MANUAL

RAVEN-RTM
Revision 0
Printed October 2018

RAVEN Requirements Traceability Matrix

Andrea Alfonsi

Prepared by
Idaho National Laboratory
Idaho Falls, Idaho 83415

The Idaho National Laboratory is a multiprogram laboratory operated by
Battelle Energy Alliance for the United States Department of Energy
under DOE Idaho Operations Office. Contract DE-AC07-05ID14517.

Issued by the Idaho National Laboratory, operated for the United States Department of Energy by Battelle Energy Alliance.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

RAVEN-RTM
Revision 0
Printed October 2018

RAVEN Requirements Traceability Matrix

Andrea Alfonsi

Contents

1 Introduction

RAVEN is a flexible and multi-purpose uncertainty quantification (UQ), regression analysis, probabilistic risk assessment (PRA), data analysis and model optimization software. Its broad spectrum of application determined the need of an integrated design (see RAVEN SDD document for details) of the software aimed to integrate multiple requirements.
This document is aimed to report the traceability matrix between software requirements (see RAVEN SRS) and requirement tests (tests that testify the software is compliant with respect its own requirements).

1.1 Dependencies and Limitations

The software should be designed with the fewest possible constraints. Ideally the software should run on a wide variety of evolving hardware, so it should follow well-adopted standards and guidelines. The software should run on any POSIX compliant system (including Windows POSIX emulators such as MinGW). The software will also make use of artificial intelligence and numerical libraries that run on POSIX systems as well. The main interface for the software will be command line based with no assumptions requiring advanced terminal capabilities such as coloring and line control.
In order to be functional, RAVEN depends on the following software/libraries.

- h5py-2.7.1
- numpy-1.12.1
- scipy-1.1.0
- scikit-learn-0.19.1
- pandas-0.20.3
- xarray-0.10.3
- netcdf4-1.4.0
- matplotlib-2.1.1
- statsmodels-0.8.0
- python-2.7
- hdf5-1.8.18
- swig

- pylint
- coverage
- lxml
- psutil
- pyside
- pillow

2 References

- ASME NQA 1 2008 with the NQA-1a-2009 addenda, "Quality Assurance Requirements for Nuclear Facility Applications," First Edition, August 31, 2009.
- ISO/IEC/IEEE 24765:2010(E), "Systems and software engineering Vocabulary," First Edition, December 15, 2010.
- LWP 13620, "Managing Information Technology Assets"

3 Definitions and Acronyms

3.1 Definitions

- **Baseline.** A specification or product (e.g., project plan, maintenance and operations [M&O] plan, requirements, or design) that has been formally reviewed and agreed upon, that thereafter serves as the basis for use and further development, and that can be changed only by using an approved change control process. [ASME NQA-1-2008 with the NQA-1a-2009 addenda edited]
- **Validation.** Confirmation, through the provision of objective evidence (e.g., acceptance test), that the requirements for a specific intended use or application have been fulfilled. [ISO/IEC/IEEE 24765:2010(E) edited]
- **Verification.**
  - The process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase.
  - Formal proof of program correctness (e.g., requirements, design, implementation reviews, system tests). [ISO/IEC/IEEE 24765:2010(E) edited]

3.2 Acronyms

API Application Programming Interfaces

ASME American Society of Mechanical Engineers

CDF Comulative Distribution Functions

DET Dynamic Event Tree

DOE Department of Energy

HDF5 Hierarchical Data Format (5)

LWRS Light Water Reactor Sustainability

NEAMS Nuclear Energy Advanced Modeling and Simulation

NHES Nuclear-Renewable Hybrid Energy Systems

INL Idaho National Laboratory

IT Information Technology

M&O Maintenance and Operations

MC Monte Carlo

MOOSE Multiphysics Object Oriented Simulation Environment

NQA Nuclear Quality Assurance

POSIX Portable Operating System Interface

PDF Probability Distribution (Density) Functions

PP Post-Processor

PRA Probabilistic Risk Assessment

QA Quality Assurance

RAVEN Risk Analysis and Virtual ENviroment

ROM Reduced Order Model

SDD System Design Description

XML eXtensible Markup Language

4 Pre-test Instructions/Environment/Setup

The test of the requirements are performed automatically through the CIS (Continuous Integration System) for each CR (Change Request). The tests are performed on each supported Operative System (see [**?**]).

5 RAVEN:SYSTEM REQUIREMENTS

5.1 Requirements Traceability Matrix

This section contains all of the requirements, requirements' description, and requirement test cases. The requirement tests are automatically tested for each CR (Change Request) by the CIS (Continuous Integration System).

5.1.1 Minimum Requirements

Requirment ID	**Requirment Description**	**Test(s)**
R-M-1	Computer: Any POSIX (and POSIX-like) system	1)"RAVEN User Manual", INL/EXT-15-34123 2)Continous Integration System
R-M-2	RAM: 2 GB per core execution (depending on the type of analysis and data genarated)	1)"RAVEN User Manual", INL/EXT-15-34123 2)Continous Integration System
R-M-3	Disk: 10 GB (size depending on the type of analysis and data generated)	1)"RAVEN User Manual", INL/EXT-15-34123 2)Continous Integration System
R-M-4	Compilers: GCC, Clang, or Intel	1)"RAVEN User Manual", INL/EXT-15-34123 2)Continous Integration System
R-M-5	Language: Python 2.7	1)"RAVEN User Manual", INL/EXT-15-34123 2)Continous Integration System
R-M-6	Version Control: Git	1)"RAVEN User Manual", INL/EXT-15-34123 2)Continous Integration System

Minimum Requirements

5.1.2 Functional Requirements

Requirment ID	**Requirment Description**	**Test(s)**

R-F-1	RAVEN shall allow support for user-defined instructions for controlling the execution stages of a simulation.	1)/raven/tests/framework/test_rom_trainer.xml 2)/raven/tests/framework/test_random.xml
R-F-2	RAVEN shall allow for user-defined resource allocation for driving external applications.	1)/raven/tests/framework/CodeInterfaceTests/generic_parallel.xml
R-F-3	RAVEN shall support a programmatic method for building up and/or downloading the necessary compiled objects/dependencies necessary for a simulation.	1)RAVEN User Manual, INL/EXT-15-34123
R-F-4	RAVEN shall provide the ability to resume a previous simulation using data generated and exported by RAVEN itself.	1)/raven/tests/framework/Samplers/Restart/test_restart_MC.xml 2)/raven/tests/framework/Samplers/Restart/test_restart_csv.xml 3)/raven/tests/framework/Samplers/Restart/test_restart_constant.xml
R-F-5	RAVEN shall allow for user-defined output types for simulation data.	1)/raven/tests/framework/test_output.xml 2)/raven/tests/framework/ROM/TimeSeries/DMD/test_traditional_dmd.xml
R-F-6	RAVEN shall allow for a standardized method for importing simulation data not previosly generated by the system itself.	1)/raven/tests/framework/test_output.xml 2)/raven/tests/framework/test_iostep_load.xml 3)/raven/tests/framework/Databases/test_load_and_push_reusing_same_hdf5.xml

Framework, I/O, Execution Control

5.1.3 Usability Requirements

Requirment ID	Requirment Description	Test(s)
R-RE-1	RAVEN shall support 1-Dimensional probability distributions including generating random numbers from them.	1)/raven/tests/framework/unit_tests/Distributions/TestDistributions.py
R-RE-2	RAVEN shall support N-Dimensional probability distributions. It shall support multivariate normal distributions and distributions defined by tabular data.	1)/raven/tests/framework/test_simple_ND_external_MC.xml
R-RE-3	RAVEN shall support a variety of samplers that use probability distributions to sample the input space.	1)/raven/tests/framework/test_Grid_Sampler.xml

Risk Evaluation

Requirment ID	Requirment Description	Test(s)
R-RA-1	RAVEN shall support adaptive sampling that use already gathered samples to determine where to locate new samples.	1)/raven/tests/framework/PostProcessors/LimitSurface/test_LimitSurface.xml
R-RA-2	RAVEN shall support importing/exporting data in CSV format.	1)/raven/tests/framework/test_iostep_load.xml
R-RA-3	RAVEN shall support generating plots from the data it generates.	1)/raven/tests/framework/test_output.xml

R-RA-4	RAVEN shall be able to generate Reduced Order Models from its data and use them to predict responses from a system.	1)/raven/tests/framework/test_rom_trainer.xml
R-RA-5	RAVEN shall be able to perform basic statistical analysis of generated data.	1)/raven/tests/framework/PostProcessors/BasicStatistics/test_BasicStatistics.xml
R-RA-6	RAVEN shall be able to perform advanced post processing of generated data, using data mining methodologies.	1)/raven/tests/framework/PostProcessors/DataMiningPostProcessor/DimensionalityReduction/test_dataMiningExactPCA.xml
R-RA-7	RAVEN shall be able to compute probability of failure based on generated data and goal functions	1)/raven/tests/framework/PostProcessors/LimitSurface/test_LimitSurface.xml 2)/raven/tests/framework/-PostProcessors/LimitSurface/test_LimitSurface_and_integral.xml

Risk Analysis

Requirment ID	**Requirment Description**	**Test(s)**
R-RM-1	RAVEN shall be able to choose the values of a set of input parameters that minimize/-maximize a goal function that depends on system output figure of merits and input parameters.	1)/raven/tests/framework/Optimizers/beale.xml

Risk Mitigation

5.1.4 Performance Requirements

Requirment ID	Requirment Description	Test(s)
R-IS-1	RAVEN shall be able to parallelize running external codes.	1)/raven/tests/framework/CodeInterfaceTests/test_LHS_Sampler_Bison_parallel.xml
R-IS-2	RAVEN shall be able to run external codes by supplying them with the needed input files and collecting the output data.	1)/raven/tests/framework/test_simple.xml
R-IS-3	RAVEN shall support storing and retrieving data in a HDF5 database.	1)/raven/tests/framework/Databases/test_2steps_same_db.xml
R-IS-4	RAVEN shall be able to provide data to a user provided python function, and retrieve the data from that.	1)/raven/tests/framework/test_Lorentz.xml
R-IS-5	RAVEN shall be able to perform various calculation tasks (simulation and post processing), and transfer data to the next task.	1)/raven/tests/framework/test_calc_and_transfer.xml
R-IS-6	RAVEN shall be able to run external codes in parallel on shared memory machines.	1)/raven/tests/framework/test_bison_mc_simple.xml 2)/raven/tests/framework/CodeInterfaceTests/test_generic_interface.xml 3)/raven/tests/framework/CodeInterfaceTests/test_generic_interface_custom_out_file.xml
R-IS-7	RAVEN shall be able to run external codes in parallel on distributed memory machines.	1)raven/cluster_tests/test_mpi.xml 2)raven/cluster_tests/test_mpiqsub_local.xml 3)raven/cluster_tests/test_pbs.xml

R-IS-8	RAVEN shall be able to run internal models in parallel on shared memory machines.	1)/raven/tests/framework/InternalParallelTests/test_internal_parallel_ROM_scikit.xml 2)/raven/tests/framework/InternalParallelTests/test_internal_parallel_extModel.xml 3)/raven/tests/framework/InternalParallelTests/test_internal_parallel_PP_LS.xml
R-IS-9	RAVEN shall be able to run internal models in parallel on distributed memory machines.	1)raven/cluster_tests/InternalParallel/test_internal_parallel_extModel.xml 2)raven/cluster_tests/InternalParallel/test_internal_parallel_PP_LS.xml 3)raven/cluster_tests/InternalParallel/test_internal_parallel_ROM_scikit.xm

Infrastructure Support

5.1.5 System Interfaces

Requirment ID	**Requirment Description**	**Test(s)**
R-SI-1	RAVEN shall be able to be coupled with external applications via input files.	1)/raven/tests/framework/CodeInterfaceTests/RAVEN/rom.xml 2)/raven/tests/framework/CodeInterfaceTests/RELAP5/test_relap5_code_interface.xml
R-SI-2	RAVEN shall be able to be coupled with external applications via Python API.	1)/raven/plugins/ExamplePlugin/tests/test_example_plugin.xml 2)/raven/plugins/ExamplePlugin/tests/test_raven_running_raven_plugin.xml 3)/raven/tests/plugins/ExamplePlugin/test_example_plugin.xml 4)/raven/tests/plugins/ExamplePlugin/test_raven_running_raven_plugin.xml

Interface with external applications


INL
Idaho National Laboratory