White Paper on

A Reference Model for Software and System Inspections

Project Deliverable for
"Inspections for Systems and Software"

Ms. Lulu He
lhe@fc-md.umd.edu

Dr. Forrest Shull (PI)
fshull@fc-md.umd.edu

1. Introduction

Software Quality Assurance (SQA) is an important component of the software development process. SQA processes provide assurance that the software products and processes in the project life cycle conform to their specified requirements by planning, enacting, and performing a set of activities to provide adequate confidence that quality is being built into the software. Typical techniques include:

- Testing
- Simulation
- Model checking
- Symbolic execution
- Management reviews
- Technical reviews
- Inspections
- Walk-throughs
- Audits
- Analysis (complexity analysis, control flow analysis, algorithmic analysis)
- Formal method

Our work over the last few years has resulted in substantial knowledge about SQA techniques, especially the areas of technical reviews and inspections. But can we apply the same QA techniques to the system development process? If yes, what kind of tailoring do we need before applying them in the system engineering context? If not, what types of QA techniques are actually used at system level? And, is there any room for improvement?

After a brief examination of the system engineering literature (especially focused on NASA and DoD guidance) [3-8] we found that:

- System and software development process interact with each other at different phases through development life cycle (Fig1.1 and Fig1.2)

- Reviews are emphasized in both system and software development. (Fig1.3). For some reviews (e.g. SRR, PDR, CDR), there are both system versions and software versions.
- Analysis techniques are emphasized (e.g. Fault Tree Analysis, Preliminary Hazard Analysis) and some details are given about how to apply them.
- Reviews are expected to use the outputs of the analysis techniques. In other words, these particular analyses are usually conducted in preparation for (before) reviews.

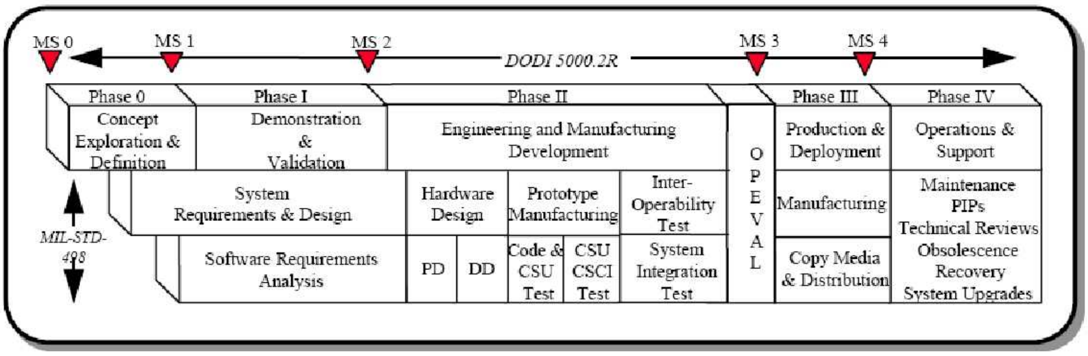

Fig 1.1 Relationship of Software to the Hardware Development Life Cycle [3]

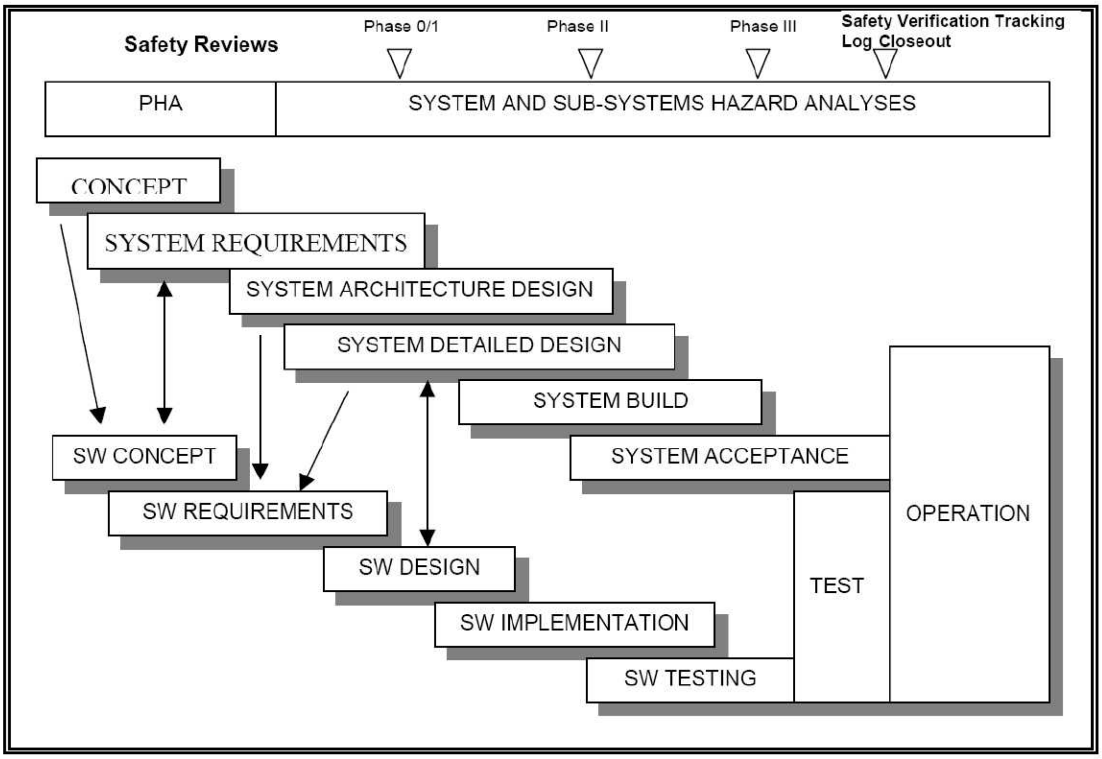

Fig 1.2 Safety, System, and Software Timeline [2]

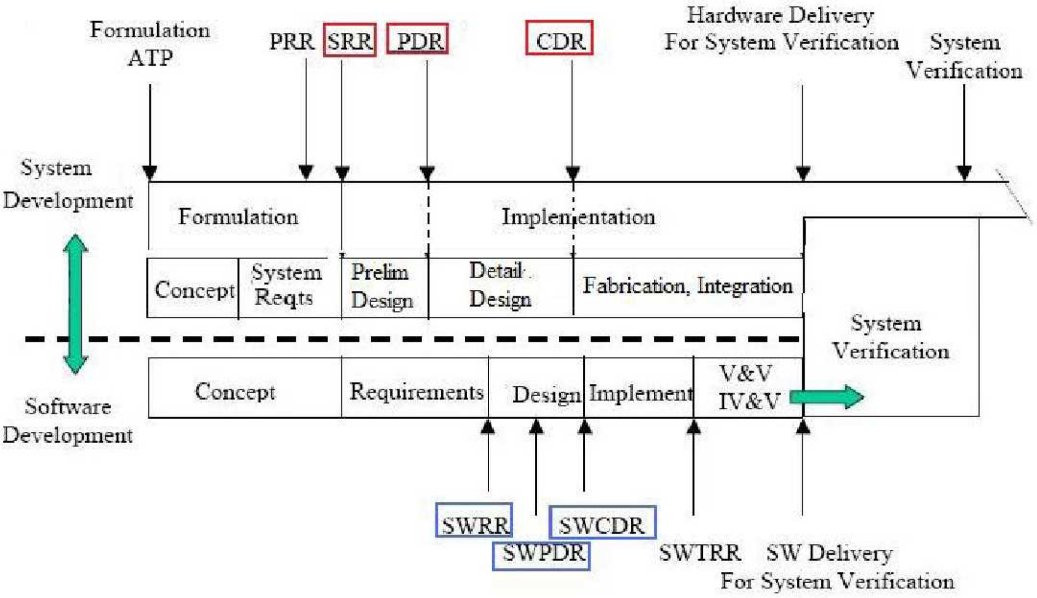

Fig 1.3 System and Software Development Process and Phasing (adapted from [5])

The goal of our work is to explore the interaction between the Quality Assurance (QA) techniques at the system level and the software level. Specifically, we are focusing on:

- What techniques do people use to review system/software quality issues during development?
- Which techniques account for both systems and software?
- How do system engineers and software engineers participate in each other's techniques?
- Is there any similarity between software inspections and system reviews? How can our knowledge and experiences in software inspection help to improve the system review process?

We began our work by focusing on **reviews** (inspections) and their role in QA of systems and software.

We formulated an initial model of recommendations based on reviewing existing guidebooks and standards [3-8]:

- The minimal set of quality reviews that should occur for both software and systems level artifacts
- The relevant expertise required at each review (e.g. whether software specialists attend each system-artifact review)

- The inputs that need to be fed into each review as well as the output
- The quality focus of each review.

This initial model, while it will never be complete, will be used to identify the most important reviews for us to focus on, and a way to begin to reason about the most important quality focus for each. We will also use this basic understanding as the input to a set of interviews we will undertake this month and next, in which we will ask NASA personnel about the degree to which their experience matches the recommendations in the model and for additional details they can provide.

2. System Engineering Process and Technical Reviews

A system in development proceeds through a sequence of distinct phases from concept to finished product, separated by Key Decision Points (KDP) (see Fig.1 [7][1]). KDPs are the events at which the *decision authority* determines the readiness of a program/project to progress to the next phase of the life cycle (or to the next KDP). Such decisions are made via milestone reviews tied to each KDP throughout the life cycle. For **KDP/milestone reviews**, external independent reviewers known as *Standing Review Board (SRB)* members evaluate the program/project and report their findings to the *decision authority*. For a program or project to prepare for the SRB, the technical teams must conduct their own **technical review process**. This process typically includes **project life-cycle reviews** and **peer reviews** at the subsystem and system level. Fig.1 shows the major phases of NASA project life-cycles, KDPs of each phase, project life-cycle reviews (for *Human Space Flight Mission*) and supporting reviews.

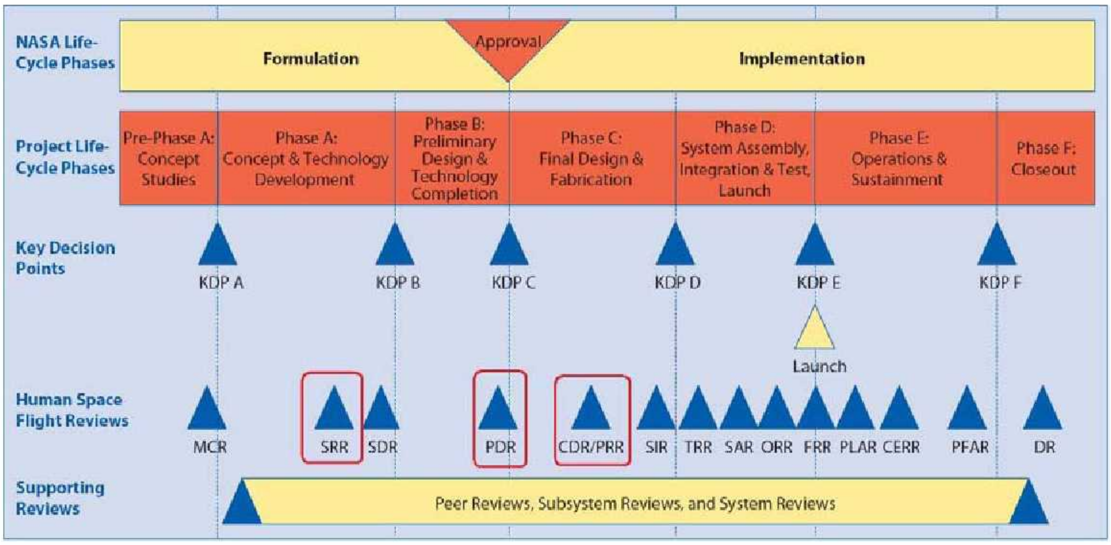
NASA Life-Cycle Phases
Formulation
Approval
Implementation
Project Life-Cycle Phases
Pre-Phase A: Concept Studies
Phase A: Concept & Technology Development
Phase B: Preliminary Design & Technology Completion
Phase C: Final Design & Fabrication
Phase D: System Assembly, Integration & Test, Launch
Phase E: Operations & Sustainment
Phase F: Closeout
Key Decision Points
KDP A
KDP B
KDP C
KDP D
KDP E
KDP F
Launch
Human Space Flight Reviews
MCR
SRR
SDR
PDR
CDR/PRR
SIR
TRR
SAR
ORR
FRR
PLAR
CERR
PFAR
DR
Supporting Reviews
Peer Reviews, Subsystem Reviews, and System Reviews

Fig 2.1 NASA project life cycle (adapted from [7])

Project Life-cycle Reviews

Project life-cycle reviews are mandatory reviews convened by the decision authority, which summarize the results of internal technical processes (peer reviews). These life-cycle reviews are used to assess the progress and health of a project and provide recommendations to the following KDP in which the decision authority for the project determines whether or not the project can proceed to the next life-cycle phase. Some examples of life-cycle reviews include **System Requirements Review (SRR), Preliminary Design Review (PDR), Critical Design**

[1] We note that the terminology in Fig 1.1 through Fig 2.1 varies depending on the sources. In the remainder of this document, we adopt the terminology from Fig 2.1 as being the most recent reference material.

Review (CDR), and **Acceptance Review (AR)**

Peer Reviews

In preparation for the life-cycle reviews, a project will conduct internal peer reviews that present technical approaches, trade studies, analyses, problem areas to **a peer group** for evaluation and comment. Peer reviews are focused, in-depth technical reviews that support the evolving design and development of a product, including critical documentation or data packages. They are often, but not always, held as supporting reviews for project life-cycle reviews such as PDR and CDR. A purpose of the peer review is to add value and reduce risk through **expert knowledge infusion, confirmation of approach, identification of defects,** and **specific suggestions for product improvements**. The results and issues that surface during these reviews are documented and reported out at the appropriate next higher element level. The peer reviews are not just meetings to share ideas and resolve issues, but are internal reviews that allow the project to **establish baseline** requirements, plans, or design.

Peer review is an independent evaluation by internal or external Subject Matter Experts (SMEs) who do not have a vested interest in the work product under review. The peer reviewers should be selected from outside the project, but they should have a similar technical background, and they should be selected for their skill and experience. Peer reviews are at a much greater level of detail than life-cycle reviews. Representatives from areas such as *manufacturing* and *quality assurance* should attend the reviews as active participants. Peer reviewers should be concerned with the **technical integrity** and **quality** of the product.

Peer reviews should be kept **simple** and **informal.** They should concentrate on a review of the documentation and minimize the viewgraph presentations. A roundtable format rather than a stand-up presentation is preferred. The peer reviews should give the full technical picture of items being reviewed.

Findings in Implementation Guidance

As to the review process, we obtained several observations when examining the existing standards and guidebooks [3-8]:

- A similar set of project life-cycle reviews (e.g. SRR, PDR, CDR, TRR, etc.) appeared in most of the guidebooks

- Specification of these project life-cycle reviews are given in the format of Entry Criteria (input) and Success Criteria. The criteria are similar but not exactly the same for a given project life-cycle review. No details are given for how to check the satisfaction of these criteria.

- Few guidebooks talked about the peer reviews in details, e.g. the types of peer reviews, participants, input/output, etc.

- Different review processes are provided in these guidebooks.

  Fig-2.2 depicts a typical flow diagram for the Technical Assessment Process [7]. Technical Assessment refers to the process of monitoring technical progress of a program/project through various kinds of technical reviews discussed above. Fig-2.2 identifies typical inputs, outputs, and activities of Technical Assessment Process.

  Fig-2.3 shows key activities in a technical review process [4]. Here technical reviews refer system level life-cycle reviews as well as lower level peer reviews.

  Fig-2.4 shows a diagram of the peer review/inspection processes [7]. Here Peer reviews/ inspections are defined as a review process for finding and fixing defects. Fig-2.4 identifies typical stages of peer reviews/inspections and participants as well as output forms of each stage.

  In summary, the three review process models differ in the following aspects:

  - Generality: Fig-2.2 and Fig-2.3 specify a general technical review process, while Fig-2.4 is about a specific type of review, i.e. peer review/inspection.
  - Ordering of Activities: the activities/stages in Fig-2.3 and Fig-2.4 are specified in a chronological manner, i.e. the previous activity needs to be finished before moving on to the next activity. Fig-2.2, however, identifies activities from the perspective of which aspects of the project to be assessed, e.g. productivity, product quality, etc. These activities (except the first activity -- preparation one) can be conducted concurrently.
  - Input/Output: Fig-2.2 is the only diagram which identifies both inputs and outputs of the review process. We will borrow these input/output categories when analyzing the specific Project Life-cycle reviews later in Section 3.
  - Level of Details: Fig-2.3 provides details of the activities conducted at each stage. [7] also provides details for each step in the peer review/inspection process in Fig-2.4. Fig-2.2 does not provide similar details because as a model of general review process it leaves all the details to the specific reviews (e.g. SRR, PDR, etc.)

Despite the difference in the process models, we found that the technical review process in Fig-2.3 and peer review/inspection process in Fig-2.4 share a very similar set of stages (activities).We can easily draw parallels between the stages of the two process models (e.g. Plan←→Planning, Familiarize←→Overview, Review←→Inspection Meeting, etc). It implies that peer review might adopt a similar review process as project life-cycle review, only less formal. In addition, we found that the same inspection process as in Fig-2.4 has also been used in software inspection [10]. We can see it as the evidence of the similarity of system reviews and software reviews. Since we have obtained a lot of experiences on the reviews/inspections in software environment, it would be interesting to see whether we can apply some of our experiences to the system context.

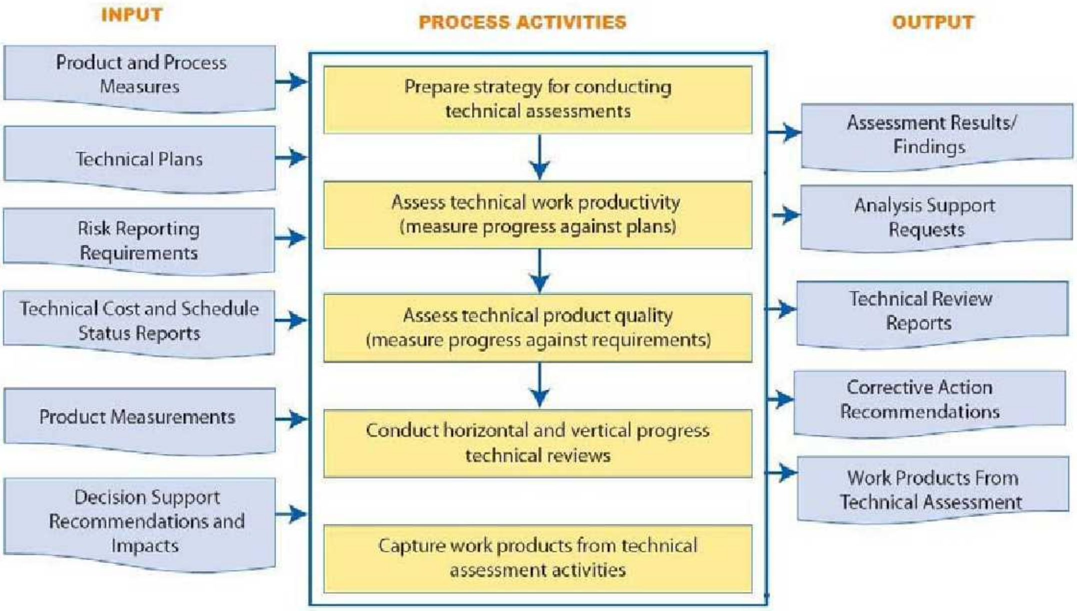
INPUT
PROCESS ACTIVITIES
OUTPUT
Product and Process Measures
Technical Plans
Risk Reporting Requirements
Technical Cost and Schedule Status Reports
Product Measurements
Decision Support Recommendations and Impacts
Prepare strategy for conducting technical assessments
Assess technical work productivity (measure progress against plans)
Assess technical product quality (measure progress against requirements)
Conduct horizontal and vertical progress technical reviews
Capture work products from technical assessment activities
Assessment Results/ Findings
Analysis Support Requests
Technical Review Reports
Corrective Action Recommendations
Work Products From Technical Assessment

Fig 2.2 Technical assessment process (adapted from [7])

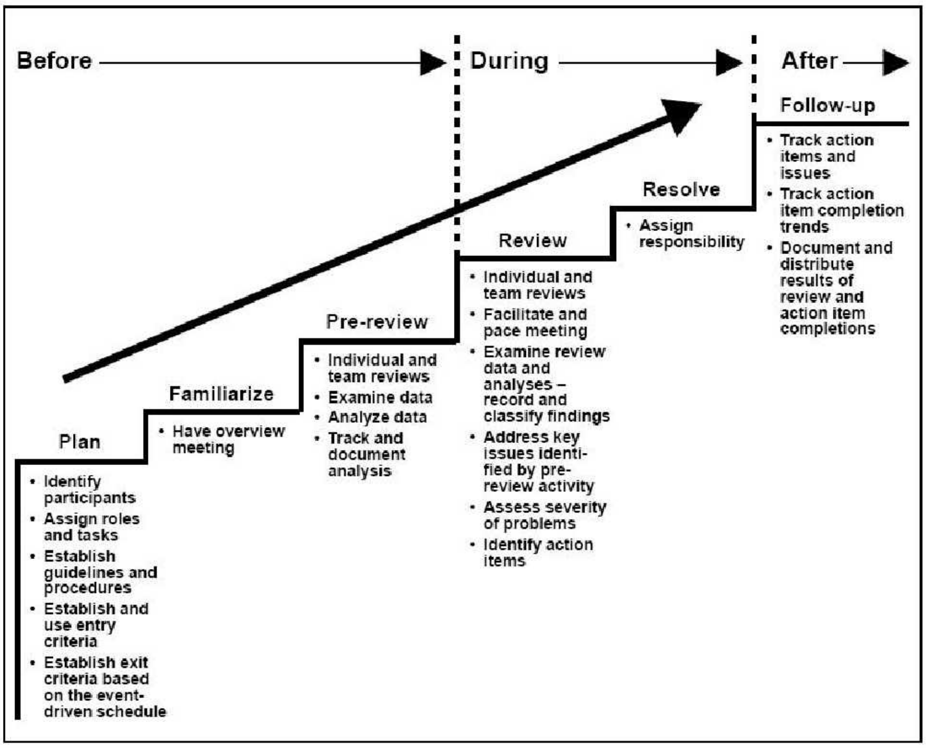
Before
During
After
Follow-up
• Track action items and issues
• Track action item completion trends
• Document and distribute results of review and action item completions
Resolve
• Assign responsibility
Review
• Individual and team reviews
• Facilitate and pace meeting
• Examine review data and analyses – record and classify findings
• Address key issues identified by pre-review activity
• Assess severity of problems
• Identify action items
Pre-review
• Individual and team reviews
• Examine data
• Analyze data
• Track and document analysis
Familiarize
• Have overview meeting
Plan
• Identify participants
• Assign roles and tasks
• Establish guidelines and procedures
• Establish and use entry criteria
• Establish exit criteria based on the event-driven schedule

Fig 2.3 Technical Review Process [4]

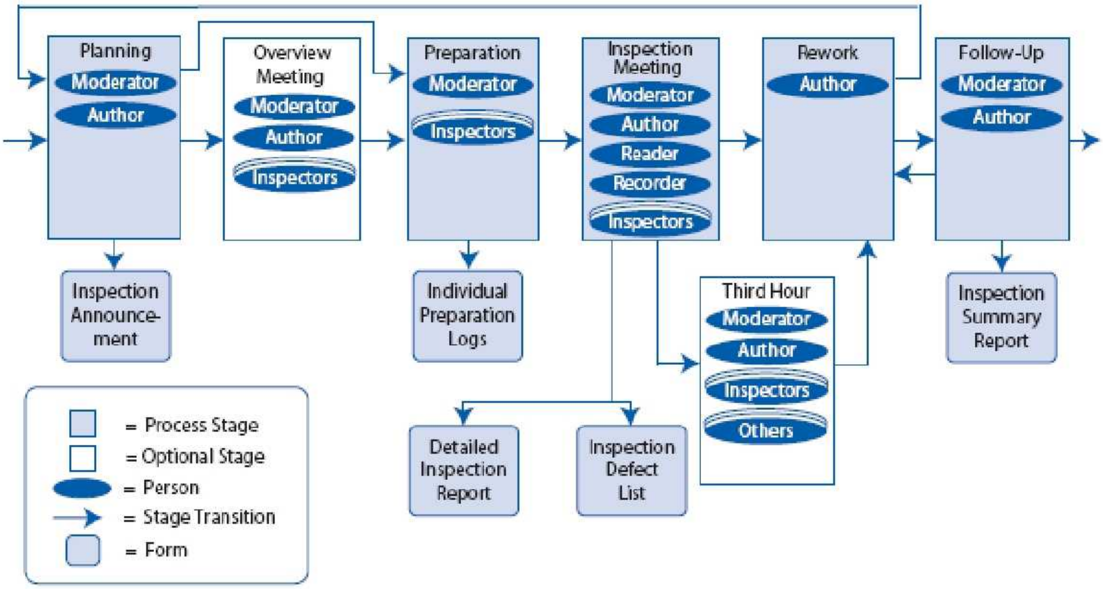

Fig2.4 Peer review/inspection process [7]

Our observations in the existing guidebooks lead us to the following questions:

- What's the minimal set of **project life-cycle reviews** that are conducted at system level?
- Do NASA projects follow a general process for **life-cycle reviews** in practice?
- For each project life-cycle reviews in the "minimum set", what types of **peer reviews** are conducted in preparation for them? For each **peer review**:
  - Does the review follow a general peer review process?
  - What is the relevant expertise required at each review (e.g. whether software specialists attend each system-artifact review)?
  - What are the inputs/outputs?

3. System Reviews: Project Life-cycle Reviews and Peer Reviews

One of our research goals is to obtain a better understanding of the various types of **peer reviews** conducted in support of **project life-cycle reviews**. Most guidebooks [1-8] have detailed specification for project life-cycle reviews, but few provide the same level of details for peer reviews. The peer review/inspection process in Fig-2.4 is the only model we found which specifically addresses peer reviews.

Since the purpose of peer reviews is to prepare for the corresponding project life-cycle reviews, we can "infer" what peer reviews would look like (e.g. goals, input/output) by analyzing the existing specifications of project life-cycle reviews. We picked the three most common project life-cycle reviews (as highlighted in Fig-1.3 and Fig-2.1), i.e. **System Requirements Review (SRR), Preliminary Design Review (PDR),** and **Critical Design Review (CDR)**. The NASA System Engineering Handbook [6] provides detailed guidance on how to conduct these project life-cycle reviews by specifying the *Purpose, Entrance Criteria* and *Success Criteria* of each review. We analyzed the given specifications as follows:

- **Entrance Criteria**: Peer reviews need to ensure that: 1) the required artifacts listed in the entrance criteria are ready; 2) some analysis has been conducted and the analysis results are ready. But not all the artifacts listed need to pass the peer reviews. As shown in Fig-2.2, part of the project life-cycle reviews is to assess the project progress and productivity, which is not in the scope of the peer review. Besides, it's not possible (not effective at least) to analyze all the required artifacts in ONE peer review considering the "simple and informal" nature of peer reviews. Therefore we classified listed artifacts according to the input categories proposed in Fig-2.2. We believe that such classification helps define the set of artifacts that are of interest to peer reviews and the number and type of peer reviews needed to "pre-review" these artifacts.

- **Success Criteria**: From the success criteria, we can "infer" the kinds of analysis and results that are needed from the peer reviews. We tried to relate the success criteria to:
  - Entrance Criteria: for each success criteria, which input artifacts were addressed
  - Checklists from Software Inspection: due to the similarity between the system peer review/inspection (Fig-2.3) and software inspection, we tried to develop a mapping between the success criteria of project life-cycle reviews SRR, PDR and CDR, and the JPL checklists of the corresponding software inspections, i.e. System Requirements Specification, Architecture Design, and Detailed Design [10]. We want to see which part of the success criteria can be addressed by JPL checklists and which part is left out (gap).

SRR – System Requirements Review

Purpose [7, pp.174]

SRR examines the functional and performance requirements defined for the system and the preliminary program or project plan and ensures that the requirements and the selected concept will satisfy the mission

Entrance Criteria [7]

1. *Successful completion of the Mission Concept Review (MCR) and responses made to all MCR Requests for Action (RFAs) and Review Item Discrepancies (RIDs).*
2. *A preliminary Systems Requirements Review (SRR) agenda, success criteria, and charge to the board have been agreed to by the technical team, project manager, and review chair prior to the SRR.*
3. *The following technical products for hardware and software system elements are available to the cognizant participants prior to the review:*

   a. system requirements document;

   b. system software functionality description;

   c. updated Operations Concept (ConOps);

   d. updated mission requirements, if applicable;

   e. baselined System Engineering Management Plan (SEMP);

   f. risk management plan;

   g. preliminary system requirements allocation to the next lower level system;

   h. updated cost estimate;

   i. technology development maturity assessment plan;

   j. updated risk assessment and mitigations (including PRA, as applicable);

   k. logistics documentation (e.g., preliminary maintenance plan);

   l. preliminary human rating plan, if applicable;

   m. software development plan;

   n. system Safety and Mission Assurance (SMA) plan;

   o. Configuration Management (CM) plan;

   p. initial document tree;

   q. verification and validation approach;

   r. preliminary system safety analysis;

   s. other specialty disciplines, as required.

Criteria 1 checks whether the prerequisite review (i.e. MCR) has been successfully conducted and relevant issues have been resolved.
Criteria 2 checks whether the needed (management) materials are ready for review.
Criteria 3 lists all the technical products need to be ready for assessment in the review.

Classifications of Input Artifacts

We classified these required artifacts according to the input categories proposed in the Technical Assessment Process model [7] (see Fig-2.2):

Product: *Requirements & Allocations*

- updated ConOps
- updated mission requirements
- system requirements document
- System Software Functionality Descriptions
- preliminary system requirements allocation to the next lower level system

Technical Plans

- System Engineering Management Plans (SEMP)
- Risk management plan
- technology development maturity assessment plan
- logistics documentation (e.g. preliminary maintenance plan);
- preliminary human rating plan
- software development plan
- system SMA (safety and mission assurance) plan
- CM plan
- initial document tree
- verification and validation approach

Technical cost and schedule status reports

- updated risk assessment and mitigations
- updated cost estimate

Safety

- preliminary system safety analysis

[Note: blue text is used to highlight the software-related artifacts]

Questions:

- Do the reviewers need to examine every technical artifact mentioned in the entry criteria?
- In addition to the availability, what are the other qualities that need to be checked for each artifact?
- Do we need to conduct peer reviews for assessing all the technical products listed or just a subset of them (e.g. those in the ***Product*** and ***Safety*** categories)?

Success Criteria [7]

1. *The project utilizes a sound process for the allocation and control of requirements throughout all levels, and a plan has been defined to complete the definition activity within schedule constraints.*
2. *Requirements definition is complete with respect to top-level mission and science requirements, and interfaces with external entities and between major internal elements have been defined.*
3. *Requirements allocation and flow-down of key driving requirements have been defined down to subsystems.*
4. *Preliminary approaches have been determined for how requirements will be verified and validated down to the subsystem level.*
5. *Major risks have been identified and technically assessed, and viable mitigation strategies have been defined*

Relating Success Criteria to Entrance Criteria

Criteria 1 involve the assessment of **process measures** (requirements allocation and control process), **technical plans**, and the **schedule and cost status** (resource constraints). --- How to define and measure "sound"?

Criteria 2 involve the assessment of **product** quality — "completeness" of the requirements definition and the internal and external interfaces. --- How about other qualities like sufficiently detailed, understandable, consistent?

Criteria 3 also involve assessment of **product** quality — requirements allocation and flow-down "defined" --- IPR/I also applied? Traceability?

Criteria 4 involve the assessment of V&V **plans** of requirements

Criteria 5 involve the assessment of risk management. All the "major" risks have been identified and assessed? What makes a "viable" mitigation strategy?

Relating Success Criteria to JPL Checklists

We tried to map the success criteria with the System Requirements Checklist as follows:

SRR Success Criteria	**Criteria 1**	**Criteria 2**	**Criteria 3**	**Criteria 4**	**Criteria 5**
System Requirements	N/A	Functionality Completeness	Traceability	Testability	<GAP>

Checklist		Interface			

Criteria 1 does not map to any checklist because it involves the assessment of process, plans and resources status, which are out of the scope of peer reviews/inspections.

Criteria 5 involves risk analysis, which does not directly map to any part of the JPL checklists. But it might be related to the following checklists in some aspect:

- Functional correctness
- The “-ilities”: For example, safety related analyses like Preliminary Hazard Analysis (PHA)
- Testability: Fault Tree Analysis, identifying all the error conditions

Corresponding JPL checklists are listed as follows:

FUNCTIONALITY

1. Are all functions clearly and unambiguously described?
2. Are all described functions necessary and together sufficient to meet mission and system objectives?

COMPLETENESS

1. Are requirements stated as completely as possible? Have all incomplete requirements been captured as TBDs?
2. Has a feasibility analysis been performed and documented?
3. Is the impact of not achieving the requirements documented?
4. Have trade studies been performed and documented?
5. Have the security issues of hardware, software, operations personnel and procedures been addressed?
6. Has the impact of the project on users, other systems, and the environment been assessed?
7. Are the required functions, external interfaces and performance specifications prioritized by need date? Are they prioritized by their significance to the system?

INTERFACES

1. Are all external interfaces clearly defined?
2. Are all internal interfaces clearly defined?
3. Are all interfaces necessary, together sufficient, and consistent with each other?

TRACEABILITY

1. Are all functions, structures and constraints traced to mission/system objectives?
2. Is each requirement stated in such a manner that it can be uniquely referenced in subordinate documents?

TESTABILITY

1. Can the system be tested, demonstrated, inspected or analyzed to show that it satisfies requirements?
2. Are requirements stated precisely to facilitate specification of system test success criteria and requirements?

CLARITY

3. Are the requirements clear and unambiguous (i.e, are there aspects of the requirements that you do not understand; can they be misinterpreted)?

PDR (Preliminary Design Review)

Purpose [7, pp.177]

PDR demonstrates that the preliminary design meets all system requirements with acceptable risks and within the cost and schedule constraints. It will show that the correct design option has been selected, interfaces have been identified, and verification methods have been described.

Entrance Criteria [7]

1. *Successful completion of the SDR or MDR and responses made to all SDR or MDR RFAs and RIDs, or a timely closure plan exists for those remaining open.*
2. *A preliminary PDR agenda, success criteria, and charge to the board have been agreed to by the technical team, project manager, and review chair prior to the PDR.*
3. *PDR technical products listed below for both hardware and software system elements have been made available to the cognizant participants prior to the review:*
    a. *Updated baselined documentation, as required.*
    b. *Preliminary subsystem design specifications for each configuration item (hardware and software), with supporting tradeoff analyses and data, as required. The preliminary software design specification should include a completed definition of the software architecture and a preliminary database design description as applicable.*
    c. *Updated technology development maturity assessment plan.*
    d. *Updated risk assessment and mitigation.*
    e. *Updated cost and schedule data.*
    f. *Updated logistics documentation, as required.*
    g. *Applicable technical plans (e.g., technical performance measurement plan, contamination control plan, parts management plan, environments control plan, EMI/EMC control plan, payload-to-carrier integration plan, quality assurance plan, producibility/ manufacturability program plan, reliability program plan).*

h. Applicable standards.
i. Safety analyses and plans.
j. Engineering drawing tree.
k. Interface control documents.
l. Verification and validation plan.
m. Plans to respond to regulatory (e.g., National Environmental Policy Act) requirements, as required.
n. Disposal plan.
o. Technical resource utilization estimates and margins.
p. System-level safety analysis.
q. Preliminary LLIL.

Criteria 1 checks whether the prerequisite review (i.e. SDR or MCR) has been successfully conducted and relevant issues have been resolved.
Criteria 2 checks whether the needed (management) materials are ready for review.
Criteria 3 lists all the technical products need to be ready for assessment in the review.

Classifications of Input Artifacts

Similarly we classified the technical products listed as follows:

Product: *Preliminary system design*

- Preliminary subsystem design specifications for each CI (software and hardware)
- Interface control documents
- Engineering drawing tree
- Updated baselined documentation, as required
- Preliminary LLIL

Technical Plans

- Verification and validation plan
- Safety analyses and plans.
- Disposal plan
- Plans to respond to regulatory requirements
- Applicable technical plans (e.g., technical performance measurement plan, contamination control plan, parts management plan, environments control plan, EMI/EMC control plan, payload-to-carrier integration plan, producibility /manufacturability program plan, reliability program plan, quality assurance plan)
- Updated logistics documentation.
- Updated technology development maturity assessment plan

Technical cost and schedule status reports

- Technical resource utilization estimates and margins.
- Updated cost and schedule data

Safety

- System-level safety analysis.

Success Criteria [7]

1. *The top-level requirements—including mission success criteria, TPMs, and any sponsor-imposed constraints—are agreed upon, finalized, stated clearly, and consistent with the preliminary design.*
2. *The flow-down of verifiable requirements is complete and proper or, if not, an adequate plan exists for timely resolution of open items. Requirements are traceable to mission goals and objectives.*
3. *The preliminary design is expected to meet the requirements at an acceptable level of risk.*
4. *Definition of the technical interfaces is consistent with the overall technical maturity and provides an acceptable level of risk.*
5. *Adequate technical interfaces are consistent with the overall technical maturity and provide an acceptable level of risk.*
6. *Adequate technical margins exist with respect to TPMs.*
7. *Any required new technology has been developed to an adequate state of readiness, or backup options exist and are supported to make them a viable alternative.*
8. *The project risks are understood and have been credibly assessed, and plans, a process, and resources exist to effectively manage them.*
9. *SMA (e.g., safety, reliability, maintainability, quality, and EEE parts) has been adequately addressed in preliminary designs and any applicable SMA products (e.g., PRA, system safety analysis, and failure modes and effects analysis) have been approved.*
10. *The operational concept is technically sound, includes (where appropriate) human factors, and includes the flow-down of requirements for its execution.*

Relating Success Criteria to Entrance Criteria

Criteria 1 & 3 involve the assessment of the *product* - **requirements and preliminary design**, and the relationship between the two (the latter needs to satisfy and be consistent with the former).

Criteria 2 involve the assessment of the *product* – the **flow-down of the requirements**. It needs to be complete, proper, and traceable.

Criteria 4 & 5 involve the assessment of the *product* – **the technical interface**. Check its consistency with the overall technical maturity and the risks involved.

Criteria 6 involve the assessment of the *product measures* – **TPM**. Adequate technical margins exist.

Criteria 7 & 8 involve the assessment of the *risks.*

Criteria 9 involve the assessment of the *product* – preliminary design with respect to the satisfactions of SMA requirements.

Criteria 10 involve the assessment of *product*- operational concept.

Relating Success Criteria to JPL Checklists

We tried to relate the Software Requirements Checklist to these criteria as follows:

PDR Success Criteria	**Criteria 1**	**Criteria 2**	**Criteria 3**	**Criteria 4**	**Criteria 5**
Arch Design Checklist	Completeness.2	Functionality.1	Traceability.1	Interface Level of Detail	Interface Level of Detail
PDR Success Criteria	**Criteria 6**	**Criteria 7**	**Criteria 8**	**Criteria 9**	**Criteria 10**
Arch Design Checklist	Performance	Completeness. 3&4	Completeness.5	Maintainability Reliability	<GAP>

Criteria 10 involves the assessment of the technical risks of operational concept, which does not map directly to any part of the JPL checklists.

Corresponding JPL checklists are listed as follows:

COMPLETENESS

1. Are the goals defined?
2. Have all TBDs been resolved in requirements and specifications?
3. Can the design support any anticipated changes in the TBD requirements?
4. Have the impacts of the TBDs been assessed?
5. Has a risk plan been made for the parts of the design which may not be feasible?
6. Have design tradeoffs been documented? Does the documentation include the definition of the trade space and the criteria for choosing between tradeoffs?
7. Has design modeling been performed and documented?
8. Are all of the assumptions, constraints, decisions, and dependencies for this design documented?

FUNCTIONALITY

1. **Are the specifications for the modules consistent with the full functionality required for the module in the SRD and SIS-1?**
2. Is an abstract algorithm specified for each sublevel module?
3. Will the selected design or algorithm meet all of the requirements for the module?

TRACEABILITY

1. Are all parts of the design traced back to requirements in SRD, SIS-1, other project documents?

2. Can all design decisions be traced back to trade studies?

3. Has the impact of special or unusual features of inherited designs on the current design been addressed?

4. Are all known risks from inherited designs identified and analyzed?

MAINTAINABILITY

1. Is the design modular?

2. Do the modules have high cohesion and low coupling?

PERFORMANCE

1. Has performance modeling been performed when appropriate and has it been documented?

2. Are all performance parameters specified (e.g., real time constraints, memory size, speed requirements, amount of disk I/O)?

3. Do processes have time windows (e.g., flags may be needed to "lock" structures, semaphores, some code may need to be non-interruptible)?

4. Have all critical paths of execution been identified and analyzed?

RELIABILITY

1. Does the design provide for error detection and recovery (e.g. input checking)?

2. Are abnormal conditions considered?

3. Are all error conditions specified completely and accurately?

4. Does the design satisfy all systems integrity commitments for this product?

CDR (Critical Design Review)

Purpose[7, , pp.178]

CDR demonstrates that the maturity of the design is appropriate to support proceeding with full scale fabrication, assembly, integration, and testing, and that the technical effort is on track to complete system development and mission operations in order to meet mission performance requirements within the identified cost and schedule constraints.

Entrance Criteria[7]

1. *Successful completion of the PDR and responses made to all PDR RFAs and RIDs, or a timely closure plan exists for those remaining open.*
2. *A preliminary CDR agenda, success criteria, and charge to the board have been agreed to by the technical team, project manager, and review chair prior to the CDR.*
3. *CDR technical work products listed below for both hardware and software system elements have been made available to the cognizant participants prior to the review:*
    a. *updated baselined documents, as required;*

b. product build-to specifications for each hardware and software configuration item, along with supporting tradeoff analyses and data;

c. fabrication, assembly, integration, and test plans and procedures;

d . Technical data package (e.g., integrated schematics, spares provisioning list, interface control documents, engineering analyses, and specifications);

e. operational limits and constraints;

f. technical resource utilization estimates and margins;

g. acceptance criteria;

h. command and telemetry list;

i. verification plan (including requirements and specifications);

j. validation plan;

k. launch site operations plan;

l. checkout and activation plan;

m. disposal plan (including decommissioning or termination);

n. updated technology development maturity assessment plan;

o. updated risk assessment and mitigation;

p. update reliability analyses and assessments;

q. updated cost and schedule data;

r. updated logistics documentation;

s. software design document(s) (including interface design documents);

t. updated LLIL;

u. subsystem-level and preliminary operations safety analyses;

v. system and subsystem certification plans and requirements (as needed); and

w. system safety analysis with associated verifications.

Criteria 1 checks whether the prerequisite reviews (e.g. PDR) have been successfully conducted and relevant issues have been resolved.

Criteria 2 checks whether the needed (management) materials are ready for review.

Criteria 3 lists all the technical products that need to be ready for assessment in the review.

Classifications of Input Artifacts

Product: ***critical system design***

- updated baselined documents
- product build-to specifications for each hardware and software configuration item, along with supporting tradeoff analyses and data;
- technical data package (e.g., integrated schematics, spares provisioning list, interface control documents, engineering analyses, and specifications)
- updated LLIL
- command and telemetry list
- software design document(s) (including interface design documents)
- updated logistics documentation

Technical Plans

- fabrication, assembly, integration, and test plans and procedures;

- verification & validation plan;
- acceptance criteria;
- launch site operations plan
- checkout and activation plan
- disposal plan
- updated technology development maturity assessment plan
- system and subsystem certification plans and requirements

Technical cost and schedule status reports

- operational limits and constraints;
- technical resource utilization estimates and margins
- updated cost and schedule data;

Risk

- updated risk assessment and mitigation
- update reliability analyses and assessments

Safety

- subsystem-level and preliminary operations safety analyses
- system safety analysis with associated verifications

Success Criteria [7]

1. *The detailed design is expected to meet the requirements with adequate margins at an acceptable level of risk.*
2. *Interface control documents are appropriately matured to proceed with fabrication, assembly, integration, and test, and plans are in place to manage any open items.*
3. *High confidence exists in the product baseline, and adequate documentation exists or will exist in a timely manner to allow proceeding with fabrication, assembly, integration, and test.*
4. *The product verification and product validation requirements and plans are complete.*
5. *The testing approach is comprehensive, and the planning for system assembly, integration, test, and launch site and mission operations is sufficient to progress into the next phase.*
6. *Adequate technical and programmatic margins and resources exist to complete the development within budget, schedule, and risk constraints.*
7. *Risks to mission success are understood and credibly assessed, and plans and resources exist to effectively manage them.*
8. *SMA (e.g., safety, reliability, maintainability, quality, and EEE parts) have been adequately addressed in system and operational designs, and any applicable SMA plan products (e.g., PRA, system safety analysis, and failure modes and effects analysis) have been approved.*

Relating Success Criteria to Entrance Criteria

Criteria 1 involve the assessment of the *product* -**detailed design** should meet the requirements baselined in previous reviews.
Criteria 2 involve the assessment of the *product* – **Interface design**
Criteria 3 involve the assessment of the quality of ***product*** and ***documentation*.**
Criteria 4 involve the assessment V&V requirements and **plans**.
Criteria 5 involve the assessment of testing **plans**.
Criteria 6&7 involve the assessment of technical and management-related **risks**.
Criteria 8 involve the assessment of ***product*** – system and operational design with respect to the satisfaction of SMA requirements

Relating Success Criteria to JPL Checklists

We tried to relate the Detailed Design Checklist to these criteria as follows:

CDR Success Criteria	**Criteria 1**	**Criteria 2**	**Criteria 3**	**Criteria 4**
Detailed Design Checklist	Traceability Functionality	Interface	Completeness Correctness	Testability
CDR Success Criteria	**Criteria 5**	**Criteria 6**	**Criteria 7**	**Criteria 8**
Detailed Design Checklist	Testability	N/A	<GAP>	Maintainability Performance Reliability

Criteria 6 does not map to any checklist because it involves the management-related risks (technical margins, plans and resources), which are out of the scope of peer reviews/ inspections.
Criteria 7 involves risk analysis and mitigation in general, which does not map directly to any part of the JPL checklists. Discussion in SRR section also applies here.

Corresponding JPL checklists are listed as follows:

COMPLETENESS
1. Have the specifications for all units in the program set been provided?
2. Have all the acceptance criteria been described?
3. Have the algorithms (e.g., in PDL) used to implement this unit been specified?
4. Have all the calls made by this unit been listed?

5. Has the history of inherited designs been documented along with known risks?

CORRECTNESS

1. Is there logic missing?
2. Are literals used where a constant data name should be used?
3. Are all conditions handled (greater-than, equal-to, less-than-zero, switch/case)?
4. Are branches correctly stated (the logic is not reversed)?

FUNCTIONALITY

1. Does this design implement the specified algorithm?
2. Will this design fulfill its specified requirement and purpose?

INTERFACE

1. Do argument lists match in number, type, and order?
2. Are all inputs and outputs properly defined and checked?
3. Has the order of passed parameters been clearly described?
4. Has the mechanism for passing parameters been identified?
5. Are constants and variables passed across an interface treated as such in the unit's design (e.g. a constant should not be altered within a subroutine)?
6. Have all the parameters and control flags passed to and returned by the unit been described?
7. Have the parameters been specified in terms of unit of measure, range of values, accuracy, and precision?
8. Is the shared data areas mapped consistently by all routines that access them?

MAINTAINABILITY

1. Does this unit have high internal cohesion and low external coupling (i.e., changes to this unit do not have any unforeseen effects within the unit and have minimal effect on other units)?
2. Has the complexity of this design been minimized?
3. Does the header meet project standards (e.g., purpose, author, environment, nonstandard features used, development history, input and output parameters, files used, data structures used, units invoking this one, units invoked by this one, and explanatory notes)?
4. Does the unit exhibit clarity, readability, and modifiability to meet maintenance requirements?

PERFORMANCE

1. Do processes have time windows?
2. Have all the constraints, such as processing time and size, for this unit been specified?

RELIABILITY

1. Are default values used for initialization and are they correct?

2. Are boundary checks performed on memory accesses (i.e., arrays, data structures, pointers, etc.) to insure that only the intended memory locations are being altered?
3. Is error checking on inputs, outputs, interfaces, and results performed?
4. Are meaningful messages issued for all error conditions?
5. Do return codes for particular situations match the global definition of the return code as documented?
6. Are undesired events considered?

TESTABILITY

1. Can each unit be tested, demonstrated, analyzed, or inspected to show that they satisfy requirements?
2. Does the design contain checkpoints to aid in testing (e.g., conditionally compiled code, data assertion tests)?
3. Can all logic be tested?
4. Have test drivers, test data sets, and test results for this unit been described?

TRACEABILITY

1. Are all parts of the design traced back to the requirements?
2. Can all design decisions be traced back to trade studies?
3. Have all the detailed requirements for each unit been specified?
4. Have the unit requirements been traced to the SSD-1? Have the SSD-1 specifications been traced to the unit requirements?
5. Has a reference to the code or the code itself been included?

Summary

The relationship between Success Criteria and JPL checklists are summarized as follows:

	SRR	PDR	CDR
Quality Aspects in JPL Checklist Covered	Completeness Functionality Interface Testability Traceability	Completeness Functionality Maintainability Performance Reliability Traceability	Completeness Correctness Interface Functionality Maintainability Performance Reliability Testability Traceability
Quality Aspects in JPL Checklist Not Covered	Compliance Consistency Correctness Data Usage Maintainability Performance Reliability	Compliance Consistency Correctness Data Usage	Compliance Consistency Data Usage
Software Artifacts as Input	System Software Functionality Descriptions	Preliminary subsystem design specifications for software CI	Software design docs (including interface design docs)

The first row listed all the checklist categories which are covered in the success criteria of SRR, PDR, and CDR respectively. The checklists might help us to prepare and conduct the peer reviews leading up to those project life-cycle reviews.

By "quality aspects not covered in JPL checklists", we mean that the checklist categories are not directly, specifically addressed in the success criteria. There are two possible situations:

- For the qualities like "Consistency" and "Correctness", we think they are so important that there is no reason not to assess them in project life-cycle reviews and include as the success criteria. One reason for such omission is that they are "basic" qualities which should be met by every technical product. It's like a "common sense" known to all technical teams so the success criteria does not address them specifically. We could confirm this hypothesis by interviewing with the people from "the field".
- For other qualities e.g. "Data Usage", the gap might be due to the difference between software and system development. Even for those qualities covered, e.g.

"Performance", "Testability", "Reliability", it's not a complete mapping. There still might be some parts of JPL checklists in categories that can not apply to system context, or some aspects of system context are not addressed by checklists developed for software system. We need to be very careful when dealing with such differences.

Risk Analysis is one aspect in the success criteria which does not map directly to any part of the JPL checklists. But as we discussed before in SRR section, it might be related to the following checklist categories

- Functional correctness
- The "-ilities": For example, safety related analyses like Preliminary Hazard Analysis (PHA)
- Testability: Fault Tree Analysis, identifying all the error conditions

References

[1] NASA System Safety Handbook: http://www.dfrc.nasa.gov/Business/DMS/PDF/DHB-S-001.pdf

[2] NASA Software Safety Guidebook: http://www.hq.nasa.gov/office/codeq/doctree/871913.pdf

[3] SOFTWARE SYSTEM SAFETY HANDBOOK, A Technical & Managerial Team Approach, December 1999 (Joint Software System Safety Committee) (available at http://sunnyday.mit.edu/safety-club/)

[4]DAU System Engineering Fundamentals
[5] MSFC-HDBK 3173 Project Management and System Engineering Handbook
[6] NASA SP-2007-6105, NASA SYSTEMS ENGINEERING HANDBOOK (Rev. 1) (DEC 2007)
[7] NPR 7120.5, NASA Space Flight Program and Project Management Processes and Requirements. Washington, DC, 2007.
[8] NPR 7123.1, Systems Engineering Processes and Requirements. Washington, DC, 2007

[9]System Analysis, Design, and Development, concepts, principles, and practices. By Charles S. Wasson, John Wiley & Sons, Inc publication, 2006.

[10]Http://fc-md.umd.edu/eb