

# **Software Risk Management Marshall Space Flight Center**

December 7, 2005

**Tim Crumbley**  
**tim.crumbley@nasa.gov**  
**(256) 544-5978**



## *Software Risk Management*

- Software Risk Management is a software engineering practice with processes, methods, and tools for managing risks in a software project.
- Software Risk Management provides a disciplined environment for
  - proactive decision-making to assess what could go wrong (risks),
  - determine which risks are important to deal with (prioritize),
  - implement strategies to deal with those risks (mitigation).
- Software Risk are managed throughout the software development life cycle
- Software Risk Management Task Review Process
  - Software risks are monitored and reviewed through Flight Software monthly project status meetings and various project reviews.
  - Software Assurance Organizational Audits.



## *Project Monitoring and Control*

### ● **Software Risk Management and Monitoring is a key element in the overall project monitoring and control process**

- Identification of the data to be monitored and collects quantifiable data on items including cost, schedule, and quality
- Use metrics to track project progress, product quality, and the effectiveness of the organization's standard processes
- Employ a combination of metrics and quality criteria to determine software quality
- Analyze the 'Critical Path' network which is the key to understanding the impact of slippages on the software project's delivery date
- Monitor stakeholder involvement via attendance at relevant meetings and reviews
- Monitor data management against the plan
- **Monitor project risks**
  - Track actual data on cost and schedules on a defined periodic basis
  - Ensure work products/artifacts are processed in accordance with the Project Software Configuration Management Plan and the Software Data Management Plan
  - Analyzes project data and issues to determine if there are significant deviations from the documented plans requiring corrective actions
  - Establish the need for taking corrective action
  - Participate with software team members in revising project plans
  - Review changes with affected stakeholders
  - Negotiate approval of changes with Flight Software management and the customer
  - Uses periodic management reviews and status meetings to keep the team, Flight Software Management and customer informed
  - Proactively participate in management's periodic review processes (e.g., monthly status meetings, metrics database, in-process reviews, quarterly reviews)
  - Use monthly status meeting with Flight Software Management to communicate the status of work products/artifacts and to discuss issues and the status of corrective actions
  - Develop and maintain a project 'Action Items' database to ensure communication, control, and timely closure of all items
  - At management reviews, use quantifiable data to facilitate communication on the status of cost, schedule, and quality against planned estimates
  - Ensure open lines of communication between project team members, project managers, and customer
  - Maintain Monthly Status Presentations in the project Software Development Library (SDL), and submit Monthly Metric Data to SEPG to be placed into the Flight Software PAL.
  - The Project software lead and software team participate in formal milestone reviews.
  - Ensure follow-up information is communicated and manage actions to closure
  - The Project software lead tracks corrective actions and proposed enhancements
  - Track changes to closure
  - Ensure that revisions to plans or processes are instituted within the project
  - Software configuration management baselines and tracks configuration of work products/artifacts
  - Track updates to baselined work products/artifacts and configurations
  - Ensure work products/artifacts are processed in accordance with the Project Software Configuration Management Plan and the Software Data



*Products and documentation are used in performing  
Software Risk Management activities.*

- The following products and documentation are used in performing Software Risk Management activities.
  - Project risk data, processes and charts
  - Project Schedules
  - System and Hardware Requirements
  - System Verification Plans
  - Flight Software Development Process Description Document processes
  - Project Monthly Software Status Meetings
  - Project Software Products
  - Software Change Request System
  - Avionics Status Meetings
  - Inter group Project Coordination, Planning and Status Meetings
  - Project and Organization Software Metric data
  - Project Software Schedules
  - Flight Software Process Asset Library (PAL)
  - Flight Software Risk Management Plan



# Flight Software Risk Checklist Approach

- A generic set of flight software risks that are considered throughout the project lifecycle.
- Software risks that have been identified on previous projects.
- The generic set of flight software risks are identified by development phases of a project,
- This checklist is only an aid to start software managers and software engineers thinking
- **Biggest problem is not managing known risk, the problem is knowing how to identify the appropriate software risks**

<b>System Requirements Phase</b>		
<i>13 checklist Items</i>		
<b>Software Planning Phase</b>		
<i>66 Checklist Items</i>		
<b>Software Requirements Phase</b>		
<i>73 Checklist Items</i>		
<b>Software Design Phase</b>		
<i>52 Checklist Items</i>		
<b>Software Implementation Phase</b>		
<i>87 Checklist Items for Coding &amp; Unit Test Phase</i>		
<i>97 Checklist Items for Integration &amp; System Testing Phase</i>		
<b>Acceptance Testing &amp; Release</b>		
<i>33 Checklist Items</i>		



● **SW Risk Checklist (Partial Sample)**

Software Requirements Phase	<u>RISK</u> Yes/No/Partial	<u>ACTION</u> Accept/Work
<b>Software Schedule:</b>		
Is there an adequate software Schedule in place?		
Is it being followed?		
Are changes to schedule being tracked?		
Are changes to schedule made by due process, in a planned manner or are events changing the schedule with no decision of whether there is something wrong in the process or program that needs to change to make schedule?		
Has it been chosen to meet the needs of software development or is just a time/date when systems will need the software?		
Has all the slack/contingency time on the critical path been used up?		
Are software metrics kept and reported regularly? Monthly?		
Are deviations to the development plan being tracked? Trended?		
Are the trends reported in a manner to allow timely and appropriate software and project management decisions?		



## *NASA Software Process Asset Library*

### *Software Risk Management Assets*

- **NASA Software Process Asset Library (PAL)**  
(<http://swpal.msfc.nasa.gov/>)
- **The NASA PAL has two goals:**
  - Encourage sharing of software engineering assets within the agency.
  - Provide a repository of software engineering needs.
- **Searching on word "RISK", 9 items matched the criteria.**
  - IRD - Risk Analysis Matrix
  - Boeing 314 - Software Risk Management
  - JSC - IRD - Risk Analysis Matrix
  - JSC - IRD - Risk Analysis Matrix
  - USA SFOC-PM0248 Risk Management Process
  - JSC - Boeing 314 - Software Risk Management
  - JSC - USA SFOC-PM0248 Risk Management Process
  - JSC - USA SFOC-PM0248 Risk Management Process
  - Flight Software Branch (EI32) Risk Management Plan