|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Laboratory-wide | Template |  | eCR Number: 619232 |  |
| Manual: 13B - Quality Assurance Program Implementing Documents | | | | |

|  |
| --- |
| Document ID: SDD-513  Revision ID: 0  Effective Date: 10/01/2018 |
| IT System Design Description |
|  |
|  |
|  |
|  |
| RAVEN and RAVEN Plug-ins Software Design Description (SDD) |

|  |  |  |
| --- | --- | --- |
|  | | |
| RAVEN and RAVEN Plug-ins Software Design Description (SDD) | | |
| SDD-513 | | |
|  | | |
| **Prepared by:** |  | |
| [Required. For EDMS controlled documents, reference the eCR.] |  |  |
| Technical Lead |  | Date |
| **Reviewed by:** |  | |
| [Required for QL-1 and QL-2. For EDMS controlled documents, reference the eCR.] |  |  |
| Independent Reviewer/Review Team |  | Date |
| **Approved by:** |  | |
| [Required. For EDMS controlled documents, reference the eCR.] |  |  |
| IT Asset Owner |  | Date |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Applicability: | System Design Description |  | eCR Number: |  |
| Manual: | | | | |

CONTENTS

[1. INTRODUCTION 3](#_Toc525138300)

[1.1 Initial Remarks 3](#_Toc525138301)

[1.2 System Purpose 3](#_Toc525138302)

[1.3 System Scope 3](#_Toc525138303)

[1.4 Dependencies and Limitations 4](#_Toc525138304)

[2. REFERENCES 4](#_Toc525138305)

# INTRODUCTION

## Initial Remarks

The RAVEN software and its supported plug-ins are deployed using and Agile development process (see def.). Even if the core software design structure will be unlikely changed, the detailed RAVEN Software Design document is maintained in the GitHub (see def.) or GitLab (see def.) repositories. This approach allows for the expansion of the design description in case new components/modules are added during the deployment/development process.

The RAVEN (or its supported plug-ins) Software Design document is stored in *“<project\_folder>/doc/sqa/sdd/”* (where *<project\_folder>* can be either “raven” or the plug-in name) and can be generated launching the command “make”.

In this document a short and general description of the software design is reported.

## System Purpose

RAVEN is a flexible and multi-purpose uncertainty quantification, regression analysis, probabilistic risk assessment, data analysis and model optimization software. 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 system is modeled by third party software (e.g., RELAP5-3D, MAAP5, BISON, etc.) and accessible to RAVEN either directly (software coupling) or indirectly (via input/output files). 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.

## System Scope

RAVEN’s scope is to provide a set of capabilities to build analysis flows based on UQ, PRA, Optimization and Data Analysis techniques. The main objective of the software is to help the engineer/user to:

* identify the best design (on any physics/model), its safety and confidence;
* asses risk and procedures to better mitigate it
* identify main drivers/events to act on for reducing impact/consequences of anomalous dynamic behaviors of the system under analysis

More details about the scope of the software is reported in the GitHub-maintained RAVEN Software Design document.

## 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.

No evident limitations are envisioned for the RAVEN software design and its further expansions.

# 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,” FirstEdition, December 15, 2010.

LWP‑13620, “IT Asset Management,” Rev. 16, December 23, 2013.

1. Alfonsi et al., “RAVEN Software Design Document”, RAVEN-SDD.

Page number code—do not delete or use this marker for anything else!