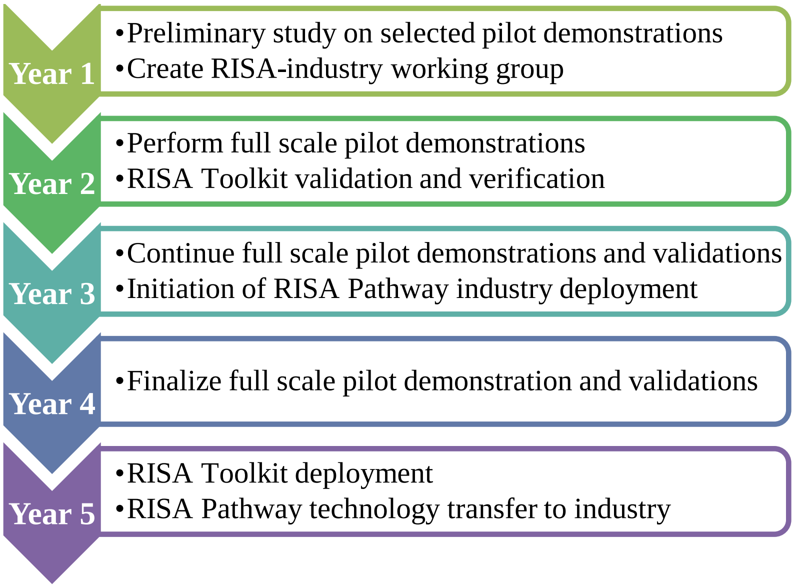
The RISA Toolkits are a set of computer software that will be used through the industry application pilot demonstrations and deployed to U.S. nuclear industry to support risk-informed margin management (RIMM) analysis. Since not all software has been fully verified and validated, the RISA Pathway will perform an appropriate degree of software Verification and Validation (V&V) to give clear understanding to future RISA toolkit users in US nuclear industry. The industry will engage to participate in the selected pilot demonstration projects. The RISA Pathway will continuously communicate with industry to develop issues and to collect feedback. The RISA toolkit deployment will have the following four process steps:

1. Select tools and methods.
2. Confirm verification and validation status.
3. Pilot demonstrations using selected tools and methods.
4. Industry deployment and feedback.

In order to meet above four deployment steps for a tool development, the RISA Pathway proposes maximum five years project plan as shown in Figure 1-3. For the first year, the RISA Pathway will focus on the initiation of selected pilot demonstration projects and will deliver preliminary results on the case studies. The feedback on selected pilot demonstration projects will be communicated with industry through a RISA Pathway working team. Based on the preliminary studies, the pilot demonstration projects will be extended to full-scale analysis during next two years. The validation and verification of associated the RISA toolkit will be done in this period. For the last two years of the project, the RISA Pathway will develop optimized methods of R&D results implementation to industry as well as long-term support plan by research institutes for U.S. nuclear industry to provide sustainable benefit from risk-informed margin management. It is noted that the project timeline could be different to the technical maturity and development status of using the tool and method.

1.4.1 Selection of RISA Toolkit

The tools and methods used in the RISA Pathway should have high-confidence and enough technical maturity and ability to cover a wide RIMM area range. Advanced technologies should be applied to the RISA Toolkits, such as multi-physics and multi-scale analysis, and cutting-edge computational proficiency as well as capability of uncertainty control if necessary. The toolkit should also have the capability to support risk-informed decision-making for both probabilistic and deterministic elements of safety. Current RISA toolkit includes various computer simulation tools that can cover wide range of work scopes. Many of tools are currently available in related industry, and well validated with mature technology level. However, there are still tools under development or needing to be verified and/or validated to confirm the technical maturity and suitability for the RISA Pathway framework.



Year 1
- Preliminary study on selected pilot demonstrations
- Create RISA-industry working group

Year 2
- Perform full scale pilot demonstrations
- RISA Toolkit validation and verification

Year 3
- Continue full scale pilot demonstrations and validations
- Initiation of RISA Pathway industry deployment

Year 4
- Finalize full scale pilot demonstration and validations

Year 5
- RISA Toolkit deployment
- RISA Pathway technology transfer to industry

Figure 1-3 Notional 5-year plan of a RISA Pathway Industry Application

1.4.2 Verification and Validation (V&V) of the RISA Toolkit

To provide confidence during industry deployment, the selected tools and methods should address quality-assurance levels appropriate for industry use. Well-known methods such as V&V and uncertainty quantification of the produced result will enhance credibility of the selected tools and enable industry to use them with confidence. The selected RISA Toolkits will be examined to confirm V&V status to show its Technology Readiness Level (TRL) and to assure quality of the outcomes. The RISA Pathway will deliver the annual report of V&V examination and confirmation data for the selected RISA toolkit. The deliverable will include specific information of the tool such as capability and features, quality assurance program, developer/independent V&V record, separation/integral tests history, user documents, and feedback.

1.4.3 Pilot demonstration using RISA Toolkit

As of FY-2020, a total of eight RISA pilot demonstration projects are proposed during comprehensive discussion with U.S. nuclear industries and related institutes. Each project has its own selected tools and methods to show optimum RIMM, which aims to enhance both safety and economics. The RISA Pathway will maintain strong engagement with the U.S. nuclear industry to perform each pilot demonstration project. Direct participation of industry will provide better understanding on arising issue and can facilitate promoting innovative solutions, as well as smooth deployment of RISA toolkit in the future.

1.4.4 RISA Toolkit Industry Deployment and Feedback

Successful deployment of the RISA Toolkits to the U.S. nuclear industry is the one main goal of the RISA Pathway. The US nuclear industry has been involved in the RISA Pathway from its initiation and will support the pilot demonstration project by using the RISA toolkit. During the pilot demonstration project phase, industry will have enough time to experience selected tools and methods. The RISA Pathway will organize its annual meeting with leading U.S. nuclear representatives to develop additional pilot demonstration projects, collect feedback to upgrade on-going projects, and discuss an effective RISA toolkit implementation strategy. Licensing and regulation issues will be also addressed. The result of this meeting will be published and shared for long-term maintenance of the RISA toolkit.

2. METHODS FOR TECHNICAL MATURITY EVALUATION

The RISA Pathway toolkit mainly represents related computer software to be used for pilot demonstration projects as listed in Section 1.3. Since simulation appears to be the only feasible option to capture realistic aspects of the multi-physics behavior of a complex system, the software user should clearly understand features, characteristics and any potential issues of selected tool. The RISA Pathway focuses on use of modeling and simulation tool for risk-informed approach, thus, probabilistic risk assessment (PRA). This implies selected software should have own PRA capability or could be coupled with other PRA tools to apply for RIMM of the nuclear power plant.

The technical maturity evaluation of the RISA toolkit will therefore focus on the capability, usability, and applicability of PRA. If the toolkit is still under development, the developer must have responsibility for further maintenance once development is finalized. Version control, V&V history, license maintenance, quality assurance programs and customer service are also required for sustainable use of the software. The RISA Pathway will summarize above information on RISA toolkit to facilitate deployment of the tools to industry.

Figure 2-1 is the schematic diagram of RISA Toolkit maturity evaluation procedure. The list of requirements that toolkit should fulfilled are selected from Requirement Traceability Matrix (RTM) method. Since 2015, the RISA Pathway has been using RTM to evaluate development status and V&V proposal for RELAP-7, the next generation nuclear system safety analysis code [3]. This method has been showing efficient analysis result by capturing necessary requirement for developers by reflecting needs from the users. The RTM will be then evaluated by Phenomena Identification and Ranking Technology (PIRT) to categorize importance of the requirements. The RISA Pathway has been using the PIRT method to review the technical maturity of NEUTRINO, a smooth particle hydrodynamics computational fluid dynamics software [4]. Three levels of ranking, high - medium - low, will be used for each requirement. Finally, the Technology Readiness Level (TRL) method will give a maturity level of each requirement. The RISA Toolkit maturity assessment matrix is proposed in Appendix A.

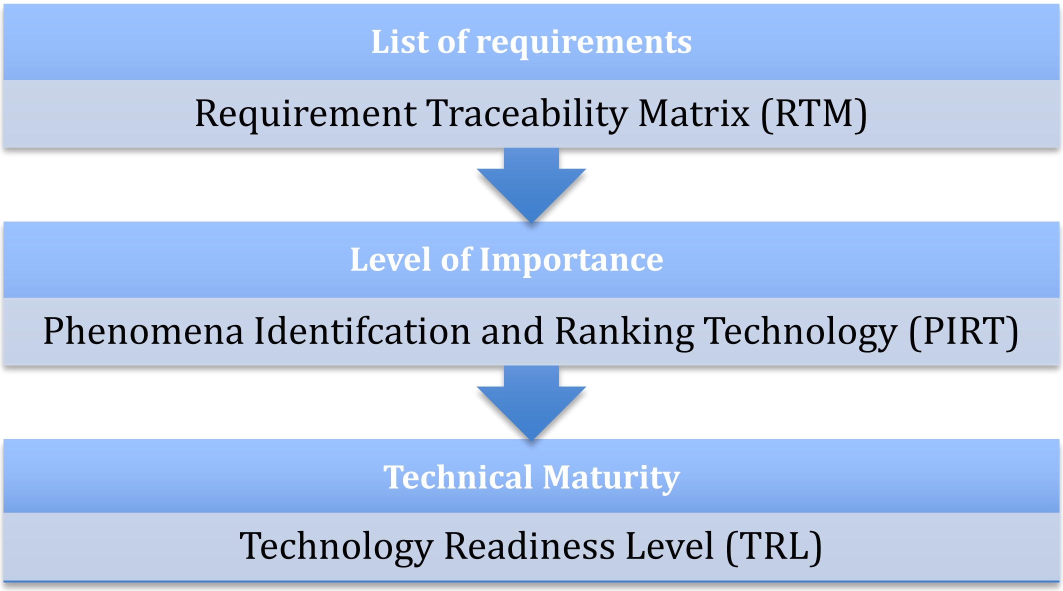


Figure 2-1 RISA Toolkit maturity evaluation procedure

2.1 Requirement Traceability Matrix (RTM)

The Requirements Traceability Matrix (RTM) is created to associate specific requirements with the work products that satisfy them. Tests are also associated the requirements on which they are based so documented proof exists that the product has met the requirement. The matrix provides unique identifiers for each requirement and ensures completeness that lower-level requirements come from higher-level requirements. This matrix provides a tangible item that can be examined by the customer and the provider alike. Information that may not be clearly explained in descriptive text will be directly traceable forwards and backwards from a requirement or from an associated test. Traceability is used to manage change and provides a basis for test planning. It is a tool to ensure that the software process has been completed from initial definition to completion of a product. Use of an RTM provides a software quality checkpoint. It is however noted that RISA toolkit V&V status assessment will not use a traceability matrix but will only specify requirements for target software.

The requirements for RISA Toolkit will focus on higher technical maturity and capability or applicability of PRA. The following requirements are proposed for RISA Toolkit. The requirements could be changed/added based on type of the tool.

Development level

- RISA Toolkit should have highest development level that can be immediately used for industrial application. If the tool is still under development, the development goal should clearly indicate commercialization plan. Use of under-developed or lower technical maturity tool will decrease credibility of risk-informed analysis outcome.

Use of proven technology

- Some of newly developed tool uses cutting edge technology to build the software. However, such forefront technology may not always applicable to existing approaches. The candidate of RISA Toolkit should be applicable to the current U.S. nuclear power plants.

PRA capability/applicability

- Since the main goal of the risk-informed analysis is to apply PRA to the safety margin management, the RISA Toolkit should have its own PRA capability or applicability to an existing PRA tool.

Documentation

- RISA Toolkit should have a recognizable list of documents which user can access easily, including user manual, theory manual, development plan, QA plan, V&V plan and results, etc.

System requirements

- Users may use diverse computer and operating systems, and RISA Toolkit should be operated in different computer systems. Hence, the system requirements and related verification test result should be documented properly.

Easy installation

- RISA Toolkit should be installed easily without developer’s support. Facility of installation will be reviewed.

Graphic user interface (GUI)

- GUI will provide applicable information to users. Any type of graphical out-put generator will be listed.

Version control

- For every step of a software update, the version history of the tool should be maintained for users. Related documents and manuals should also be updated.

V&V history

- In high-level definition, verification is to check the tool is correctly built while validation means to confirm the outcome of verified code is correct. Since V&V result is one of most important parameters for the quality of code outcome, the history of V&V activity by developer and third party should be documented. This includes regression test list for code verification.

QA program

- Quality assurance program will provide standard of RISA Toolkit quality. This will include ASME Nuclear Quality Assurance (NQA-1) certification program to support nuclear industry for high quality products and services.

Web page

- Dedicated web page should be maintained in order to communicate with users.

User support

- The code developer should provide feedback on the problems reported from users.

Training program

- For easy deployment to industry, training program by developer or RISA Pathway will be a good solution.

License

- The developer should clearly indicate details of licensing.

2.2 Phenomena Identification and Ranking Technique (PIRT)

The PIRT (Phenomena Identification and Ranking Technique) is a method to systematically gather information from experts on a specific issue (e.g., NPP accident) and rank the importance of the information. The goal is to support decision-making such as determining which information should have high priority for research on that subject. The PIRT was first developed in the late 1980s and has been successfully applied in nuclear technology such as nuclear reactor safety analysis. This is also an essential sub-process for BEPU (Best Estimate Plus Uncertainty) analysis such as CSAU (Code Scaling, Applicability, and Uncertainty), the BEPU methodology acknowledged by the NRC. From a perspective of BEPU analysis, the importance of PIRT comes from the fact that the information obtained from the PIRT (i.e., relative importance of phenomena) is used to determine the code uncertainty input parameters and their realistic boundaries. In general, experts determine the relative importance of phenomena by considering their influence on the relevant plant safety metrics (e.g., peak cladding temperature).

For RISA Toolkit V&V status assessment, three level ranking methods will be applied to requirements set in Section 2.1 by RTM:

High: Dominate impactful requirement. Require high level of technical maturity.

Medium: Moderate impactful requirement. Require medium level of technical maturity.

Low: Small or no impactful requirement. Satisfy with basic level of technical maturity.

2.3 Technology Readiness Level (TRL)

Technology Readiness Levels (TRL) is a type of measurement system used to assess the maturity level of a particular technology. Each technology project is evaluated against the parameters for each technology level and is then assigned a TRL rating based on the projects progress. There are nine technology readiness levels as shown in Figure 2-2. TRL 1 is the lowest and TRL 9 is the highest. When a technology is at TRL 1, scientific research is at the beginning and those results are being translated into future research and development. TRL 2 occurs once the basic principles have been studied and practical applications can be applied to those initial findings. TRL 2 technology is very speculative, as there is little to no experimental proof of concept for the technology. When active research and design begin, a technology is elevated to TRL 3. Generally, both analytical and laboratory studies are required at this level to see if a technology is viable and ready to proceed further through the development process. Often during TRL 3, a proof-of-concept model is constructed.

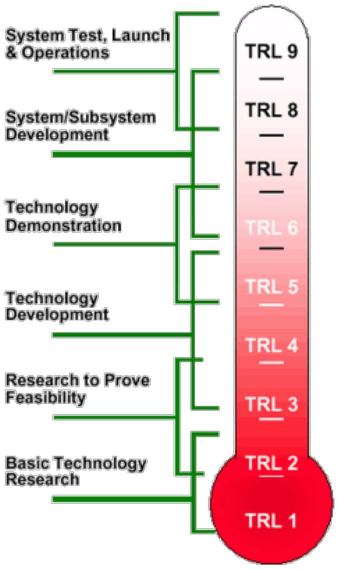


Figure 2-2 Schematic diagram of Technology Readiness Level (TRL)

Once the proof-of-concept technology is ready, the technology advances to TRL 4. During TRL 4, multiple component pieces are tested with one another. TRL 5 is a continuation of TRL 4, however, a technology that is at 5 is identified as a breadboard technology and must undergo more rigorous testing than technology that is only at TRL 4. Simulations should be run in environments that are as close to realistic as possible. Once the testing of TRL 5 is complete, a technology may advance to TRL 6. A TRL 6 technology has a fully functional prototype or representational model. TRL 7 technology requires that the working model or prototype be demonstrated in a space environment. TRL 8 technology has been tested and "flight qualified" and it's ready for implementation into an already existing technology or technology system. Once a technology has been "flight proven" during a successful mission, it can be called TRL 9. The U.S. Department of Energy proposed Technology Readiness Assessments (TRAs) and developing Technology Maturation Plans (TMPs) to assist individuals and teams that will be involved in conducting program and project management for the acquisition of capital assets [6]. TRAs and TMPs activities are a tool to assist in identifying technology risks and enable the correct quantification of scope, cost and schedule impacts in the project. Recent application of TRL in nuclear industry was to review maturity of advanced nuclear fuels and material development [7]. The purpose of the research was to evaluate existing technologies and identify R&D needs while developing advanced fuels and materials.

Similar to the RTM list, required parameters for critical issues were defined and measured its TRL. The level of TRL was also again defined based on the goal of advanced fuel and material development.

Based on general TRL, specific TRL for RISA Toolkit is proposed in Table 2-1. The RISA Pathway aims deploy higher TRL toolkit to industry, which will be TRL 7 or higher. For the RISA toolkit, the industry pilot demonstration project is the best practice to upgrade TRL 6 or lower toolkit to TRL 7 or higher.

Table 2-1 Description of Technology Readiness Level (TRL) for RISA Toolkit

Level of Tech. Development	Technology Readiness Level	TRL Definition	Application to RISA Toolkit
System Operation	TRL 9	Actual system operated over the full range of expected conditions.	The toolkit technology is in its final form and routinely use under the full range of industrial purpose.
System Development	TRL 8	Actual system completed and qualified through test and demonstration.	The toolkit has been proven to operate in its final form and under expected conditions. In almost all cases, this TRL represents the complete of R&D. Entire planned and proposed V&V of the toolkit are finalized by developer/user.
	TRL 7	Full-scale, similar (prototypical) system demonstrated in relevant environment	Full-scale demonstration of actual/prototypical technology/toolkit in relevant operation environment. Full-scale experiment and validation are performed. Development of the toolkit is virtually complete.
Technology Demonstration	TRL 6	Engineering/pilot-scale, similar (prototypical) system validation in relevant environment	Engineering-scale models or prototypes are tested in a relevant environment. Experiment and validation in engineering/pilot-scale environment including scaling effect testing, which can support operational system, design. This level represents completion of technology development for operational demonstration and prototype toolkit is ready for test. The prototype toolkit will have capability of performing all the functions that will be required from actual operational system. The operating environment for the testing should closely represent the actual operating environment. Major difference between TRL 5 is the scale-up from laboratory to engineering size.
Technology Development	TRL5	Laboratory scale, similar system validation in relevant environment	The basis of technology is fully integrated into the toolkit and ready for larger scale demonstration and validation. This level will include results from the laboratory scale test, analysis of the differences between the laboratory and actual operating system/environment, and analysis of experiments/demonstrations results for actual operating system/environment application. The major difference between TRL 4 and 5 is the increase of the toolkit fidelity and environment to the actual application. Verification is complete and the toolkit development level is close to prototypical.
	TRL 4	Component and/or system validation in laboratory environment	The basis of technology for toolkit is partly integrated and can be apply for component level demonstration. This is relatively "low fidelity" compared with the actual level of toolkit completion level. The expected maturity of this level includes the integrated experiments and validation, examination of scaling effect and actual application. Verification and regression test could be included. TRL 4-6 represents the bridge from scientific research to engineering. TRL 4 is the first step in determining whether the basic modeling will work in the toolkit.

Research to Prove Feasibility	TRL 3	Analytical and experimental critical function and/or characteristic proof of concept	Actual R&D is started for toolkit development. This includes analytical studies and laboratory-scale studies to validate the phenomena of separate technology. This level will have results of laboratory tests performed to measure parameters of interest and comparison to analytical predictions for critical toolkit functions. At TRL 3, actual R&D progresses to experiments and verifications. Validation could be done for part of the toolkit development, but system level validation is not yet initiated.
Basic Technology Research	TRL 2	Technology concept and/or application formulated	Progressed from TRL 1, technical options may be developed in TRL 2. However, still no activity was performed to prove assumptions and concept. Literature studies will outline the toolkit development concept. Most of activity in this level is analytical research and paper studies to understand goal of the R&D. Related experiments and V&V works could be designed during this level.
	TRL 1	Basic principles observed and reported	This is the lowest level of technology readiness. Scientific research begins to be translated into applied R&D. Available information includes published research or other references that identify the principles that underline the technology. No actual R&D started.

3. GUIDANCE FOR SOFTWARE V&V AND QA

This chapter summarizes guidance for software V&V and quality assurance program (QA) of US NRC and INL to give insight for managing quality of the RISA toolkit during its development and industry deployment. It is recommended that developers should take into account both guide to improve credibility and quality of RISA toolkit.

3.1 V&V related NRC Regulatory Guide 1.168

The US Nuclear Regulation Committee (NRC)'s regulatory guide 1.168 provides guidance to the staff of US NRC considers acceptable for use in complying with NRC regulations with respect to verification, validation, reviews, and audits for computer software used in safety systems of nuclear power plants. The regulation. Since the RISA Toolkit aims to use in safety related NPP system simulation, the toolkit developers as well as independent V&V performer and reviewer should consider this regulation to support future license acquirement. Originated from IEEE (Institute of Electrical and Electronics Engineers) Standard and became compliance with 10 CFR Part 50, the latest version of NRC RG 1.168 is revision 2, published in July 2013. This regulation was also mentioned in RELAP-7 Software Verification and Validation Plan [8].

Based on the 10 CFR Part 50 Appendix A and B, development of the software V&V, review and audit process should require development of appropriate quality assurance program. These requirements include:

- Criterion I, “Organization,” specifies that applicants must (1) assure that an appropriate quality assurance program is established and effectively executed, and (2) verify (for example, by checking, auditing, and inspection) that activities affecting safety-related functions have been correctly performed.
- Criterion II, “Quality Assurance Program,” requires, in part, that activities affecting quality be accomplished under suitably controlled conditions. Controlled conditions include the use of appropriate equipment, suitable environmental conditions for accomplishing the activity, and assurance that all prerequisites for the given activity have been satisfied. The criterion also requires that the program consider the need for verification of quality by inspection and test.
- Criterion III, “Design Control,” requires, in part, that design control measures provide for verifying or checking the adequacy of design.
- Criterion XI, “Test Control,” requires, in part, that a test program be established to assure that all testing required to demonstrate that structures, systems, and components will perform satisfactorily in service is identified and performed in accordance with written test procedures, which incorporate the requirements, and acceptance limits contained in applicable design documents.
- Criterion XVII, “Quality Assurance Records,” requires, in part, retention of records to furnish evidence of activities affecting quality. The records shall include identification and descriptions of actions taken in connection with any changes or design deficiencies noted.
- Criterion XVIII, “Audits,” requires, in part, that a comprehensive system of planned and periodic audits be carried out to verify compliance with all aspects of the quality assurance program and to determine the effectiveness of the program.

These six criteria should take into account for develop, upgrade, independent and developer V&V, maintenance and QA to ensure software integrity and reliability, which will improve product credibility during industry deployment of RISA Toolkit. More detail of each criterion could be found in NRC webpage link [9].

3.2 Software Quality Assurance program of INL

The Idaho National Laboratory has laboratory-wide procedure (LWP) application for all employees who propose, develop, modify, acquire, administer, maintain, or retire information technology (IT) assets, such as computer software, unless specifically excluded. This program includes software quality assurance (SQA) program and V&V procedures covering entire phases of software development activities. This set of documents and procedures will improve credibility and quality of the RISA toolkit.

The INL QA program requires all software applications need to have "Safety Software Determination" and "Quality Level Determination" in order to determine whether it is safety or non-safety and quality level (QL-1, QL-2 and QL-3). Once that is determined, the applicant can prepare related documents for quality level application as well as planning and performing V&V. This activity could be done either by subcontracted QA program specialist or developer own. The level of V&V also depends on quality level of the software.

There is no consistent definition for the term "quality level". The QL only serves as designator to identify the unmitigated risk or potential consequence level associated with the failure of an item or activity and facilitates communication for a common understanding of the rigor to be applied through the appropriate implementation procedures. The INL QA program proposes following guidance for QLD.

- Quality level 1 (QL-1) equates to high-unmitigated risk or high potential consequence level of failure.
- Quality level 2 (QL-2) equates to medium-unmitigated risk or medium potential consequence level of failure.
- Quality level 3 (QL-3) equates to low unmitigated risk.
- Quality level 4 (QL-4) equates to a no risk item or service.

The lists of applicable INL documents, templates, tools and forms are shown below. It is noted that following list of documents are not available to the public.

Program Description

- PDD-13610, *“Software Quality Assurance Program”*

Program Implementation Procedures

- LWP-13620, *“Managing Information Technology Assets”*
- LWP-20000-01, *“Conduct of Research Plan”*

Templates

- TEM-135, *“IT System Requirements Specification”*
- TEM-140, *“IT Software Design Description*
- TEM-141, *“Software Test Plan”*
- TEM-142, *“Software Quality Assurance Plan”*
- TEM-143, *“Configuration Management Plan”*
- TEM-162, *“IT Project Management Plan”*
- TEM-214, *“IT System RTM”*
- TEM-215, *“IT Asset Maintenance Plan”*
- TEM-216, *“Software Verification and Validation Plan*

Tools

- Enterprise Architecture Repository / Capabilities ＆ Technology Management (CTM)
- QLD and SSD applications
- Safety Software Listing in CTM

Forms

- Form 414.A89, *“Quality Level Determination Risk Analysis Method”*
- Form 414.A91, *“Quality Level Determination Safety Software Analysis Method”*
- Form 562.29, *“Software Product Review Report and Checklist”*
- Form 562.33, *“INL SQA Assessment Checklist”*
- Form 562.37, *“Safety Software Determination”*
- Form 562.38, *“Safety Software Training Documentation Form”*
- Form 562.40, *“IT Asset Management Responsibilities Checklist”*

Application of document is shown in Figure 3-1 (Appendix E of LWP- 13620, *“Managing Information Technology Assets”*). The RISA toolkit developer should first review PDD-13610 and LWP-13620 to understand procedure of improving quality and set quality level determination (QLD) and software V&V plan (SVVP) for future QA application. Either developer or third party such as QA specialist or independent V&V performer could do this activity.

Providing this set of documents will improve credibility of the RISA toolkit and eventually facilitate industry deployment of risk-informed tools and methods.

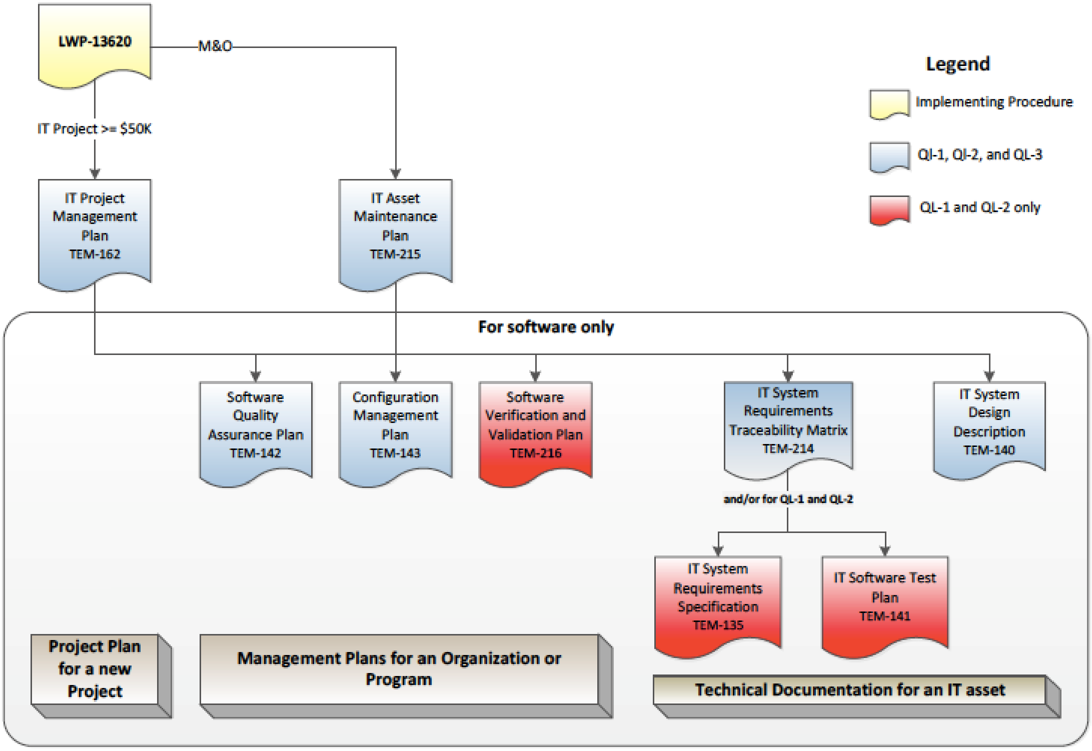


Figure 3-1 IT Asset document flow diagram

4. TECHNOLOGY MATURITY ASSESSMENT OF EMRALD

4.1 Overview

Event Modeling Risk Assessment using Linked Diagrams (EMRALD) is a dynamic PRA (Probabilistic Risk Assessment) tool developed in INL since 2005 to conduct dynamic PRA (DPRA) simulation. EMRALD was initially developed as an extension to the static PRA tool SAPAHIRE (Systems Analysis Programs for Hands-on Integrated Reliability Evaluations), which is also developed in INL since late 1980's. To use in DPRA, the time-dependent methodology is used in EMRALD for probabilistic assessment in time discretized event sequence calculations.

The early stage of development focused on three major areas: (1) develop discrete event modeling software library; (2) design simulation object model for dynamic simulation; and (3) build user friendly GUI. The first use of the EMRALD was to construct PRA framework for small-modular reactor (SMR) development [10]. In this research EMRALD was used as scenario controller unit between SAPHIRE and physics based tools and shown successful feature as a flexible object-oriented software model. Under LWRS program, this dynamic PRA approach with SAPHIRE was first applied for INL's Advanced Test Reactor case study [11]. The study set concept of risk-informed approach and applied to ATR safety related scenarios with conventional and dynamic PRA (DPRA) methodologies.

Supported by US DOE's LWRS Risk-Informed Safety Margin Characterization (RISMC) Pathway, EMRALD has been improved to conduct both PRA modeling logic control and risk analysis as well as development of user interface like conventional PRA modeling. During this period most of studies were focused on natural hazards such as flooding and earthquake related NPP accident scenarios [12]. Other applications performed at physical security, operator actions and flexible use of NPP equipment research (i.e. FLEX). EMRALD has been showing good coupling capabilities with various other tools such as SAPHIRE (static PRA), RAVEN (uncertainty quantification), RELAP5-3D (thermal hydraulics), NEUTRINO (flooding) and MASTODON (earthquake) [13]. Recent development also supported by US DOE's Technology Commercialization Fund (TCF) program and Small Business Innovation Research (SBIR) program for industry demonstration and extension of the application.

The main focuses of the EMRALD development are:

- Simplify modeling process by using corresponding traditional PRA method structure
- Support GUI for easy model build and visualize complex interactions
- Allow coupling with other analysis tools either one- or two-way communication
- Provide the sequence and timing of events for specified outcomes

EMRALD also includes traditional PRA modeling by using basic events, fault and event trees. These are all captured in a dynamic framework of state diagrams, which is displayed in a user-friendly GUI. Like other PRA codes, EMRALD uses concept of "diagram", which represents areas of a PRA model, control logic, accident scenario, etc. Each "diagram" consists of piece to define states, actions, events, and logic of the simulation, which also called as "dynamic components".

Current version of EMRALD has web-browser Chrome based GUI to build the model. The solver only works under Windows OS. User must acquire access permission from the EMRALD user group administrator through INL's safety portal (https://safety.inl.gov). More information is available at https://emrald.inl.gov.

4.2 Features

The main objective of the EMRALD development is to provide dynamic PRA modeling by avoiding complex software structure and steep initial learning curve use. The target features of the EMRALD are therefore (1) intuitive web-based GUI for easy use, (2) includes example PRA models and (3) open source communication framework for flexible coupling. EMRALD has three diagrams (component, system and plant diagrams), which are controlled by different dynamic components (state, action, event and logic tree). From GUI, user can edit properties, diagram, logic, and format. Drag-and-drop is available for model edit. To couple with other codes, EMRALD has an additional interface diagram for external simulation integration. This diagram allows couple with other code by network messaging protocol.

Current version of EMRALD shows static PRA model in default showing components and a scenario related to normal reactor coolant system and emergency core cooling system (ECCS) with related components such as valves, pumps, diesel generator, tank, etc. User can add new component and scenarios by right clicking left side menu.

EMRALD uses web-based technology to build the model. It has single page application and full GUI is available for entire feature. Most of operations allow click and drag-and-drop. Saved model and output files could be saved in local or network storage as a JSON file and can be edited by Windows text editor.

EMRALD UI is built with common web-based computer languages such as JavaScript, HTML5 (Hypertext Markup Language) and CSS3 (Cascading Style Sheets). The model input and output saved file uses JSON (JavaScript Object Notation) format, which has advantage of human-readable text structure. The software libraries used in EMRALD's UI are mxGraph, jQuery, jQuery-ui and AngularJS.

The main features of EMRALD are as follows.

4.2.1 Diagrams

Using diagram is common in various PRA codes. Each diagram contains components to be analyzed, calculation models and various condition and status of the simulation. EMRALD has three representative diagrams: component, system, and plant. Diagrams can handle from component to entire plant scale and can set multiple conditions and states during the simulation. The Boolean logic evaluation is available for both component and system diagram and could simplify the evaluation model by combine with component logic event.

Component

Component diagram could be made for each component of the nuclear power plant such as pump, tank, valve, etc. Basic PRA state, action and event could be set in one or multiple components. To include rare event in overall result, it is recommended to model time related piece traditional PRA case in EMRALD and apply to static PRA analysis. Component diagram can have one state active at a time. The Boolean logic could be applied.

System

System diagram uses logic tree to evaluate system, which could have multiple component diagrams, using leaf node concept to evaluate "Active" and "Failed". Evaluation will be continuous during the change of the component logic event evolution. System diagram can add "Component Logic Event" to evaluate logic diagram. The Boolean logic could be applied.

Plant

Plant diagram sets scenario of the simulation, which is similar to event tree of conventional PRA approach. This diagram should include start and terminate state and other essential states based on the scenario to be evaluated.

4.2.2 Dynamic Components

Four dynamic components are available to control each diagram.

States

States are a logical control flags of the component or system from initiation to termination of the simulation. User can set simulation condition of initial condition, run, stop and post-processing conditions. State component can control actions during the simulation and monitoring during the event.

Actions

Actions are the execution set-up control flags from starting of the simulation or specific trigger is initiated. This logic will be executed based on user defined transition indicator, variable change and executable applications. The action can also be started by specific signal from coupled external tool. User can customize action as needed.

Events

Events are monitoring feature based on specific criteria user provided. Multiple criteria and/or monitoring are possible for different monitoring point or variables during the simulation. Two different types of events are available for the monitoring: time and condition based events. The time-based events could be activated by time, failure rate or distributions. The condition-based events are monitored due to state change, component logic event, variable condition event and external code signal.

Logic Tree

Logic trees use standard Boolean gates to map and evaluate the behavior of components. These trees use the assigned Boolean value for the component state to solve for the top value of the tree. Logic trees are linked to the event type component logic tree to evaluate associate component state changes.

4.2.3 User Interface

EMRALD provides graphical user interface for entire software operation. The interface uses web browser with single page application. The list of diagrams and dynamic components are located at left side of the layout, which has drop down option for selecting the model or components. Properties could be edited by right clicking of the operating mouse. User can define name, description and type of the property, and can add additional user information as needed. Use can easily edit state, action and event of the diagram, component and logic tree for designated simulation scenario. Drag and drop is also available for most of interface.

4.2.4 Solver Engine

EMRALD is designed to perform dynamic simulation of PRA (DPRA) by discrete event modeling structure. EMRALD solver engine consists with three-phase discrete event simulation, which is also independent to time steps: (1) set event and execute immediate action for each existing and new state; (2) execute event action when conditional criteria is fulfilled; and (3) proceed to next event. The first phase is to analyze state events are correctly set then execute actions and sampling for the model. The final phase will proceed to next event in timeline.

Once web-based model editor sets model, user will download JSON format input file as well as set of solvers. The solver engine includes an executable file to open and run the input file. EMRALD allows the user to directly edit model through the solver engine for setup and test before the actual simulation. This solver engine interface shows entire information of model, simulation details, coupled code, log, etc. From this interface use can verify model and parameters, detail modification, check errors, monitor calculation as well as debugging of the simulation.

4.2.5 Coupling Capability

EMRALD enables couple with other codes by one or two-way communication method. For one-way coupling, EMRALD can execute software already installed in local computer. User can define directory and file of the executable, input data from the EMRALD script and can call results from coupled code to update calculation condition of the EMRALD. Two-way coupling enables coupled code affected from EMRALD calculation throughout the simulation. To control external code, EMRALD uses XMPP (Extensible Messaging and Presence Protocol) application program interface (API) to control external code, which is already included in EMRALD solve engine. User must include XMPP client package into the external coupled code to be controlled by EMRALD in- and output configurations. EMRALD also has standard messaging protocol for code coupling by using JSON format, which includes entire the information of actions and events.

4.3 Verification and Validation Status

Since EMRALD is non-physics application, no specific activities were performed for validation with existing experiments. During LWRS projects, EMRALD results have been benchmarked with SAPHIRE code result. The verification of the code could be done through unit and function test to confirm the code is working correctly.

4.3.1 Unit testing

EMRALD uses xUnit package to verify code is working correctly [14]. xUnit represents unit testing framework developed late 1990's. This package has been widely used to verify Java, C# and other .NET computer languages based-codes. Unit test is one of verification method to confirm the individual unit or module of the source code is correctly working together with associated control data, operational function and other procedures. The unit test has advantage of easy identification of error since this method uses automatic test of each part of the entire code. This will improve code stability and reduce debugging time. Like regression test, unit test also provides code verification once code has been updated or changed. Unit test method can make test case files from each part of the code to entire system, which gives bottom-up way of the verification and reduces uncertainties.

The unit test files to verify functions and optional result are in the TestingFile directory, which is easy to access by users. Users can specify depth and repeat times of the test as well as to verify source code change will not affect to the code results. Three functional tests are available for EMRALD:

- PRA actions test
- PRA events test
- External code coupling test

4.3.2 Benchmark study

Calculation result has been compared with other PRA application, SAPHIRE, as a benchmark study. Two studies have been done under external hazard research of LWRS program. First study was done for demonstration of external hazards analysis [15]. The study aims to demonstrate combined deterministic and probabilistic analysis of station black out (SBO) scenario due to seismic event with internal flooding in generic PWR. EMRALD was coupled with RELAP5-3D thermal-hydraulics analysis tool and NEUTRINO flooding simulation software for dynamic PRA approach and compared with SAPHIRE. The comparison result showed in good agreement for entire components.

Second study was done for extension of the seismic-flooding combined SBO scenario to cover uncertainty quantification by using RAVEN multi-purpose risk and data analysis code. EMRALD was used for dynamic PRA for best-estimate plus uncertainty (BEPU) analysis along with RELAP5-3D [13].

Similar to previous study, results from SAPHIRE were in good agreement with dynamic PRA with EMRALD. Figure 3-1 is result comparison of SAPHIRE and EMRALD for both research studies.

Additional benchmark studies have been done under different programs. However, these results are restricted due to copyright.

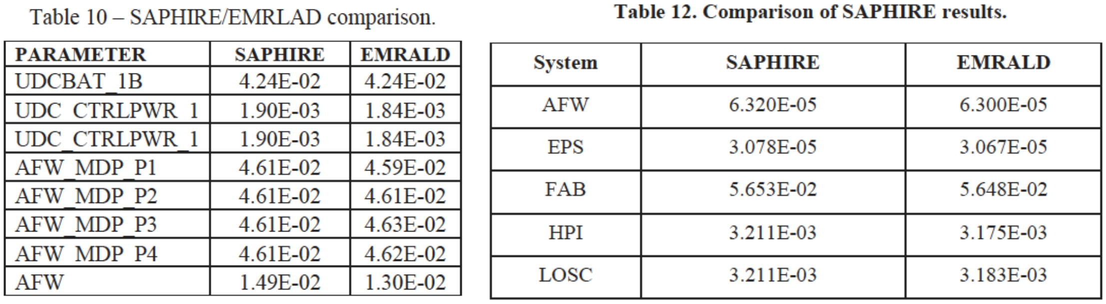


Table 10 – SAPHIRE/EMRLAD comparison.

PARAMETER	SAPHIRE	EMRALD
UDCBAT_1B	4.24E-02	4.24E-02
UDC_CTRLPWR_1	1.90E-03	1.84E-03
UDC_CTRLPWR_1	1.90E-03	1.84E-03
AFW_MDP_P1	4.61E-02	4.59E-02
AFW_MDP_P2	4.61E-02	4.61E-02
AFW_MDP_P3	4.61E-02	4.63E-02
AFW_MDP_P4	4.61E-02	4.62E-02
AFW	1.49E-02	1.30E-02

Table 12. Comparison of SAPHIRE results.

System	SAPHIRE	EMRALD
AFW	6.320E-05	6.300E-05
EPS	3.078E-05	3.067E-05
FAB	5.653E-02	5.648E-02
HPI	3.211E-03	3.175E-03
LOSC	3.211E-03	3.183E-03

Figure 4-1 Result comparison between SAPHIRE and EMRALD [15, 13]

4.4 Technical Maturity Assessment

Following information describes EMRALD technical maturity assessment results based on the requirements as the RISA Toolkit.

4.4.1 Development level

In mid-2000's, INL’s internal R&D funding grant supported initial development of EMRALD to provide dynamic simulation of PRA and to support SAPHIRE as discrete event simulator. The code was Windows based stand-alone software including GUI. It was used for SMR risk-informed study. Under LWRS program, EMRALD became web-browser based model building tool. From 2016, the code has been used mostly under LWRS program to perform dynamic PRA calculation for combined natural hazard analysis by coupling with RELAP5-3D and NEUTRINO code. Recently, the use of EMRALD has been expended to other DOE supported funding program such as TCF and SBIR.

First version was released in December 2015 (v0.0.0.1) then followed by v0.0.1.0 in December 2017. Latest version is v0.0.1.1. Code developers are using GitHub repository hosting services to update and maintain source code, but code system is not maintained nor version controlled by GitHub system. Currently, INL is hosting code repository, hence, user has to acquire permission to access the code.

Basic information as well as video tutorials is provided in https://emrald.inl.gov. Though no manual has been officially published, theoretical basis are well explained in publicly available reports and conference papers. The help page is also available at EMRALD application web page.

Most of key features are already developed: GUI along with visualization of time dependent system modeling, control and post logic to evaluate conditions with coupled code, and capability of coupling to be used in dynamic PRA. However, EMRALD does not provide statistical sampling models and is currently limited to Windows OS use.

4.4.2 Use of proven technology

EMRALD is generated with technically matured computer programming languages such as JavaScript, HTML5 and CSS3. These are commonly used in web-browser and embedded software. EMRALD uses JavaScript as main language for entire graphical interfaces. The fill-in forms are made with HTML5 and menu and navigation styling files are made with CSS3.

Following open source libraries are used for solver engine:

- mxGraph: Fully client side JavaScript diagramming library that uses SVG (Scalable Vector Graphics) and HTML for rendering. More information is available at https://github.com/jgraph/mxgraph.
- jQuery: Support query HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. More information is available at https://jquery.com.
- -jQuery-ui: Curate set of user interface interactions, effects, widgets, and themes built on top of the jQuery JavaScript library. More information is available at https://jqueryui.com. The software us
- -AngularJS: Handle data transport between the model and the presentation views for the client user interface data storage. More information is available at https://angularjs.org.

Input file has JSON format, which has compatibility and flexibility of use. For coupling, EMRALD uses Extensible Messaging and Presence Protocol (XMPP) communication protocol. Both methods are highly matured technology. PRA capability/applicability

4.4.3 PRA capability/applicability

EMRALD is developed as dynamic simulation capable PRA tool. It has discrete event model and external code coupling capability. However, EMRALD modeling is limited to normal and emergency core cooling system related scenario.

4.4.4 Documentation

INL has responsibility on maintaining and developing EMRALD. As of September 2020, no official manuals or V&V related documents have been published. Most of available documents are technical reports and articles for conference and communications such as newsletter. A conference paper title of "EMRALD, Dynamic PRA for Traditional Modeler" gives comprehensive information on the code structure and examples [16].

EMRALD development team provides tutorial video clips at official web site: http://emrald.inl.gov. For video tutorials are available: user interface introduction, running notepad, first run of code and basic component example. These tutorials provide basic information and example of use of the code.

Help page is also available in the EMRALD application web page. Main features of the code are explained including instructions of each diagram, logic tree, coupling external simulator, events, state and actions. As of September 2020, tutorial and FAQ are under preparation.

To use widely as RISA Toolkit, publication of official manual is recommended.

4.4.5 System requirements

EMRALD is compatible with Chrome web-browser installed in Microsoft Windows (64-bit Windows 7 and 10).

4.4.6 Easy installation

EMRALD uses web-browser based model builder. A set of solver engine needs to downloaded and unzip for use. User must open EMRALD_SIM.exe file to run the code. Since executable file is MS-DOS compatible executable, no installation is necessary for use.

4.4.7 Graphic user interface (GUI)

EMRALD has been developed as GUI application. Figure 4-2 shows EMRALD model editor captured from Chrome web browser. Logic tree could be also edited from GUI.

EMRALD_SIM.exe solver engine is shown in Figure 4-3. The result of calculation will be saved as TXT format.

EMRALD also saves calculation log file with JSON format, which user can extract value of variables at each calculation by additional script. User can use external plotting software to draw graph.

Drag and drop and right click are also available.

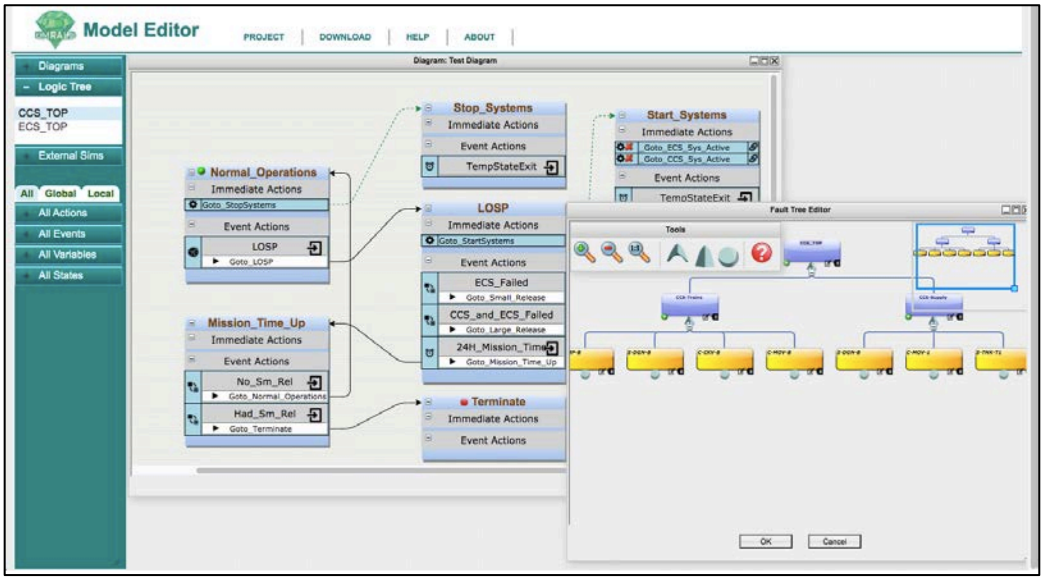
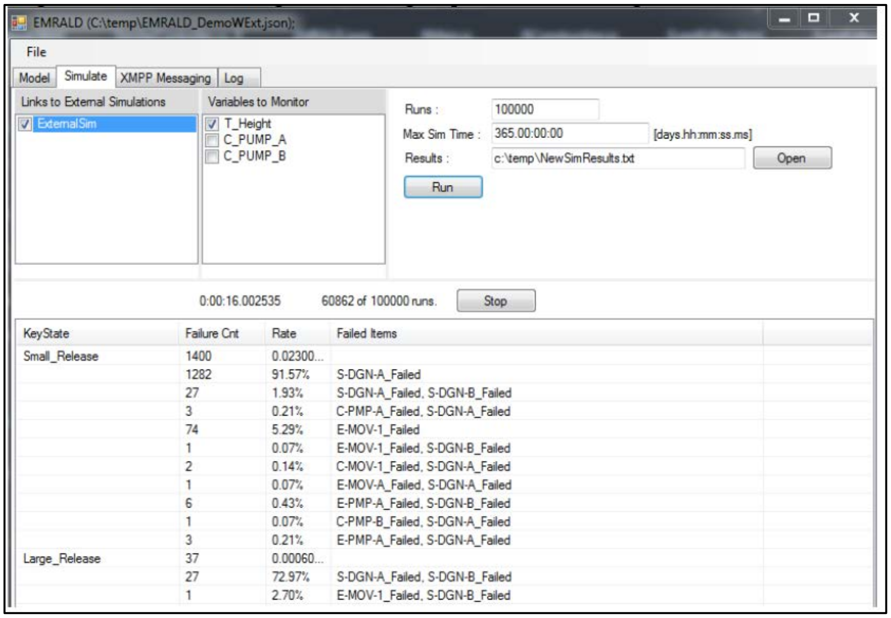


Figure 4-2 EMRALD Model Editor (diagram and fault tree editor)



Links to External Simulations

☑ ExternalSim

Variables to Monitor

☑ T_Height
☐ C_PUMP_A
☐ C_PUMP_B

Runs : 100000
Max Sim Time : 365.00:00:00 [days.hh:mm:ss.ms]
Results : c:\temp\NewSimResults.txt

0:00:16.002535 60862 of 100000 runs.

KeyState	Failure Cnt	Rate	Failed Items
Small_Release	1400	0.02300...	
	1282	91.57%	S-DGN-A_Failed
	27	1.93%	S-DGN-A_Failed, S-DGN-B_Failed
	3	0.21%	C-PMP-A_Failed, S-DGN-A_Failed
	74	5.29%	E-MOV-1_Failed
	1	0.07%	E-MOV-1_Failed, S-DGN-B_Failed
	2	0.14%	C-MOV-1_Failed, S-DGN-A_Failed
	1	0.07%	E-MOV-A_Failed, S-DGN-A_Failed
	6	0.43%	E-PMP-A_Failed, S-DGN-B_Failed
	1	0.07%	C-PMP-B_Failed, S-DGN-A_Failed
	3	0.21%	E-PMP-A_Failed, S-DGN-A_Failed
Large_Release	37	0.00060...	
	27	72.97%	S-DGN-A_Failed, S-DGN-B_Failed
	1	2.70%	E-MOV-1_Failed, S-DGN-B_Failed

Figure 4-3 EMRALD solver engine (EMRALD_SIM.exe)

4.4.8 Version control

As of September 2020, version v0.0.1.1 is latest edition of the EMRALD. Developer is maintaining changes of each version in GitHub development site. However, GitHub is not currently using for updating the code. Current version history is as shown in Table 4-1.

Table 4-1 EMRALD version history

Version	Release date	Major changes
0.0.0.1	Dec. 14, 2015	First stable version for demonstration
0.0.1.0	Dec. 15, 2017	Able to use JSON file and local save. Entire features of diagram and property edit are verified.
0.0.1.1	Sep. 15, 2020	Fix message generation and UI issues

The development team is planning to use GitHub repository to maintain automatic version control for each change. Version comparison study will be done once GitHub system is ready for EMRALD.

4.4.9 V&V activity/history

As of September 2020, no official V&V was performed for EMRALD. Verification was partly done by unit test and benchmark study with other PRA application, SAPHIRE.

4.4.10 QA program

As of September 2020, no official QA program was initiated.

4.4.11 Web page

Informative webpage https://emrald.inl.gov is available. This site includes basic information of the code, tutorial videos and contact information for permission to use.

4.4.12 User support

As of September 2020, no official user support program is operating. However, developer will support technical issues and training as necessary.

It is however sustainable resource is necessary for continuous support from INL EMRALD development team.

4.4.13 Training program

EMRALD development team organizes training program as needed.

4.4.14 License

EMRALD is not open source and publicly available. As of September 2020, no official license contract is necessary. However, user need to request permission to access INL's restricted web site for EMRAL model editor.

4.5 Conclusion and Remark

EMRALD has unique feature of web-based model edit and run in local personal computer, which provides install-free GUI option to users. The code performs PRA by either conventional or dynamic discrete event modeling method. It also has good coupling capability and demonstrated with RELAP5-3D thermal-hydraulics code and NEUTRINO flooding simulator. Technical maturity assessment result is shown in Table 4-2.

As a RISA Toolkit, EMRALD has been used in combined natural hazard analysis. From 2020, demand of EMRALD use has been steeply increased from various nuclear industries. In terms of RISA Pathway activity, following remarks are addressed.

- Official verification activity including regression test and documentation is necessary
- Official manual needs to be published when version 1 is released
- More demonstration study by coupling with physics-based codes (i.e. RELAP5-3D) will improve credibility
- User interface needs to be upgraded and fix minor bugs
- Consider INL's QA program to improve code credibility (Review INL's LWP-13620, *“Managing Information Technology Assets”*)

Table 4-2 EMRALD technology maturity assessment result

Requirements	**Importance**	**Description**	**Technology Readiness Level (TRL)**
Development level	High	Fundamental development including GUI for EMRALD technology is mostly finalized. Coupling with RELAP5-3D and NEUTRINO is fully demonstrated. Manuals and related documents are needed. Limited to Windows OS. More industrial use and verification is needed.	7
Use of proven technology	High	Mature technologies used in EMRALD. No specific issue was found.	9
PRA capability/applicability	High	EMRALD has been showing good result for classical and dynamic PRA by coupling with other codes. However, application is limited to core cooling system simulation.	7
Documentation	Medium	Manual or other supporting documents are not available. Web page provides tutorial videos for beginners. Need to complete help page.	3
System requirements	Low	Only works with Chrome web browser in Windows OS. No demand to be used in Linux system.	7
Easy installation	Medium	Model editor is web-based and execution is MS-DOS based file. No installation is needed.	9
Graphic user interface (GUI)	Medium	GUI is available for both model editing and execution. Output visualization is not available.	7
Version control	Medium	Official beta or alpha version is still under development. Developer's version is also available. No specific activity was done for version comparison study.	4
V&V activity/history	Medium	No validation activity is needed for EMRALD. Regression tests are needed for verification.	2
QA program	High	No QA program was initiated	1
Web page	High	EMRALD web page has basic information and tutorial videos. Regular update is necessary.	7
User support	High	Developer supports directly to the issue.	7
Training program	Medium	Development team organizes training program as needed.	8
License	Medium	No official license contract exists. User needs permission from developer.	3

5. TECHNOLOGY MATURITY ASSESSMENT OF HUNTER

5.1 Overview

Developed by INL, the HUNTER (Human Unimodel for Nuclear Technology to Enhance Reliability) is a framework to implement computational-based human reliability analysis (HRA) to nuclear power plant risk analysis. HUNTER aims to provide simple and dynamic modeling of the human cognitive model during risk analysis during nuclear power plant accident scenario analysis by coupling external applications such as RAVEN. During the analysis, HUNTER will produce a human error probability (HEP) and will give additional information such as manual action task duration to coupled codes to enhance risk analysis.

The development of HUNTER was initiated in 2014 supported by LWRS program. The initial approach was given to study advantage of simulation based HRA in order to implement in risk-informed analysis [17]. The concept of the HUNTER framework was proposed in 2015 by setting up a development roadmap of computational-based HRA [18]. In this study, the development team identified the goal and theoretical basis of HUNTER framework. An initial demonstration study was performed for HRA during flooding scenarios by coupling with RAVEN [19]. In 2016, development of the HUNTER framework was more progressed by RAVEN-HUNTER coupled application using dynamic complexity to solve performance-shaping factor (PSF), and demonstrated applicability to station black out (SBO) scenario. More demonstrations were done in various research activities including dynamic PRA of multi-unit plant [20].

The HUNTER framework aims to develop the following features:

- Use of simulation and modeling to integrate virtual operator model
- Dynamic modeling of human cognition and actions
- Incorporation of above elements into PRA framework

As of September 2020, the basic concept of HUNTER framework needs to be revised upon evolution of RAVEN code. The form of the code also needs to develop as shape of standalone computational software.

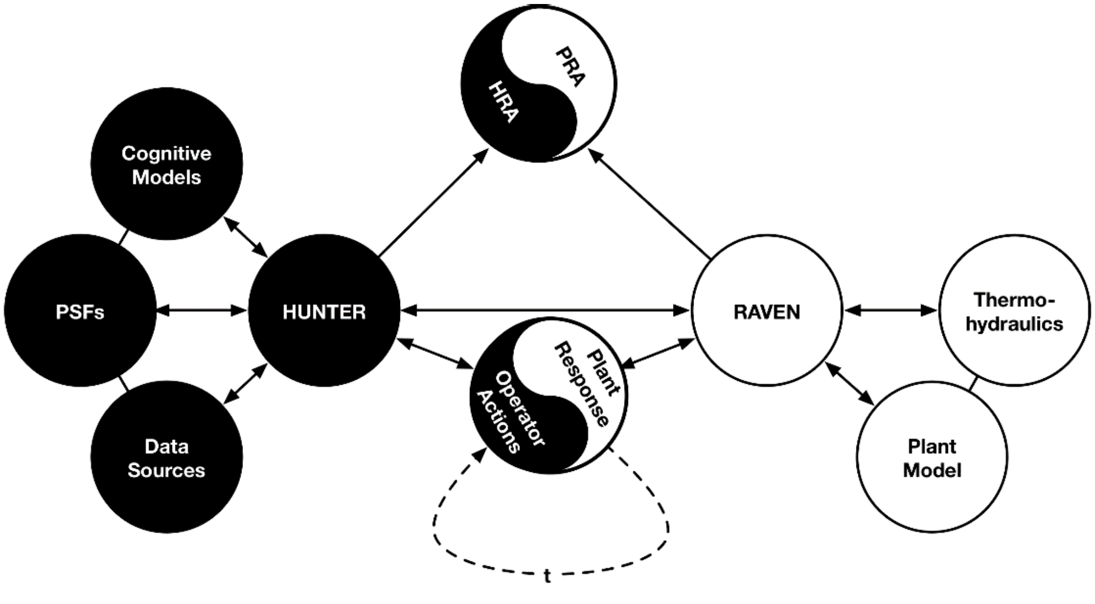


Figure 5-1 Conceptual design of HUNTER framework

5.2 Features

The main purpose of the HUNTER framework is to perform computational-based HRA for human behavior to reduce uncertainties during PRA incorporated with various accident scenario analyses. As shown in Figure 5-1, the HUNTER framework consists three main domains: cognitive model, PSF and data source. RAVEN code is one of the important applications to be coupled to allow HRA-PRA and operator action-plant response communications. RAVEN will also be coupled with physics-based thermal-hydraulics and other NPP modeling software to provide HRA-PRA results to plant simulation.

The HUNTER framework uses the GOMS (Goals, Operators, Methods and Selection rules) method to analyze human error probabilities (HEPs) and complexity as performance shaping factor (PSF) to evaluate dynamic complexity of the modeling. A nominal human error probability model was used to quantify HEP.

5.2.1 Human cognitive modeling

Developed early 1980s, the Goals, Operators, Methods, and Selection rules (GOMS) method originally aimed to analyze human information processing for human-computer interaction observation that describes user's cognitive structure on these four components. This method is widely used in computer system design since this method can quantify and predict human's reaction while using the proposed system. Definition of each component in this method is:

- Goals represent the high-level tasks the human seeks to complete
- Operators are the available actions the human can take
- Methods are the steps or sub-goals the human takes toward completing Goals
- Selection rules are the decisions the humans make.

This method has the advantage of using human factors to model by categorizing particular types of actions and can predict human behavior or task duration. Applying to the nuclear industry, the GOMS method has already been used for human factor analysis as GOMS-HRA. GOMS-HRA features a selection of task level primitives, representing the most basic action types by following operators:

- Actions: Performing required physical actions on the control boards or in the field
- Checking: Looking for required information on the control boards or in the field
- Retrieval: Obtaining required information on the control boards or in the field
- Instruction Communication: Producing verbal or written instructions or receiving verbal or written instructions
- Selection: Selecting or setting a value on the control boards or in the field
- Decisions: Deciding based on procedures or without available procedures

The GOMS-HRA model is integrated into the HUNTER framework. Each event of certain accident scenarios is decomposed into the sequence of task level primitives, which are used to calculate completion time and HEP values for each step of event.

5.2.2 Performance shaping factor (PSF) modeling

Traditional HRA uses sets of performance shaping factors (PSFs), which can define the context of the task. A PSF is a factor that influences human performance and related HEPs in HRA. However, this method has the limitation that same source of error could be represented and applied across multiple PSFs in one calculation, which may inflate the level of risk through double counting.

The computational-based HRA in the HUNTER framework aims to use dynamic modeling of complexity to reduce uncertainties of PSFs and to improve HRA flexibility. This approach allows analyzing more realistic accident scenarios by reshaping the event tree from dynamic simulation, and will remove potential errors or uncertainties. The dynamic modeling of complexity can also include influence from time spent during the task. During the development of HUNTER, technical challenges were identified: tracing the PSF selection procedure could be difficult; and manual scenario modeling could lead to error through analyst subjectivity. The HUNTER framework, therefore, proposed and demonstrated auto-populating of complexity and additional PSFs based on available scenario information.

5.2.3 Human Error Probability (HEP) quantification

One of main goal of the HUNTER framework is the automatic calculation of HEPs by dynamic simulation. Conventional HRA includes a probabilistic description for HEP quantification. The HUNTER framework uses nominal HEPs based on the error from each different task and action during the scenario. These nominal HEPs are intentionally formulated to describe generic human actions to support their application in many different tasks. The overall HEP will be then calculated by multiplying task-specific PSF multipliers and the nominal HEP value. The PSF is a dominant value on deciding reliability of human performance. Therefore, quantification of the task-specific overall HEP is the main value to decide human reliability error and assessment results. From the automatic calculation function of the HUNTER framework, the user can easily capture human error during the simulation.

HEP quantification is also greatly affected by the degree of complexity. The HUNTER framework is designed to consider change of complexity during the simulation and as the modeled event progresses, and the change occurs relative to the nominal HEP value.

Since the HUNTER framework uses the GOMS-HRA method for human cognitive modeling, the Technique for Human Error Rate Prediction (THERP) methodology is used for quantifying the operator HEP from each task. The THERP method uses template-matching technology, which is similar by selecting closest sub-tasks from an existing worksheet during analysis. From this method, nominal GOMS-HRA errors can be quantified. To quantify SPAR-H (Standardized Plant Analysis Risk-HRA) errors, the HUNTER framework uses the SPAR-H method to quantify HEPs from procedure-based actions.

5.3 Verification and Validation Status

The HUNTER framework is still in under development stage, and actual V&V activities are not being performed. Another issue of the HUNTER framework is that the form of the application needs to be improved as standalone software. The current version is still in the initial stage of code format.

The development roadmap is well planned from the beginning of framework planning: theoretical basis of computational-based HRA was fully reviewed and a comparison study was performed, various computational platforms were clearly reviewed, an in-depth study was performed for essential technical basis, technical gaps and research needs are well identified, and an initial demonstration was performed for flooding and station black out (SBO) scenario.

5.3.1 Verification of technical basis

Verification activities were conducted for the following existing and newly developed HUNTER framework technologies:

- Benefit of computational-based HRA compare to conventional static method
- Review of applicable HRA models in and out of nuclear industry
    - Technique for human error rate prediction (THERP)

    - Human cognitive reliability (HCR)
    - Goals-Operators-Methods-Selection Rules (GOMS)
    - Improved manpower personnel research integration tool (IMPRINT)
- Available computational platforms
    - Accident dynamic simulator (ADS)
    - Adaptive dynamic accident progression trees (ADAPT)
    - Monte Carlo and dynamic event trees (MCDET)
    - Risk analysis and virtual control environment (RAVEN)
- Probability quantification methods
    - Goals-Operators-Methods-Selection Rules (GOMS) based
    - Standardized plant analysis risk-HRA (SPAR-H) based
- Newly developed HUNTER framework technologies
    - GOMS-HRA method and quantification
    - Dynamic complexity modeling for performance shaping factor (PSF)
    - Coupling technology for using RAVEN

5.3.2 Identification of technology gaps and research needs

For each stage of HUNTER framework development, the identification of technology gaps and research needs have been clearly addressed by the development team and stated in publications. As of September 2020, the following issues were identified and to be developed as future activities:

- Develop HUNTER framework as a form of software and its interface
- Enhance dynamic HRA modeling capabilities
- Perform scenario-based demonstration and V&V activities
- Develop coupling capabilities with physics-based tools

5.3.3 Scenario-based demonstration study

Upon initial development of the HUNTER framework, the following scenario-based demonstrations were performed:

- Flooding scenario case study [INL/EXT-15-36741]
- Station black out case study [INL/EXT-16-39015]
- Multi-unit plant seismic induced station black out case study [reliability eng]

5.4 Technical Maturity Assessment

Following information describes the HUNTER framework technical maturity assessment results based on the requirements for the RISA Toolkit.

5.4.1 Development level

Under the LWRS program, the actual development of the HUNTER framework was done for 2014-2016 mostly focused on developing the technical basis of dynamic HRA. Development of theoretical background and modeling methodologies are mostly completed, and well documented. However, demonstration and V&V activities are very limited. Most of all the standalone code still needs to be developed. The framework uses RAVEN code as coupling and in/output data management application. Since RAVEN code has evolved compared to the HUNTER framework during the development period, existing coupling method needs to be reviewed.

In order to respond to recent increases in demand for the dynamic HRA, the development team needs to perform a comprehensive reorganization of the development plan and develop standalone code platform as soon as possible.

5.4.2 Use of proven technology

The HUNTER framework has been reviewed of available technologies and improved existing technologies to be used in dynamic HRA application: the GOMS cognitive method was updated as the GOMS-HRA model, the computational-based complexity simulation method was developed for PSFs, and advanced HEP quantification models were developed based on nominal error quantification methods.

5.4.3 PRA capability/applicability

The HUNTER framework is initially targeted to enhance PRA by adding HRA results. Once code is ready for use, it will also be applicable to various PRA tools.

5.4.4 Documentation

Most of available documents are technical reports under the LWRS program and related conference and journal articles. Theory and user manual and release notes should be prepared as code development progresses.

To use widely as part of the RISA Toolkit, publication of an official manual is recommended.

5.4.5 System requirements

No information available.

5.4.6 Easy installation

No information available.

5.4.7 Graphic user interface (GUI)

No information available.

5.4.8 Version control

No information available.

5.4.9 V&V activity/history

No information available.

5.4.10 QA program

No information available.

5.4.11 Web page

No information available.

5.4.12 User support

No information available.

5.4.13 Training program

No information available.

5.4.14 License

No information available.

5.5 Conclusion and Remark

The HUNTER framework has low technical maturity to be used in the RISA Toolkit immediately. However, technical basis of the framework is fully developed but needs effort to produce actual standalone software. The initial demonstration studies have shown potential benefits of dynamic HRA, which can also give high synergy along with dynamic PRA. Technical maturity assessment results are shown in Table 5-1.

As part of the RISA Toolkit, the HUNTER framework has been used in flooding, station black out and multi-unit seismic accident analysis. From 2020, the need for dynamic HRA is increasing as a part of risk-informed systems analysis. In terms of RISA Pathway activity, the following remarks need to be addressed:

- Immediate development of standalone software is necessary
- Review INL's software development guidance to prepare future QA and higher credibility
- Re-organize and review development roadmap to build standalone software
- Develop coupling module for physics-based RISA Toolkit (i.e. RELAP5-3D)
- Set of manuals and supporting documents is necessary for industrial deployment

Table 5-1 HUNTER framework technology maturity assessment result

Requirements	Importance	Description	Technology Readiness Level (TRL)
Development level	High	Technical bases are well studied and developed. However, still needs to develop standalone software.	3
Use of proven technology	High	Mature theories are used as basis. No information available for code itself.	5
PRA capability/applicability	High	HUNTER framework is designed as PRA use. Need more demonstration.	7
Documentation	Medium	No information	N/A
System requirements	Low	No information	N/A
Easy installation	Medium	No information	N/A
Graphic user interface (GUI)	Medium	No information	N/A
Version control	Medium	No information	N/A
V&V activity/history	Medium	No information	N/A
QA program	High	No information	N/A
Web page	High	No information	N/A
User support	High	No information	N/A
Training program	Medium	No information	N/A
License	Medium	No information	N/A

REFERENCES

[1] C. Smith, et al., Risk-Informed Systems Analysis (RISA) Pathway Technical Program Plan, INL/EXT-19-55721, Idaho National Laboratory, 2019

[2] Y-J Choi, Assessment of verification and validation status - RELAP5-3D and RAVEN, INL/EXT-19-56151, Idaho National Laboratory, 2019

[3] C. Smith, Y-J. Choi, J. Yoo, RELAP-7 Software Verification and Validation Plan Requirements Traceability Matrix (RTM) Part 1 – Physics and numerical methods, INL/EXT-15-36684, Idaho National Laboratory, 2015

[4] J. Yoo, Y-J Choi, Advanced Validation Risk-Informed Approach for the Flooding Tool Based Upon Smooth Particle Hydrodynamics - Validation and Development Status of NEUTRINO, INL/EXT-18-44976, Idaho National Laboratory, 2018

[5] http://www.neutrinodynamics.com/

[6] U.S. DOE, Technology Readiness Assessment Guide, DOE G 413.3-4, 2009

[7] J. Carmack, Technology Readiness Levels for Advanced Nuclear Fuels and Materials Development, INL/EXT-14-31243, Idaho National Laboratory, 2014

[8] J. Yoo, Y-J Choi, Additional Model Datasets and Results to Accelerate the Verification and Validation of RELAP7, INL/EXT-16-40577, Idaho National Laboratory, 2016

[8] The RELAP5-3D Code Development Team, RELAP5-3D Code Manual Volume I: Code Structure, System Models and Solution Methods, INL/MIS-15-36723, Rev 4.4, Idaho National Laboratory, 2018

[9] Verification, validation, reviews, and audits for digital computer software used in safety systems of nuclear power plants, Regulatory Guide 1.168, Revision 2, 2013.

[10] C. Smith, et al., A Framework to Expand and Advance Probabilistic Risk Assessment to Support Small Modular Reactors, INL/EXT-12-24345, Idaho National Laboratory, 2012

[11] C. Smith, et al., Risk Informed Safety Margin Characterization (RISMC) - Advanced Test Reactor Demonstration Case Study, INL/EXT-12-27015, Idaho National Laboratory, 2012

[12] S. Prescott, C. Smith, R. Sampath, Incorporating dynamic 3D simulation into PRA, INL/CON-14-33680, Idaho National Laboratory, 2015

[13] S. Prescott, et al., Risk-informed external hazards analysis for seismic and flooding phenomena for a generic PWR, INL/EXT-17-42666, Idaho National Laboratory, 2017

[14] http://xunit.net

[15] S. Prescott, et al., Demonstration of external hazards analysis, INL/EXT-16-39353, Idaho National laboratory, 2016

[16] S. Prescott, C. Smith, L. Vang, EMRALD, Dynamic PRA for Traditional Modeler, PSAM2018

[17] R. Boring, et al., Simulation and non-simulation based human reliability analysis approaches, INL/EXT-14-33903, Idaho National Laboratory, 2014

[18] R. Boring, et al., A research roadmap for computation-based human reliability analysis, INL/EXT-15-36051, Idaho National Laboratory, 2015

[19] R. Boring, et al., Proof-of-Concept Demonstrations for Computation-Based Human Reliability Analysis: Modeling Operator Performance During Flooding Scenarios, INL/EXT-15-36741, Idaho National Laboratory, 2015

[20] R. Boring, et al., Multi-unit dynamic PRA, Journal of Reliability Engineering and System Safety 185, 303-317, 2019

APPENDIX A. RISA TOOLKIT TECHNOLOGY MATURITY ASSESSMENT MATRIX

Requirements	**Importance**	**Description**	**Technology Readiness Level (TRL)**
Development level			
Use of proven technology			
PRA capability/applicability			
Documentation			
System requirements			
Easy installation			
Graphic user interface (GUI)			
Version control			
V&V history			
QA program			
Web page			
User support			
Training program			
License			