policy NEWS.

TEAM ALIGNS SOFTWARE SAFETY STANDARD WITH TODAY'S BEST PRACTICES

MAY 7, 2013

Revisions to the NASA Software Safety Standard update and streamline the requirements for safety-critical software without changing the basic software safety principles laid down through 15 years of experience.

A cross-agency team revised NASA-STD-8719.13C Software Safety Standard, and the changes took effect on May 7, 2013. This document pertains only to safety-critical software. The revisions bring the standard up-to-date with best practices in software safety design, analysis and development.

WHAT'S NEW.

The following are highlights from the revised standard:

1. An extensive section on **criticality analysis** was **added**. (See Section 5 and Appendix A.)

WHY IT MATTERS

Software safety criticality analyses need to be performed in order to better scope and tailor the software safety effort. If a Software Safety Litmus Test indicates the need for software safety on a project, the level of criticality helps determine the necessary software safety processes, analyses and safety design needed.

RATIONALE

While tailoring based on criticality always was expected after a Software Safety Litmus Test, there was confusion over how to perform it. The added Software Safety Criticality Analysis section provides details that help ensure software safety risks are tailored correctly. One size does not fit all for requirements. The new standard reflects the importance of considering the levels and severity of risks associated with specific programs and projects, and even different software within a project.

---

2. The number of **requirements** was **reduced** from 171 in revision B to 66 in revision C.

WHY IT MATTERS

The change allows more flexibility in meeting the requirements while maintaining the basic safety practices that practitioners need.

RATIONALE

Now, the requirements are clearer and streamlined. Also, requirements now covered in NPR 7150.2 NASA Software Engineering Requirements were removed, eliminating any duplication.

---

3. The **Software Safety Litmus Test** was **streamlined** and its use within the development life cycle was clarified. (See Appendix A.)

WHY IT MATTERS

The revisions will lead to improved understanding and application of the Software Safety Litmus Test among practitioners.

continued on back

RATIONALE
Because the litmus test determines if software is safety critical, it is imperative that this test be completed correctly. For this reason, the litmus test and requirements for conducting it were made clearer and easier to read.

---

4. **Detailed appendices** were **added** including information on potential software issues, a list of design recommendations, and checklists for commercial off-the-shelf software, tools and facility safety. (See Appendices B through F.)

WHY IT MATTERS
The standard is now a great resource to reference when executing software assurance practices.

RATIONALE
Providing these tips and practices in the standard helps with planning software safety efforts. The Software Safety Guidebook is still the best source for detailed safety practice options.

---

5. Revisions throughout the standard **address facility safety** in addition to flight systems.

WHY IT MATTERS
The standard never directly addressed facility safety before. This component now must be considered when working on software.

RATIONALE
Software affects more than flight systems and the standard needed to better reflect that.

---

6. Language was added regarding the **roles** of **acquirer** and **provider**. (See Section 4.)

WHY IT MATTERS
The new language clearly outlines NASA's role as the acquirer, contractors' roles as providers and the specifics of what should be in contracts between the two parties.

RATIONALE
The additions are meant to help ensure successful acquisitions and clarify NASA's role in overseeing them.

---

7. **Requirements** were **numbered**.

WHY IT MATTERS
Previously, the individual requirements were not identified by separate numbers.

RATIONALE
Numbers make it easier to reference specific requirements.

TAKE ACTION
Software Assurance practitioners, Safety and Mission Assurance technical authorities, and safety officers should review the changes to this standard and share it with project managers, software engineers and engineering technical authorities.

Have questions regarding the new standard? Contact Martha Wetherholt, NASA Technical Fellow for Software Assurance, at Martha.Wetherholt@nasa.gov.

or

View the official NASA document at
http://www.hq.nasa.gov/office/codeq/doctree/871913.htm