Project Title: XYZ

Software Assurance Classification Report

**Prepared by:**

|  |  |
| --- | --- |
| **Name** | **Date** |
| Leslie J. Johnson, NASA LaRC  Software Assurance | 7/11/2013 |

**NASA Langley Research Center**

**Hampton, VA 23681**

**Revision and History Page**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision**  **No.** | **Change**  **No.** | **Description** | **Release**  **Date** |
|  |  | Initial Release | 7/11/2013 |

Table of Contents

SECTION PAGE

1. INTRODUCTION 4

1.1 Background 4

2. REFERENCE Documents 4

3. SUMMARY 4

3.1 Software Classification 4

3.2 Software Safety 5

3.3 Software Assurance Effort 5

Appendix A: ACRONYMS 6

# INTRODUCTION

This report contains the software assurance classification assessment which identifies and evaluates the characteristics of software in determining the software’s classification and level of software assurance to be applied to a Project.

## Background

The purpose of this code is to post-process data acquired in a wind tunnel test from an array of microphones.  Currently, the code is planned for deployment exclusively in the Now Flow Facility (NFF) in building 700, although if the code is found useful enough it may be used in other measurement facilities at a later date.

The software package XYZ takes pressure data which has been acquired by the facility data acquisition system and performs statistical pre-processing on the data, before generating maps of estimates and strengths sources on the model.  This is similar to the previous code in that regard, although this upcoming code is smaller in scale, with the intent of providing fast, initial results, rather than the most accurate post-processing.  It will operate on a single workstation, spending several minutes post-processing the data.  It will have no interaction with the facility operation, and simply downloads the data from the facility data storage system after acquisition is complete.

It is not a real-time code for use in the facility.

# REFERENCE Documents

The following documents were used or referenced in the development of this report:

|  |  |
| --- | --- |
| **Document No.** | **Document Title** |
| NPR 7150.2A | NASA Software Engineering Requirements |
| NASA-STD-8739.8 | NASA Software Assurance Standard |
| NASA-STD-8719.13B | NASA Software Safety Standard |
| LAPD 5300.1 | Program/Product Assurance |
| LPR 7150.2 | LaRC Software Engineering Requirements |
| LPR 5300.1 | Product Assurance Plan |
| LMS-CP-4754 | Software Assurance (SA) for Development and Acquisition |

# SUMMARY

The following paragraphs summarize the results and describe the details used to determine the software classification assessment for this report.

## Software Classification

According to LPR 7150.2, the software components for this Project are classified as Class D - Basic Science/Engineering Design and Research and Technology Software which is defined as:

*Basic Science/Engineering Design:*

*1. Ground software that performs secondary science data analysis, or*

*2. ground software tools that support engineering development, or*

*3. ground software used in testing other Class D software systems, or*

*4. ground software tools that support mission planning or formulation, or*

*5. ground software that operates a research, development, test, or evaluation laboratory (i.e., not a Major Engineering/Research Facility), or*

*6. ground software that provides decision support for non-mission critical situations.*

As such, the Project shall following the instructions and complete the compliance matrix in LMS-CP-7150.5, *Class D Software*, which applies to all Class D software that is not safety-critical.

## Software Safety

The Software Safety Litmus Test below is applied to all projects with software to determine if the software is safety-critical. If the software is determined to be safety-critical, then the project must adhere to the NASA-STD-8719.13, NASA Software Safety Standard.

A software component is considered safety-critical if it meets **any** of the following criteria:

|  |  |
| --- | --- |
| **Criteria:** | **Software**  **components** |
| 1. Resides in a safety-critical system (as determined by a hazard analysis) **AND** at least one of the following apply: | No |
| 1. Causes or contributes to a hazard |  |
| 1. Provides control or mitigation for hazards |  |
| 1. Controls safety-critical functions |  |
| 1. Processes safety-critical commands or data |  |
| 1. Detects and reports, or takes corrective action, if the system reaches a specific hazardous state |  |
| 1. Mitigates damage if a hazard occurs |  |
| 1. Resides on the same system (processor) as safety-critical software |  |
| 1. Processes data or analyzes trends that lead directly to safety decisions (e.g., determining when to turn power off to a wind tunnel to prevent system destruction) | No |
| 1. Provides full or partial verification or validation of safety-critical systems, including hardware or software subsystems. | No |

The software components in this Project do not reside in a safety-critical system; process data or analyze trends that lead directly to safety decisions or provide full or partial verification or validation of safety-critical systems.

The LaRC Safety and Mission Assurance Office have determined that the software components in this Project are not safety-critical.

## Software Assurance Effort

The software assurance effort is based on the software class and impacts from potential failure. In accordance with LMS-CP-4754 the Mission Assurance Branch software assurance personnel will perform spot checks on selected Class D software developments at LaRC to ensure compliance with LMS-CP-7150.5.

# Appendix A: ACRONYMS

|  |  |
| --- | --- |
| CP | Center Process |
| LaRC | Langley Research Center |
| LAPD | Langley Policy and Directives |
| LMS | Langley Management System |
| LPR | Langley Procedural Requirements |
| NASA | National Aeronautics and Space Administration |
| NPR | NASA Procedural Requirement |
| SA | Software Assurance |
| STD | Standard |