A-P-T RESEARCH, INC.

Software System Safety, Software Criticality, and Software Hazard Control Categories for Information Systems

Mike Pessoney, Shannon Stump

30th International System Safety Conference
Atlanta, GA

6 August – 10 August 2012

Presentation Contents

- Accolades for attendance
- Background for paper
- Basics of Software Safety Process
- Software Control Issues Addressed with Discussion
- Experience
- Summary

Accolades to the Attendees

- You are very brave to attend this presentation
  - Addresses only a small feature of the chain of events leading to software safety
  - Presents what is perhaps a minority opinion
- I'll try to make it worth your while anyway
  - A new way of thinking of software control and assigning software control levels
  - Easier to use for information systems
  - Permits full level of control range for information systems

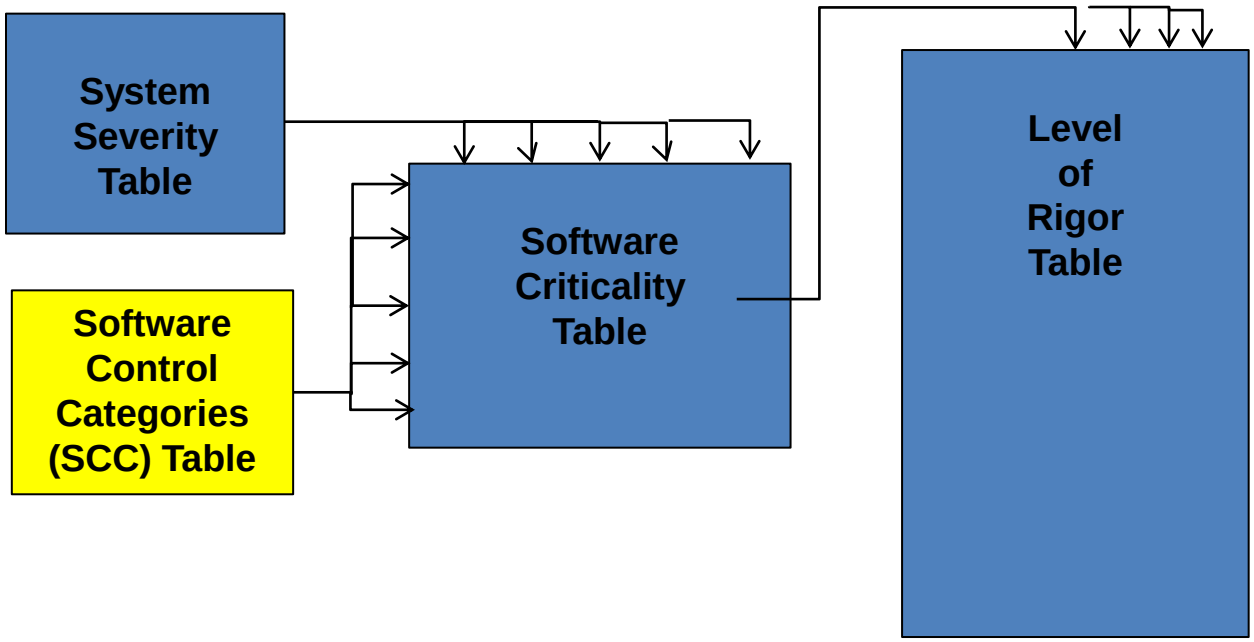
Background for Paper

1. MIL-STD-882E and the JSSSEH have almost identical tables for Software Control
2. For an information system, the authors found the control tables difficult to use
3. An alternate Software Hazard Control table was developed and used
4. The Alternate table may be of use on other information systems

Basics of Software Safety Process

- Following slides briefly describe part of the processes used in software system safety

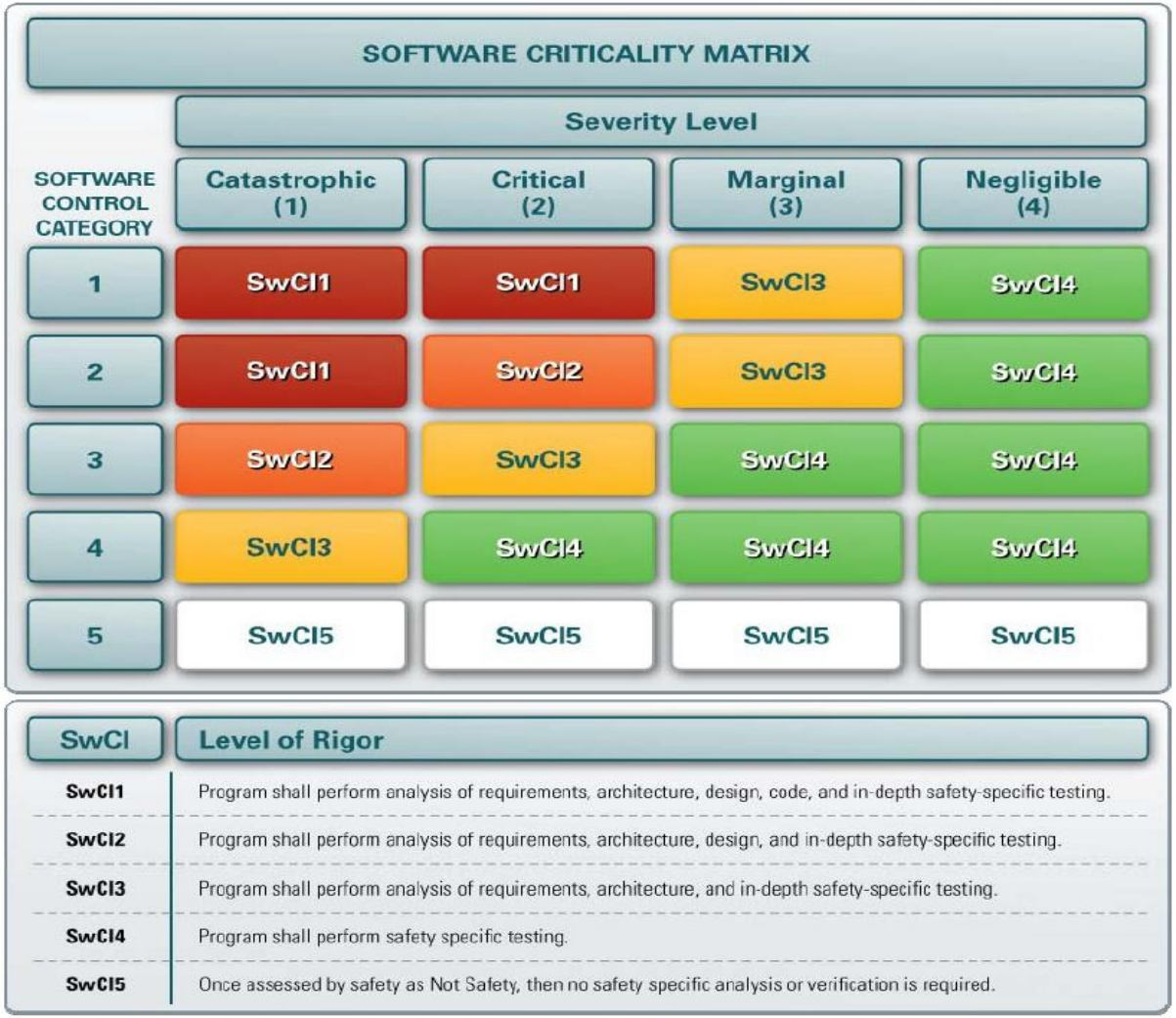
Software Control Categories



Severity Categories from MIL-STD-882E

SEVERITY CATEGORIES		
Severity Category	**Severity Level**	**Environment, Safety, and Occupational Health Mishap Result Criteria**
Catastrophic	1	Could result in one or more of the following: death, permanent total disability, irreversible significant environmental impact, or loss exceeding $10M.
Critical	2	Could result in one or more of the following: permanent partial disability, injuries or occupational illness that may result in hospitalization of at least three personnel, reversible significant environmental impact, or loss exceeding $1M but less than $10M.
Marginal	3	Could result in one or more of the following: injury or occupational illness resulting in 10 or more lost work days, reversible moderate environmental impact, or loss exceeding $100K but less than $1M.
Negligible	4	Could result in one or more of the following: injury or illness resulting in less than 10 lost work days, minimal environmental impact, or loss less than $100K.

Software Safety Criticality Matrix from MIL-STD-882E



SOFTWARE CRITICALITY MATRIX

SOFTWARE CONTROL CATEGORY	Severity Level: Catastrophic (1)	Critical (2)	Marginal (3)	Negligible (4)
1	SwCI1	SwCI1	SwCI3	SwCI4
2	SwCI1	SwCI2	SwCI3	SwCI4
3	SwCI2	SwCI3	SwCI4	SwCI4
4	SwCI3	SwCI4	SwCI4	SwCI4
5	SwCI5	SwCI5	SwCI5	SwCI5

SwCI	Level of Rigor
SwCI1	Program shall perform analysis of requirements, architecture, design, code, and in-depth safety-specific testing.
SwCI2	Program shall perform analysis of requirements, architecture, design, and in-depth safety-specific testing.
SwCI3	Program shall perform analysis of requirements, architecture, and in-depth safety-specific testing.
SwCI4	Program shall perform safety specific testing.
SwCI5	Once assessed by safety as Not Safety, then no safety specific analysis or verification is required.

Sample of Table of Required Actions for Level of Rigor

Software Integrity Assurance Task	SED S/W LOR			
	A SwCI1	B SwCI2	C SwCI3	D SwCI4
General				
Peer Reviews of all development artifacts are conducted at each phase (requirements, design, code, and test).	R	R	R	R
All design and software components containing safety critical functionality are identified as safety critical and linked to the appropriate requirement(s) in the SRS.	R	R	R	R
All safety critical functions identified through the FHA and all safety critical software components are documented and linked to the individual hazards identified in the hazard analysis.	R	R	R	R
System Functional Hazard Analysis (performed at System level and software architecture level IAW SAE ARP 4761).	R	R	R	
Planning Phase				
Software Development Plan shall include process activities related to software safety process to include software integrity assurance tasks.	R	R	R	R

Issue 1: Allow Level 1 Information

- Current software control categories allow no Level 1 control for information systems
- Software hazard control categories allow for appropriate Level 1 control for information systems

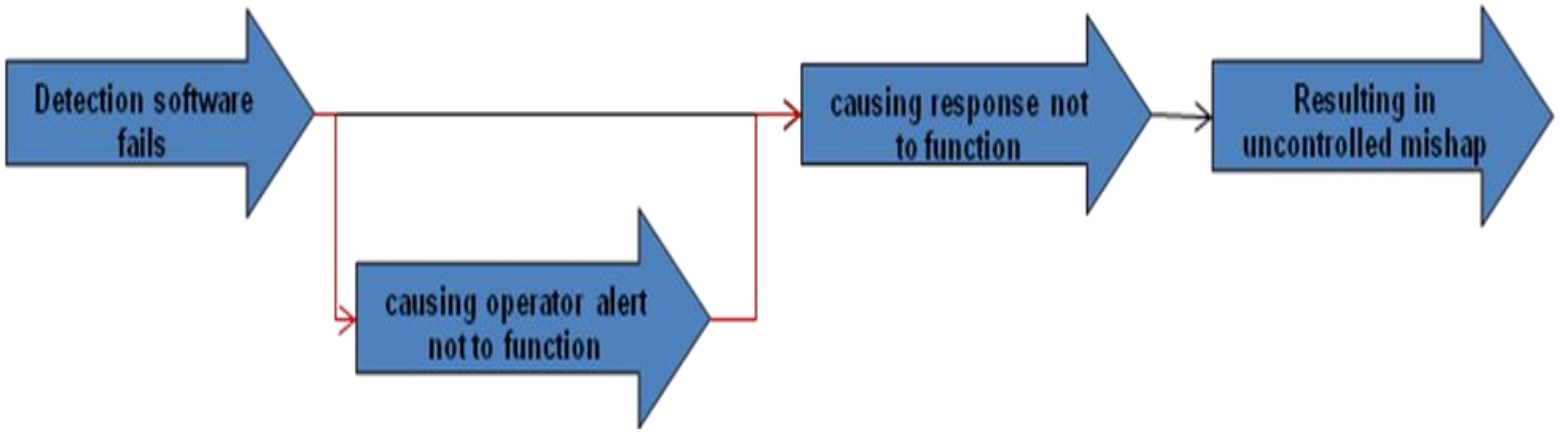
Hazard Control Example (1)

- Consider two aircraft systems where one has an engine fire control system that detects and responds to an engine fire while the other detects the engine fire condition and alerts the pilot to respond.
- The engine fire detection software is of equal safety significance in both systems, but is assigned level 1 control in the first scenario and level 2 control in the second scenario.

Hazard Control Example (2)

- A failure of the detection software has equal chances for a mishap in both systems.
- When a single software exception, failure, fault, or delay may lead directly to a mishap the software logically should be considered to have the highest level of hazard control.

Hazard Control Example (3)



- **<u>Operator or external system involvement is incidental to the level of control</u>.**

Example Software Hazard Control Categories (SHCC) Table for Information Systems

Level 1

Level	Name	Description	Examples
1	Sole Source Immediate (Now)	A software function or requirement that necessitates immediate response from an operator or external system based on data provided for mitigation or control over a hazard and potential immediate mishap. The software collection, distribution, display, or warning function or requirement provides the only information source.	Single source: Entering a minefield, stall warning, low oil pressure, red traffic light, engine overheat, fire alarm, tornado warning, impending collision, medical evacuation request message, no pulse, breathing interrupted

Issue 2: Address Hazards

- Current software control categories are based on (and named for) level of software control of safety-significant systems, subsystems and/or components
  - Specific hazards are associated with these safety-significant systems, subsystems, and/or components
  - Software control of the hazard (not the system) is the direct method of assessing software control
- All developed software hazard control categories are based on the level of software control of hazards (not subsystems)

MIL-STD-882E Software Control Categories (SCC) Level 1

Control Level	Name	MIL-STD-882E
1	AT Autonomous	Software functionality that exercises autonomous control authority over potentially safety-significant hardware systems, subsystems, or components (hazards) without the possibility of predetermined safe detection and intervention by a control entity to preclude the occurrence of a mishap or hazard. (This definition includes complex system/software functionality with multiple subsystems, interacting parallel processors, multiple interfaces, and safety-critical functions that are time critical.)

Issue 3: Simplify Discriminators

- Current software control categories discriminators are:
  - Autonomous
  - semi autonomous
  - redundant fault tolerant
  - informational
- Software hazard control categories discriminators are:
  - Sole source / multiple source
  - Immediate/ eventual

Software Hazard Control Categories (SHCC) Level 1

Level	Name	Description
1	Sole Source Immediate	Software functionality that exercises sole source control over hardware systems, subsystems, or component hazards without the possibility of intervention by an independent control entity to preclude the occurrence of a hazard and potential immediate mishap. (This definition includes complex system/software functionality with multiple subsystems, interacting parallel processors, multiple interfaces, and safety-critical functions that are time critical.)
		A software function or requirement that provides sole source information to a control entity that necessitates immediate response from an operator or external system based on data provided for mitigation or control over a hazard and potential immediate mishap. The software collection, distribution, display, or warning function or requirement provides the only information source.

Experience

- SHCC used to assess software level of control on an information system
  - Resulting control levels on this project were equivalent to use of the MIL-STD-882E SCC
  - Individual assessments were much easier to determine and the same for multiple evaluators
  - A better understanding of the effects of software control was shared by all evaluators

Summary

1. MIL-STD-882E and the JSSSEH have almost identical tables for Software Control
2. For an information system, the authors found the control tables difficult to use
3. An alternate Software Hazard Control table was developed and used
4. The alternate table may be of use on other information systems