Since its founding, the National Aeronautics and Space Administration (NASA) has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA technical report server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI report series, which include the following report types:

- **Technical Publication**: Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- **Technical Memorandum**: Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- **Contractor Report**: Scientific and technical findings by NASA-sponsored contractors and grantees.

- **Conference Publication**: Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- **Special Publication**: Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- **Technical Translation**: English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, and organizing and publishing research results.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at www.sti.nasa.gov
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA STI help desk at 301-621-0134
- Phone the NASA STI help desk at 301-621-0390
- Write to:
  NASA STI Help Desk
  NASA Center for AeroSpace Information
  7115 Standard Drive
  Hanover, MD 21076-1320
Systems Engineering Handbook

National Aeronautics and Space Administration
NASA Headquarters
Washington, D.C. 20546

December 2007
# Table of Contents

## Preface

## Acknowledgments

## 1.0 Introduction

1.1 Purpose ........................................................................................................................................ 1  
1.2 Scope and Depth ......................................................................................................................... 1

## 2.0 Fundamentals of Systems Engineering

2.1 The Common Technical Processes and the SE Engine ................................................................. 4  
2.2 An Overview of the SE Engine by Project Phase ......................................................................... 6  
2.3 Example of Using the SE Engine ................................................................................................. 7  
  2.3.1 Detailed Example ................................................................................................................... 8  
  2.3.2 Example Premise .................................................................................................................. 8  
  2.3.2.1 Example Phase A System Design Passes ...................................................................... 8  
  2.3.2.2 Example Product Realization Passes ............................................................................ 12  
  2.3.2.3 Example Use of the SE Engine in Phases B Through D .............................................. 14  
  2.3.2.4 Example Use of the SE Engine in Phases E and F ...................................................... 14  
2.4 Distinctions Between Product Verification and Product Validation ............................................. 15  
2.5 Cost Aspect of Systems Engineering ......................................................................................... 16

## 3.0 NASA Program/Project Life Cycle

3.1 Program Formulation .................................................................................................................... 19  
3.2 Program Implementation ............................................................................................................... 21  
3.3 Project Pre-Phase A: Concept Studies ......................................................................................... 22  
3.4 Project Phase A: Concept and Technology Development ............................................................ 22  
3.5 Project Phase B: Preliminary Design and Technology Completion ............................................ 24  
3.6 Project Phase C: Final Design and Fabrication ........................................................................... 25  
3.7 Project Phase D: System Assembly, Integration and Test, Launch ............................................ 25  
3.8 Project Phase E: Operations and Sustainment ............................................................................ 28  
3.9 Project Phase F: Closeout ............................................................................................................ 28  
3.10 Funding: The Budget Cycle ....................................................................................................... 29

## 4.0 System Design

4.1 Stakeholder Expectations Definition .............................................................................................. 31  
  4.1.1 Process Description ............................................................................................................... 33  
  4.1.1.1 Inputs ............................................................................................................................. 33  
  4.1.1.2 Process Activities .......................................................................................................... 33  
  4.1.1.3 Outputs ......................................................................................................................... 35  
  4.1.2 Stakeholder Expectations Definition Guidance ........................................................................ 35  
  4.1.2.1 Concept of Operations ................................................................................................. 35  
4.2 Technical Requirements Definition .............................................................................................. 40  
  4.2.1 Process Description ............................................................................................................... 40  
  4.2.1.1 Inputs ............................................................................................................................. 41  
  4.2.1.2 Process Activities .......................................................................................................... 41  
  4.2.1.3 Outputs ......................................................................................................................... 41  
  4.2.2 Technical Requirements Definition Guidance ........................................................................ 41  
  4.2.2.1 Types of Requirements................................................................................................. 41

Please purchase PDF Split-Merge on www.verypdf.com to remove this watermark.
Table of Contents

4.2.2.2 Human Factors Engineering Requirements................................................................. 45
4.2.2.3 Requirements Decomposition, Allocation, and Validation ........................................... 45
4.2.2.4 Capturing Requirements and the Requirements Database ............................................ 47
4.2.2.5 Technical Standards .................................................................................................... 47

4.3 Logical Decomposition ....................................................................................................... 49
4.3.1 Process Description ......................................................................................................... 49
  4.3.1.1 Inputs ......................................................................................................................... 49
  4.3.1.2 Process Activities....................................................................................................... 49
  4.3.1.3 Outputs ...................................................................................................................... 51
4.3.2 Logical Decomposition Guidance .................................................................................... 52
  4.3.2.1 Product Breakdown Structure .................................................................................. 52
  4.3.2.2 Functional Analysis Techniques ............................................................................... 52

4.4 Design Solution Definition ................................................................................................ 55
4.4.1 Process Description ......................................................................................................... 55
  4.4.1.1 Inputs ......................................................................................................................... 55
  4.4.1.2 Process Activities....................................................................................................... 56
  4.4.1.3 Outputs ...................................................................................................................... 61
4.4.2 Design Solution Definition Guidance ............................................................................. 62
  4.4.2.1 Technology Assessment ........................................................................................... 62
  4.4.2.2 Integrating Engineering Specialties into the Systems Engineering Process ............... 62

5.0 Product Realization ............................................................................................................ 71
5.1 Product Implementation ..................................................................................................... 73
5.1.1 Process Description ......................................................................................................... 73
  5.1.1.1 Inputs ......................................................................................................................... 73
  5.1.1.2 Process Activities....................................................................................................... 74
  5.1.1.3 Outputs ...................................................................................................................... 75
5.1.2 Product Implementation Guidance .................................................................................. 76
  5.1.2.1 Buying Off-the-Shelf Products ................................................................................ 76
  5.1.2.2 Heritage ..................................................................................................................... 76
5.2 Product Integration ............................................................................................................. 78
5.2.1 Process Description ......................................................................................................... 78
  5.2.1.1 Inputs ......................................................................................................................... 79
  5.2.1.2 Process Activities....................................................................................................... 79
  5.2.1.3 Outputs ...................................................................................................................... 79
5.2.2 Product Integration Guidance .......................................................................................... 80
  5.2.2.1 Integration Strategy .................................................................................................. 80
  5.2.2.2 Relationship to Product Implementation ................................................................. 80
  5.2.2.3 Product/Interface Integration Support ...................................................................... 80
  5.2.2.4 Product Integration of the Design Solution ............................................................... 81
  5.2.2.5 Interface Management ............................................................................................. 81
  5.2.2.6 Compatibility Analysis ............................................................................................. 81
  5.2.2.7 Interface Management Tasks .................................................................................. 81
5.3 Product Verification ........................................................................................................... 83
5.3.1 Process Description ......................................................................................................... 83
  5.3.1.1 Inputs ......................................................................................................................... 83
  5.3.1.2 Process Activities....................................................................................................... 84
  5.3.1.3 Outputs ...................................................................................................................... 89
5.3.2 Product Verification Guidance ........................................................................................ 89
  5.3.2.1 Verification Program ................................................................................................. 89
  5.3.2.2 Verification in the Life Cycle .................................................................................... 89
  5.3.2.3 Verification Procedures ............................................................................................ 92
5.3.2.4 Verification Reports ................................................................................................................. 93
5.3.2.5 End-to-End System Testing ......................................................................................................... 93
5.3.2.6 Modeling and Simulation .............................................................................................................. 96
5.3.2.7 Hardware-in-the-Loop ............................................................................................................... 96
5.4 Product Validation ............................................................................................................................... 98
  5.4.1 Process Description ......................................................................................................................... 98
    5.4.1.1 Inputs ........................................................................................................................................ 98
    5.4.1.2 Process Activities ..................................................................................................................... 99
    5.4.1.3 Outputs .................................................................................................................................... 104
  5.4.2 Product Validation Guidance ........................................................................................................ 104
    5.4.2.1 Modeling and Simulation ......................................................................................................... 104
    5.4.2.2 Software .................................................................................................................................. 104
5.5 Product Transition .............................................................................................................................. 106
  5.5.1 Process Description ......................................................................................................................... 106
    5.5.1.1 Inputs ........................................................................................................................................ 106
    5.5.1.2 Process Activities ..................................................................................................................... 107
    5.5.1.3 Outputs .................................................................................................................................... 109
  5.5.2 Product Transition Guidance ....................................................................................................... 110
    5.5.2.1 Additional Product Transition Input Considerations ................................................................. 110
    5.5.2.2 After Product Transition to the End User—What Next? ........................................................... 110

6.0 Crosscutting Technical Management ................................................................................................. 111

6.1 Technical Planning .............................................................................................................................. 112
  6.1.1 Process Description ......................................................................................................................... 112
    6.1.1.1 Inputs ........................................................................................................................................ 112
    6.1.1.2 Process Activities ..................................................................................................................... 113
    6.1.1.3 Outputs .................................................................................................................................... 122
  6.1.2 Technical Planning Guidance ....................................................................................................... 122
    6.1.2.1 Work Breakdown Structure .................................................................................................... 122
    6.1.2.2 Cost Definition and Modeling .................................................................................................. 125
    6.1.2.3 Lessons Learned ..................................................................................................................... 129
6.2 Requirements Management ................................................................................................................ 131
  6.2.1 Process Description ......................................................................................................................... 131
    6.2.1.1 Inputs ........................................................................................................................................ 131
    6.2.1.2 Process Activities ..................................................................................................................... 132
    6.2.1.3 Outputs .................................................................................................................................... 134
  6.2.2 Requirements Management Guidance .......................................................................................... 134
    6.2.2.1 Requirements Management Plan ............................................................................................. 134
    6.2.2.2 Requirements Management Tools ............................................................................................ 135
6.3 Interface Management ......................................................................................................................... 136
  6.3.1 Process Description ......................................................................................................................... 136
    6.3.1.1 Inputs ........................................................................................................................................ 136
    6.3.1.2 Process Activities ..................................................................................................................... 136
    6.3.1.3 Outputs .................................................................................................................................... 137
  6.3.2 Interface Management Guidance .................................................................................................. 137
    6.3.2.1 Interface Requirements Document ............................................................................................ 137
    6.3.2.2 Interface Control Document or Interface Control Drawing ...................................................... 137
    6.3.2.3 Interface Definition Document .................................................................................................. 138
    6.3.2.4 Interface Control Plan .............................................................................................................. 138
6.4 Technical Risk Management ............................................................................................................... 139
  6.4.1 Process Description ......................................................................................................................... 140
    6.4.1.1 Inputs ........................................................................................................................................ 140
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1.1</td>
<td>Process Activities</td>
<td>138</td>
</tr>
<tr>
<td>6.1.2</td>
<td>Outputs</td>
<td>138</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Technical Risk Management Guidance</td>
<td>139</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Role of Continuous Risk Management in Technical Risk Management</td>
<td>140</td>
</tr>
<tr>
<td>6.2.3</td>
<td>The Interface Between CRM and Risk-Informed Decision Analysis</td>
<td>141</td>
</tr>
<tr>
<td>6.2.4</td>
<td>Selection and Application of Appropriate Risk Methods</td>
<td>142</td>
</tr>
<tr>
<td>6.3.1</td>
<td>Configuration Management</td>
<td>143</td>
</tr>
<tr>
<td>6.3.2</td>
<td>Process Description</td>
<td>144</td>
</tr>
<tr>
<td>6.3.3</td>
<td>Inputs</td>
<td>144</td>
</tr>
<tr>
<td>6.3.4</td>
<td>Process Activities</td>
<td>144</td>
</tr>
<tr>
<td>6.3.5</td>
<td>Outputs</td>
<td>145</td>
</tr>
<tr>
<td>6.4.1</td>
<td>CM Guidance</td>
<td>146</td>
</tr>
<tr>
<td>6.4.2</td>
<td>What Is the Impact of Not Doing CM?</td>
<td>146</td>
</tr>
<tr>
<td>6.4.3</td>
<td>When Is It Acceptable to Use Redline Drawings?</td>
<td>147</td>
</tr>
<tr>
<td>6.5.1</td>
<td>Technical Data Management</td>
<td>148</td>
</tr>
<tr>
<td>6.5.2</td>
<td>Process Description</td>
<td>149</td>
</tr>
<tr>
<td>6.5.3</td>
<td>Outputs</td>
<td>149</td>
</tr>
<tr>
<td>6.6.1</td>
<td>Technical Data Management</td>
<td>150</td>
</tr>
<tr>
<td>6.6.2</td>
<td>Technical Data Management Guidance</td>
<td>150</td>
</tr>
<tr>
<td>6.7.1</td>
<td>Technical Assessment</td>
<td>151</td>
</tr>
<tr>
<td>6.7.2</td>
<td>Technical Assessment Guidance</td>
<td>151</td>
</tr>
<tr>
<td>6.8.1</td>
<td>Decision Analysis</td>
<td>152</td>
</tr>
<tr>
<td>6.8.2</td>
<td>Decision Analysis Guidance</td>
<td>153</td>
</tr>
<tr>
<td>7.0</td>
<td>Special Topics</td>
<td>154</td>
</tr>
<tr>
<td>7.1</td>
<td>Engineering with Contracts</td>
<td>155</td>
</tr>
<tr>
<td>7.1.1</td>
<td>Introduction, Purpose, and Scope</td>
<td>156</td>
</tr>
<tr>
<td>7.1.2</td>
<td>Acquisition Strategy</td>
<td>156</td>
</tr>
<tr>
<td>7.1.3</td>
<td>Prior to Contract Award</td>
<td>157</td>
</tr>
</tbody>
</table>
Table of Contents

7.4.1 Basic HF Model .................................................................................................................. 247
7.4.2 HF Analysis and Evaluation Techniques ........................................................................... 247
7.5 Environmental, Nuclear Safety, Planetary Protection, and Asset Protection Policy Compliance ........ 256
  7.5.1 NEPA and EO 12114 ......................................................................................................... 256
  7.5.1.1 National Environmental Policy Act .............................................................................. 256
  7.5.1.2 EO 12114 Environmental Effects Abroad of Major Federal Actions ...................... 257
7.5.2 PD/NSC-25 ....................................................................................................................... 257
7.5.3 Planetary Protection ........................................................................................................ 258
7.5.4 Space Asset Protection .................................................................................................... 260
  7.5.4.1 Protection Policy ......................................................................................................... 260
  7.5.4.2 Goal .......................................................................................................................... 260
  7.5.4.3 Scoping ..................................................................................................................... 260
  7.5.4.4 Protection Planning ................................................................................................. 260
7.6 Use of Metric System .......................................................................................................... 261

Appendix A: Acronyms .............................................................................................................. 263
Appendix B: Glossary ................................................................................................................ 266
Appendix C: How to Write a Good Requirement ....................................................................... 279
Appendix D: Requirements Verification Matrix ......................................................................... 282
Appendix E: Creating the Validation Plan (Including Validation Requirements Matrix) ............... 284
Appendix F: Functional, Timing, and State Analysis .................................................................. 285
Appendix G: Technology Assessment/Insertion ....................................................................... 293
Appendix H: Integration Plan Outline ....................................................................................... 299
Appendix I: Verification and Validation Plan Sample Outline .................................................. 301
Appendix J: SEMP Content Outline ........................................................................................ 303
Appendix K: Plans ..................................................................................................................... 308
Appendix L: Interface Requirements Document Outline .......................................................... 309
Appendix M: CM Plan Outline ................................................................................................ 311
Appendix N: Guidance on Technical Peer Reviews/Inspections .............................................. 312
Appendix O: Tradeoff Examples ................................................................................................ 316
Appendix P: SOW Review Checklist ........................................................................................ 317
Appendix Q: Project Protection Plan Outline ............................................................................ 321
References .................................................................................................................................. 323
Bibliography .............................................................................................................................. 327
Index ........................................................................................................................................... 332
Figures

2.0-1 SE in context of overall project management .......................................................... 4
2.1-1 The systems engineering engine .............................................................................. 5
2.2-1 A miniaturized conceptualization of the poster-size NASA project life-cycle process flow for flight and ground systems accompanying this handbook .............................................. 6
2.3-1 SE engine tracking icon ......................................................................................... 8
2.3-2 Product hierarchy, tier 1: first pass through the SE engine ..................................... 9
2.3-3 Product hierarchy, tier 2: external tank .................................................................. 10
2.3-4 Product hierarchy, tier 2: orbiter ............................................................................ 10
2.3-5 Product hierarchy, tier 3: avionics system ............................................................... 11
2.3-6 Product hierarchy: complete pass through system design processes side of the SE engine .................................................................................................................. 11
2.3-7 Model of typical activities during operational phase (Phase E) of a product ........... 14
2.3-8 New products or upgrades reentering the SE engine ............................................. 15
2.5-1 The enveloping surface of nondominated designs .................................................. 16
2.5-2 Estimates of outcomes to be obtained from several design concepts including uncertainty ................................................................. 17
3.0-1 NASA program life cycle ....................................................................................... 20
3.0-2 NASA project life cycle ......................................................................................... 20
3.10-1 Typical NASA budget cycle ................................................................................. 29
4.0-1 Interrelationships among the system design processes ........................................... 31
4.1-1 Stakeholder Expectations Definition Process .......................................................... 33
4.1-2 Product flow for stakeholder expectations ............................................................. 34
4.1-3 Typical ConOps development for a science mission ............................................. 36
4.1-4 Example of an associated end-to-end operational architecture ............................. 36
4.1-5a Example of a lunar sortie timeline developed early in the life cycle .................... 37
4.1-5b Example of a lunar sortie DRM early in the life cycle ........................................... 37
4.1-6 Example of a more detailed, integrated timeline later in the life cycle for a science mission .............................................................................................................. 38
4.2-1 Technical Requirements Definition Process .......................................................... 40
4.2-2 Characteristics of functional, operational, reliability, safety, and specialty requirements ........................................................................................................... 43
4.2-3 The flowdown of requirements .............................................................................. 46
4.2-4 Allocation and flowdown of science pointing requirements .................................... 47
4.3-1 Logical Decomposition Process ............................................................................ 49
4.3-2 Example of a PBS ................................................................................................... 52
4.3-3 Example of a functional flow block diagram ......................................................... 53
4.3-4 Example of an N2 diagram ..................................................................................... 54
4.4-1 Design Solution Definition Process ....................................................................... 55
4.4-2 The doctrine of successive refinement .................................................................... 56
4.4-3 A quantitative objective function, dependent on life-cycle cost and all aspects of effectiveness .......................................................... 58
5.0-1 Product realization .................................................................................................. 71
5.1-1 Product Implementation Process ............................................................................ 73
5.2-1 Product Integration Process .................................................................................. 78
5.3-1 Product Verification Process .................................................................................. 84
5.3-2 Bottom-up realization process .............................................................................. 90
5.3-3 Example of end-to-end data flow for a scientific satellite mission ....................... 94
5.4-1 Product Validation Process .................................................................................... 99
5.5-1 Product Transition Process .................................................................................... 106
6.1-1 Technical Planning Process ................................................................................... 112
6.1-2 Activity-on-arrow and precedence diagrams for network schedules .................... 116
6.1-3 Gantt chart ........................................................................................................... 118
6.1-4 Relationship between a system, a PBS, and a WBS ............................................. 123
6.1-5 Examples of WBS development errors ................................................................... 125
6.2-1 Requirements Management Process ..................................................................... 131
6.3-1 Interface Management Process ............................................................................. 136

Please purchase PDF Split-Merge on www.verypdf.com to remove this watermark.
# Table of Contents

6.4-1 Technical Risk Management Process ................................................................. 140
6.4-2 Scenario-based modeling of hazards ................................................................. 141
6.4-3 Risk as a set of triplets ...................................................................................... 141
6.4-4 Continuous risk management ........................................................................... 142
6.4-5 The interface between CRM and risk-informed decision analysis ............... 143
6.4-6 Risk analysis of decision alternatives ............................................................... 144
6.4-7 Risk matrix ....................................................................................................... 145
6.4-8 Example of a fault tree ..................................................................................... 146
6.4-9 Deliberation ...................................................................................................... 147
6.4-10 Performance monitoring and control of deviations ....................................... 149
6.4-11 Margin management method ......................................................................... 150
6.5-1 CM Process ...................................................................................................... 151
6.5-2 Five elements of configuration management .................................................. 152
6.5-3 Evolution of technical baseline ....................................................................... 153
6.5-4 Typical change control process ...................................................................... 155
6.5-5 Technical Data Management Process .............................................................. 158
6.7-1 Technical Assessment Process ....................................................................... 166
6.7-2 Planning and status reporting/feedback loop ................................................ 167
6.7-3 Cost and schedule variances .......................................................................... 190
6.7-4 Relationships of MOEs, MOPs, and TPMs ....................................................... 192
6.7-5 Use of the planned profile method for the weight TPM with rebaseline in Chandra Project .......................................................... 194
6.7-6 Use of the margin management method for the mass TPM in Sojourner .......... 194
6.8-1 Decision Analysis Process ............................................................................. 198
6.8-2 Example of a decision matrix ......................................................................... 201
6.8-3 Systems analysis across the life cycle ............................................................. 203
6.8-4 Simulation model analysis techniques .............................................................. 204
6.8-5 Trade study process ......................................................................................... 205
6.8-6 Influence diagrams ........................................................................................ 210
6.8-7 Decision tree ................................................................................................... 211
6.8-8 Utility function for a “volume” performance measure ..................................... 213
6.8-9 Risk-informed Decision Analysis Process ..................................................... 214
6.8-10 Example of an objectives hierarchy ............................................................... 215
7.1-1 Acquisition life cycle ...................................................................................... 218
7.1-2 Contract requirements development process ................................................. 223
7.1-3 CACE people/process/tools/facility paradigm ............................................... 234
7.1-4 Human factors interaction model ................................................................. 247
7.4-2 HF engineering process and its links to the NASA program/project life cycle .... 248
F-1 FFBD flowdown ................................................................................................ 286
F-2 FFBD: example 1 .............................................................................................. 287
F-3 FFBD showing additional control constructs: example 2 ............................... 287
F-4 Enhanced FFBD: example 3 ............................................................................ 288
F-5 Requirements allocation sheet .......................................................................... 289
F-6 N2 diagram for orbital equipment .................................................................... 289
F-7 Timing diagram example ................................................................................ 290
F-8 Slew command status state diagram ............................................................... 291
G-1 PBS example .................................................................................................... 294
G-2 Technology assessment process .................................................................. 295
G-3 Architectural studies and technology development ...................................... 296
G-4 Technology readiness levels ........................................................................... 296
G-5 The TMA thought process .............................................................................. 297
G-6 TRL assessment matrix .................................................................................. 298
N-1 The peer review/inspection process .............................................................. 312
N-2 Peer reviews/inspections quick reference guide

Tables

2.3-1 Project Life-Cycle Phases ................................................................. 7
4.1-1 Typical Operational Phases for a NASA Mission ............................ 39
4.2-1 Benefits of Well-Written Requirements ........................................ 42
4.2-2 Requirements Metadata ................................................................. 48
4.4-1 ILS Technical Disciplines ............................................................... 66
6.6-1 Technical Data Tasks .................................................................. 163
6.7-1 Program Technical Reviews ........................................................... 170
6.7-2 P/SRR Entrance and Success Criteria ............................................ 171
6.7-3 P/SDR Entrance and Success Criteria ........................................... 172
6.7-4 MCR Entrance and Success Criteria ............................................. 173
6.7-5 SRR Entrance and Success Criteria .............................................. 174
6.7-6 MDR Entrance and Success Criteria ............................................ 175
6.7-7 SDR Entrance and Success Criteria .............................................. 176
6.7-8 PDR Entrance and Success Criteria ............................................. 177
6.7-9 CDR Entrance and Success Criteria ............................................ 178
6.7-10 PRR Entrance and Success Criteria ............................................ 179
6.7-11 SIR Entrance and Success Criteria ............................................. 180
6.7-12 TRR Entrance and Success Criteria ............................................ 181
6.7-13 SAR Entrance and Success Criteria ............................................ 182
6.7-14 ORR Entrance and Success Criteria ............................................ 183
6.7-15 FRR Entrance and Success Criteria ............................................ 184
6.7-16 PLAR Entrance and Success Criteria ......................................... 185
6.7-17 CERR Entrance and Success Criteria ....................................... 186
6.7-18 PFAR Entrance and Success Criteria ......................................... 186
6.7-19 DR Entrance and Success Criteria ............................................. 187
6.7-20 Functional and Physical Configuration Audits .............................. 189
6.7-21 Systems Engineering Process Metrics ........................................ 196
6.8-1 Consequence Table ..................................................................... 199
6.8-2 Typical Information to Capture in a Decision Report ....................... 202
7.1-1 Applying the Technical Processes on Contract ............................... 220
7.1-2 Steps in the Requirements Development Process ......................... 224
7.1-3 Proposal Evaluation Criteria ....................................................... 227
7.1-4 Risks in Acquisition .................................................................... 228
7.1-5 Typical Work Product Documents .............................................. 230
7.1-6 Contract-Subcontract Issues ......................................................... 231
7.4-1 Human and Organizational Analysis Techniques ............................ 249
7.5-1 Planetary Protection Mission Categories ....................................... 259
7.5-2 Summarized Planetary Protection Requirements ........................... 259
D-1 Requirements Verification Matrix .................................................. 283
E-1 Validation Requirements Matrix ..................................................... 284
G-1 Products Provided by the TA as a Function of Program/Project Phase .. 294
H-1 Integration Plan Contents ................................................................ 300
M-1 CM Plan Outline .......................................................................... 311
O-1 Typical Tradeoffs for Space Systems ............................................. 316
O-2 Typical Tradeoffs in the Acquisition Process .................................. 316
O-3 Typical Tradeoffs Throughout the Project Life Cycle ........................ 316
**Table of Contents**

**Boxes**

System Cost, Effectiveness, and Cost-Effectiveness ................................................................. 16  
The Systems Engineer’s Dilemma ............................................................................................... 17  
Program Formulation ............................................................................................................ 21  
Program Implementation ........................................................................................................ 21  
Pre-Phase A: Concept Studies ................................................................................................ 22  
Phase A: Concept and Technology Development ................................................................. 23  
Phase B: Preliminary Design and Technology Completion .................................................. 24  
Phase C: Final Design and Fabrication .................................................................................... 26  
Phase D: System Assembly, Integration and Test, Launch .................................................. 27  
Phase E: Operations and Sustainment ..................................................................................... 28  
Phase F: Closeout .................................................................................................................. 28  
System Design Keys .............................................................................................................. 32  
Example of Functional and Performance Requirements ....................................................... 43  
Rationale .................................................................................................................................. 48  
DOD Architecture Framework ................................................................................................. 51  
Prototypes ................................................................................................................................ 67  
Product Realization Keys ......................................................................................................... 72  
Differences Between Verification and Validation Testing ...................................................... 83  
Types of Testing .................................................................................................................... 85  
Types of Verification .............................................................................................................. 86  
Differences Between Verification and Validation Testing ...................................................... 98  
Types of Validation ................................................................................................................. 100  
Examples of Enabling Products and Support Resources for Preparing to Conduct Validation .......................................................................................................................... 102  
Model Verification and Validation .......................................................................................... 104  
Crosscutting Technical Management Keys ........................................................................... 111  
Gantt Chart Features .............................................................................................................. 117  
WBS Hierarchies for Systems ................................................................................................. 126  
Definitions .............................................................................................................................. 132  
Typical Interface Management Checklist ............................................................................ 138  
Key Concepts in Technical Risk Management ......................................................................... 139  
Example Sources of Risk ........................................................................................................ 145  
Limitations of Risk Matrices .................................................................................................. 145  
Types of Configuration Change Management Changes ...................................................... 154  
Warning Signs/Red Flags (How Do You Know When You’re in Trouble?) ............................. 156  
Redlines Were identified as One of the Major Causes of the NOAA N-Prime Mishap ............ 157  
Inappropriate Uses of Technical Data .................................................................................... 160  
Data Collection Checklist ...................................................................................................... 162  
Termination Review .............................................................................................................. 169  
Analyzing the Estimate at Completion .................................................................................. 191  
Examples of Technical Performance Measures .................................................................... 193  
An Example of a Trade Tree for a Mars Rover ...................................................................... 207  
Trade Study Reports .............................................................................................................. 208  
Solicitations ............................................................................................................................ 219  
Source Evaluation Board ........................................................................................................ 226  
Context Diagrams .................................................................................................................. 292
Preface

Since the writing of NASA/SP-6105 in 1995, systems engineering at the National Aeronautics and Space Administration (NASA), within national and international standard bodies, and as a discipline has undergone rapid evolution. Changes include implementing standards in the International Organization for Standardization (ISO) 9000, the use of Carnegie Mellon Software Engineering Institute's Capability Maturity Model Integration (CMMI®) to improve development and delivery of products, and the impacts of mission failures. Lessons learned on systems engineering were documented in reports such as those by the NASA Integrated Action Team (NIAT), the Columbia Accident Investigation Board (CAIB), and the follow-on Diaz Report. Out of these efforts came the NASA Office of the Chief Engineer (OCE) initiative to improve the overall Agency systems engineering infrastructure and capability for the efficient and effective engineering of NASA systems, to produce quality products, and to achieve mission success. In addition, Agency policy and requirements for systems engineering have been established. This handbook update is a part of the OCE-sponsored Agencywide systems engineering initiative.

In 1995, SP-6105 was initially published to bring the fundamental concepts and techniques of systems engineering to NASA personnel in a way that recognizes the nature of NASA systems and the NASA environment. This revision of SP-6105 maintains that original philosophy while updating the Agency's systems engineering body of knowledge, providing guidance for insight into current best Agency practices, and aligning the handbook with the new Agency systems engineering policy.

The update of this handbook was twofold: a top-down compatibility with higher level Agency policy and a bottom-up infusion of guidance from the NASA practitioners in the field. The approach provided the opportunity to obtain best practices from across NASA and bridge the information to the established NASA systems engineering process. The attempt is to communicate principles of good practice as well as alternative approaches rather than specify a particular way to accomplish a task. The result embodied in this handbook is a top-level implementation approach on the practice of systems engineering unique to NASA. The material for updating this handbook was drawn from many different sources, including NASA procedural requirements, field center systems engineering handbooks and processes, as well as non-NASA systems engineering textbooks and guides.

This handbook consists of six core chapters: (1) systems engineering fundamentals discussion, (2) the NASA program/project life cycles, (3) systems engineering processes to get from a concept to a design, (4) systems engineering processes to get from a design to a final product, (5) crosscutting management processes in systems engineering, and (6) special topics relative to systems engineering. These core chapters are supplemented by appendices that provide outlines, examples, and further information to illustrate topics in the core chapters. The handbook makes extensive use of boxes and figures to define, refine, illustrate, and extend concepts in the core chapters without diverting the reader from the main information.

The handbook provides top-level guidelines for good systems engineering practices; it is not intended in any way to be a directive.

Acknowledgments


The following individuals are recognized as contributing practitioners to the content of this handbook revision:

■ Core Team Member (or Representative) from Center, Directorate, or Office
◆ Integration Team Member
● Subject Matter Expert Team Champion
▲ Subject Matter Expert

Arden Acord, NASA/Jet Propulsion Laboratory ▲
Danette Allen, NASA/Langley Research Center ▲
Deborah Amato, NASA/Goddard Space Flight Center ▲ ●
Jim Andary, NASA/Goddard Space Flight Center ▲ ●
Tim Beard, NASA/Ames Research Center ▲
Jim Bilbro, NASA/Marshall Space Flight Center ▲
Mike Blythe, NASA/Headquarters ■
Linda Bromley, NASA/Johnson Space Center ● ● ■
Dave Brown, Defense Acquisition University ▲
John Brunson, NASA/Marshall Space Flight Center ● ●
Joe Burt, NASA/Goddard Space Flight Center ▲
Glenn Campbell, NASA/Headquarters ▲
Joyce Carpenter, NASA/Johnson Space Center ● ●
Keith Chamberlin, NASA/Goddard Space Flight Center ▲
Peggy Chun, NASA/NASA Engineering and Safety Center ● ● ▲ ■
Cindy Coker, NASA/Marshall Space Flight Center ▲
Nita Congress, Graphic Designer ●
Catharine Conley, NASA/Headquarters ▲
Shelley Delay, NASA/Marshall Space Flight Center ▲
Rebecca Deschamp, NASA/Stennis Space Center ▲
Homayoon Dezfuli, NASA/Headquarters ● ●
Olga Dominguez, NASA/Headquarters ▲
Rajiv Doreswamy, NASA/Headquarters ●
Larry Dyer, NASA/Johnson Space Center ▲
Nelson Eng, NASA/Johnson Space Center ▲
Patricia Eng, NASA/Headquarters ▲
Amy Epps, NASA/Marshall Space Flight Center ▲
Chester Everline, NASA/Jet Propulsion Laboratory ▲
Karen Fashimpaur, Arctic Slope Regional Corporation ● ▲
Orlando Figueroa, NASA/Goddard Space Flight Center ■
Stanley Fishkind, NASA/Headquarters ■
Brad Flick, NASA/Dryden Flight Research Center ■
Marton Forkosh, NASA/Glenn Research Center ■
Dan Freund, NASA/Johnson Space Center ▲
Greg Galbreath, NASA/Johnson Space Center ▲
Louie Galland, NASA/Langley Research Center ▲
Yuri Gawdiak, NASA/Headquarters ● ● ▲
Theresa Gibson, NASA/Glenn Research Center ▲
Ronnie Gillian, NASA/Langley Research Center ▲
Julius Giriunas, NASA/Glenn Research Center ▲
Ed Gollop, NASA/Marshall Space Flight Center ▲
Lee Graham, NASA/Johnson Space Center ▲
Larry Green, NASA/Langley Research Center ▲
Owen Greulich, NASA/Headquarters ■
Ben Hanel, NASA/Ames Research Center ▲
Gena Henderson, NASA/Kennedy Space Center ● ▲
Amy Hemken, NASA/Marshall Space Flight Center ▲
Bob Hennessy, NASA/NASA Engineering and Safety Center ▲
Ellen Herring, NASA/Goddard Space Flight Center ● ■
Renee Hugger, NASA/Johnson Space Center ▲
Brian Hughitt, NASA/Headquarters ▲
Eric Isaac, NASA/Goddard Space Flight Center ■
Tom Jacks, NASA/Stennis Space Center ▲
Ken Johnson, NASA/NASA Engineering and Safety Center ▲
Ross Jones, NASA/Jet Propulsion Laboratory ■
John Juhasz, NASA/Johnson Space Center ▲
Stephen Kapurch, NASA/Headquarters ● ● ●
Jason Kastner, NASA/Jet Propulsion Laboratory ▲
Kristen Kehrer, NASA/Kennedy Space Center ▲
John Kelly, NASA/Headquarters ▲
Kriss Kennedy, NASA/Johnson Space Center ▲
Acknowledgments

Steven Kennedy, NASA/Kennedy Space Center
Tracey Kickbusch, NASA/Kennedy Space Center
Casey Kirchner, NASA/Stennis Space Center
Kenneth Kumor, NASA/Headquarters
Janne Lady, SAITECH/CSC
Jerry Lake, Systems Management International
Kenneth W. Ledbetter, NASA/Headquarters
Janine Lady, SAITECH/CSC
Steve Leete, NASA/Goddard Space Flight Center
William Lincoln, NASA/Jet Propulsion Laboratory
Dave Littman, NASA/Goddard Space Flight Center
John Lucero, NASA/Glenn Research Center
Paul Luz, NASA/Johnson Space Center
Todd MacLeod, NASA/Johnson Space Center
Roger Mathews, NASA/Kennedy Space Center
Bryon Maynard, NASA/Stennis Space Center
Patrick McDuffee, NASA/Johnson Space Center
Mark McElyea, NASA/Johnson Space Center
William McGovern, Defense Acquisition University
Colleen McGraw, NASA/Goddard Space Flight Center
Melissa McGuire, NASA/Glenn Research Center
Don Mendoza, NASA/Ames Research Center
Leila Meshkat, NASA/Jet Propulsion Laboratory
Elizabeth Messer, NASA/Stennis Space Center
Chuck Miller, NASA/Headquarters
Scott Mimbs, NASA/Kennedy Space Center
Steve Newton, NASA/Johnson Space Center
Tri Nguyen, NASA/Johnson Space Center
Chuck Niles, NASA/Langley Research Center
Cynthia Null, NASA/NASA Engineering and Safety Center
John Olson, NASA/Headquarters
Tim Olson, QIC, Inc.
Sam Padgett, NASA/Johnson Space Center
Christine Powell, NASA/Stennis Space Center
Steve Prahst, NASA/Glenn Research Center
Pete Prassinos, NASA/Headquarters
Mark Prill, NASA/Johnson Space Center
Neil Ray, NASA/Johnson Space Center
Ron Ray, NASA/Dryden Flight Research Center
Gary Rawitscher, NASA/Headquarters
Joshua Reinert, ISL Inc.
Norman Rioux, NASA/Goddard Space Flight Center
Steve Robbins, NASA/Johnson Space Center
Dennis Rohn, NASA/Glenn Research Center
Jim Rose, NASA/Jet Propulsion Laboratory
Arnie Ruskin,* NASA/Jet Propulsion Laboratory
Harry Ryan, NASA/Stennis Space Center
George Salazar, NASA/Johnson Space Center
Nina Scheller, NASA/Ames Research Center
Pat Schuler, NASA/Langley Research Center
Randy Seftas, NASA/Goddard Space Flight Center
Joey Shelton, NASA/Johnson Space Center
Robert Shishko, NASA/Jet Propulsion Laboratory
Burton Sigal, NASA/Jet Propulsion Laboratory
Sandra Small, NASA/Headquarters
Richard Smith, NASA/Kennedy Space Center
John Snoderly, Defense Acquisition University
Richard Sorge, NASA/Glenn Research Center
Michael Stamatakis, NASA/Headquarters
Tom Sutliff, NASA/Glenn Research Center
Todd Tofl, NASA/Glenn Research Center
John Tinsley, NASA/Headquarters
Rob Traister, Graphic Designer
Clayton Turner, NASA/Langley Research Center
Paul VanDamme, NASA/Jet Propulsion Laboratory
Karen Vaner, NASA/Stennis Space Center
Lynn Vernon, NASA/Johnson Space Center
Linda Voss, Technical Writer
Britt Walters, NASA/Kennedy Space Center
Tommy Watts, NASA/Johnson Space Center
Richard Weinstein, NASA/Headquarters
Katie Weiss, NASA/Jet Propulsion Laboratory
Martha Wetherholt, NASA/Headquarters
Becky Wheeler, NASA/Jet Propulsion Laboratory
Cathy White, NASA/Johnson Space Center
Reed Wilcox, NASA/Johnson Space Center
Barbara Woolford, NASA/Johnson Space Center
Felicia Wright, NASA/Langley Research Center
Robert Younghblood, ISL Inc.
Tom Zang, NASA/Langley Research Center

*In memory of.
1.0 Introduction

1.1 Purpose
This handbook is intended to provide general guidance and information on systems engineering that will be useful to the NASA community. It provides a generic description of Systems Engineering (SE) as it should be applied throughout NASA. A goal of the handbook is to increase awareness and consistency across the Agency and advance the practice of SE. This handbook provides perspectives relevant to NASA and data particular to NASA.

This handbook should be used as a companion for implementing NPR 7123.1, Systems Engineering Processes and Requirements, as well as the Center-specific handbooks and directives developed for implementing systems engineering at NASA. It provides a companion reference book for the various systems engineering related courses being offered under NASA's auspices.

1.2 Scope and Depth
The coverage in this handbook is limited to general concepts and generic descriptions of processes, tools, and techniques. It provides information on systems engineering best practices and pitfalls to avoid. There are many Center-specific handbooks and directives as well as textbooks that can be consulted for in-depth tutorials.

This handbook describes systems engineering as it should be applied to the development and implementation of large and small NASA programs and projects. NASA has defined different life cycles that specifically address the major project categories, or product lines, which are: Flight Systems and Ground Support (FS&GS), Research and Technology (R&T), Construction of Facilities (CoF), and Environmental Compliance and Restoration (ECR). The technical content of the handbook provides systems engineering best practices that should be incorporated into all NASA product lines. (Check the NASA On-Line Directives Information System (NODIS) electronic document library for applicable NASA directives on topics such as product lines.) For simplicity this handbook uses the FS&GS product line as an example. The specifics of FS&GS can be seen in the description of the life cycle and the details of the milestone reviews. Each product line will vary in these two areas; therefore, the reader should refer to the applicable NASA procedural requirements for the specific requirements for their life cycle and reviews. The engineering of NASA systems requires a systematic and disciplined set of processes that are applied recursively and iteratively for the design, development, operation, maintenance, and closeout of systems throughout the life cycle of the programs and projects.

The handbook's scope properly includes systems engineering functions regardless of whether they are performed by a manager or an engineer, in-house, or by a contractor.
Systems engineering is a methodical, disciplined approach for the design, realization, technical management, operations, and retirement of a system. A “system” is a construct or collection of different elements that together produce results not obtainable by the elements alone. The elements, or parts, can include people, hardware, software, facilities, policies, and documents; that is, all things required to produce system-level results. The results include system-level qualities, properties, characteristics, functions, behavior, and performance. The value added by the system as a whole, beyond that contributed independently by the parts, is primarily created by the relationship among the parts; that is, how they are interconnected. It is a way of looking at the “big picture” when making technical decisions. It is a way of achieving stakeholder functional, physical, and operational performance requirements in the intended use environment over the planned life of the systems. In other words, systems engineering is a logical way of thinking.

Systems engineering is the art and science of developing an operable system capable of meeting requirements within often opposed constraints. Systems engineering is a holistic, integrative discipline, wherein the contributions of structural engineers, electrical engineers, mechanism designers, power engineers, human factors engineers, and many more disciplines are evaluated and balanced, one against another, to produce a coherent whole that is not dominated by the perspective of a single discipline.

Systems engineering seeks a safe and balanced design in the face of opposing interests and multiple, sometimes conflicting constraints. The systems engineer must develop the skill and instinct for identifying and focusing efforts on assessments to optimize the overall design and not favor one system/subsystem at the expense of another. The art is in knowing when and where to probe. Personnel with these skills are usually tagged as “systems engineers.” They may have other titles—lead systems engineer, technical manager, chief engineer—but for this document, we will use the term systems engineer.

The exact role and responsibility of the systems engineer may change from project to project depending on the size and complexity of the project and from phase to phase of the life cycle. For large projects, there may be one or more systems engineers. For small projects, sometimes the project manager may perform these practices. But, whoever assumes those responsibilities, the systems engineering functions must be performed. The actual assignment of the roles and responsibilities of the named systems engineer may also therefore vary. The lead systems engineer ensures that the system technically fulfills the defined needs and requirements and that a proper systems engineering approach is being followed. The systems engineer oversees the project’s systems engineering activities as performed by the technical team and directs, communicates, monitors, and coordinates tasks. The systems engineer reviews and evaluates the technical aspects of the project to ensure that the systems/subsystems engineering processes are functioning properly and evolves the system from concept to product. The entire technical team is involved in the systems engineering process.

The systems engineer will usually play the key role in leading the development of the system architecture, defining and allocating requirements, evaluating design tradeoffs, balancing technical risk between systems, defining and assessing interfaces, providing oversight of verification and validation activities, as well as many other tasks. The systems engineer will usually have the prime responsibility in developing many of the project documents, including the Systems Engineering Management Plan (SEMP), requirements/specification documents, verification and validation documents, certification packages, and other technical documentation.

---

2Comments on systems engineering throughout Chapter 2.0 are extracted from the speech “System Engineering and the Two Cultures of Engineering” by Michael D. Griffin, NASA Administrator.
In summary, the systems engineer is skilled in the art and science of balancing organizational and technical interactions in complex systems. However, since the entire team is involved in the systems engineering approach, in some ways everyone is a systems engineer. Systems engineering is about tradeoffs and compromises, about generalists rather than specialists. Systems engineering is about looking at the “big picture” and not only ensuring that they get the design right (meet requirements) but that they get the right design.

To explore this further, put SE in the context of project management. As discussed in NPR 7120.5, NASA Space Flight Program and Project Management Requirements, project management is the function of planning, overseeing, and directing the numerous activities required to achieve the requirements, goals, and objectives of the customer and other stakeholders within specified cost, quality, and schedule constraints. Project management can be thought of as having two major areas of emphasis, both of equal weight and importance. These areas are systems engineering and project control. Figure 2.0-1 is a notional graphic depicting this concept. Note that there are areas where the two cornerstones of project management overlap. In these areas, SE provides the technical aspects or inputs; whereas project control provides the programmatic, cost, and schedule inputs.

This document will focus on the SE side of the diagram. These practices/processes are taken from NPR 7123.1, NASA Systems Engineering Processes and Requirements. Each will be described in much greater detail in subsequent chapters of this document, but an overview is given below.

### 2.1 The Common Technical Processes and the SE Engine

There are three sets of common technical processes in NPR 7123.1, NASA Systems Engineering Processes and Requirements: system design, product realization, and technical management. The processes in each set and their interactions and flows are illustrated by the NPR systems engineering “engine” shown in Figure 2.1-1. The processes of the SE engine are used to develop and realize the end products. This chapter provides the application context of the 17 common technical processes required in NPR 7123.1. The system design processes, the product realization processes, and the technical management processes are discussed in more details in Chapters 4.0, 5.0, and 6.0, respectively. Steps 1 through 9 indicated in Figure 2.1-1 represent the tasks in execution of a project. Steps 10 through 17 are crosscutting tools for carrying out the processes.

- **System Design Processes**: The four system design processes shown in Figure 2.1-1 are used to define and baseline stakeholder expectations, generate and baseline technical requirements, and convert the technical requirements into a design solution that will satisfy the baseline baseline stakeholder expectations. These processes are applied to each product of the system structure from the top of the structure to the bottom until the lowest products in any system structure branch are defined to the point where they can be built, bought, or reused. All other products in the system structure are realized by integration. Designers not only develop the design solutions to the products intended to perform the operational functions of the system, but also
establish requirements for the products and services that enable each operational/mission product in the system structure.

- **Product Realization Processes**: The product realization processes are applied to each operational/mission product in the system structure starting from the lowest level product and working up to higher level integrated products. These processes are used to create the design solution for each product (e.g., by the Product Implementation or Product Integration Process) and to verify, validate, and transition up to the next hierarchical level products that satisfy their design solutions and meet stakeholder expectations as a function of the applicable life-cycle phase.

- **Technical Management Processes**: The technical management processes are used to establish and evolve technical plans for the project, to manage communication across interfaces, to assess progress against the plans and requirements for the system products or services, to control technical execution of the project through to completion, and to aid in the decisionmaking process.

The processes within the SE engine are used both iteratively and recursively. As defined in NPR 7123.1, “iterative” is the “application of a process to the same product or set of products to correct a discovered discrepancy or other variation from requirements,” whereas “recursive” is defined as adding value to the system “by the repeated application of processes to design next lower layer system products or to realize next upper layer end products within the system structure. This also applies to repeating application of the same processes to the system structure in the next life-cycle phase to mature the system definition and satisfy phase success criteria.” The example used in Section 2.3 will further explain these concepts. The technical processes are applied recursively and iteratively to break down the initializing concepts of the system to a level of detail concrete enough that the technical team can implement a product from the information. Then the processes are applied recursively and
iteratively to integrate the smallest product into greater and larger systems until the whole of the system has been assembled, verified, validated, and transitioned.

2.2 An Overview of the SE Engine by Project Phase

Figure 2.2-1 conceptually illustrates how the SE engine is used during each of the seven phases of a project. Figure 2.2-1 is a conceptual diagram. For all of the details, refer to the poster version of this figure, which accompanies this handbook.

The uppermost horizontal portion of this chart is used as a reference to project system maturity, as the project progresses from a feasible concept to an as-deployed system; phase activities; Key Decision Points (KDPs); and major project reviews.

The next major horizontal band shows the technical development processes (steps 1 through 9) in each project phase. The systems engineering engine cycles five times from Pre-Phase A through Phase D. Please note that NASA’s management has structured Phases C and D to “split” the technical development processes in half in Phases C and D to ensure closer management control. The engine is bound by a dashed line in Phases C and D.

Once a project enters into its operational state (Phase E) and closes with a closeout phase (Phase F), the technical work shifts to activities commensurate with these last two project phases.

The next major horizontal band shows the eight technical management processes (steps 10 through 17) in each project phase. The SE engine cycles the technical management processes seven times from Pre-Phase A through Phase F.

Each of the engine entries is given a 6105 paragraph label that is keyed to Chapters 4.0, 5.0, and 6.0 in this handbook. For example, in the technical development processes, “Get Stakeholder Expectations” discussions and details are in Section 4.1.
2.3 Example of Using the SE Engine

To help in understanding how the SE engine is applied, an example will be posed and walked through the processes. Pertinent to this discussion are the phases of the program and project life cycles, which will be discussed in greater depth in Chapter 3.0 of this document. As described in Chapter 3.0, NPR 7120.5 defines the life cycle used for NASA programs and projects. The life-cycle phases are described in Table 2.3-1.

Use of the different phases of a life cycle allows the various products of a project to be gradually developed and matured from initial concepts through the fielding of the product and to its final retirement. The SE engine shown in Figure 2.1-1 is used throughout all phases.

In Pre-Phase A, the SE engine is used to develop the initial concepts; develop a preliminary/draft set of key high-level requirements; realize these concepts through modeling, mockups, simulation, or other means; and verify and validate that these concepts and products would be able to meet the key high-level requirements. Note that this is not the formal verification and validation program that will be performed on the final product but is a methodical runthrough ensuring that the concepts that are being developed in this Pre-Phase A would be able to meet the likely requirements and expectations of the stakeholders. Concepts would be developed to the lowest level necessary to ensure that the concepts are feasible and to a level that will reduce the risk low enough to satisfy the project. Academically, this process could proceed down to the circuit board level for every system.

<table>
<thead>
<tr>
<th>Phase</th>
<th>Purpose</th>
<th>Typical Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-Phase A Concept Studies</td>
<td>To produce a broad spectrum of ideas and alternatives for missions from which new programs/projects can be selected. Determine feasibility of desired system, develop mission concepts, draft system-level requirements, identify potential technology needs.</td>
<td>Feasible system concepts in the form of simulations, analysis, study reports, models, and mockups</td>
</tr>
<tr>
<td>Phase A Concept and Technology Development</td>
<td>To determine the feasibility and desirability of a suggested new major system and establish an initial baseline compatibility with NASA’s strategic plans. Develop final mission concept, system-level requirements, and needed system structure technology developments.</td>
<td>System concept definition in the form of simulations, analysis, engineering models, and mockups and trade study definition</td>
</tr>
<tr>
<td>Phase B Preliminary Design and Technology Completion</td>
<td>To define the project in enough detail to establish an initial baseline capable of meeting mission needs. Develop system structure end product (and enabling product) requirements and generate a preliminary design for each system structure end product.</td>
<td>End products in the form of mockups, trade study results, specification and interface documents, and prototypes</td>
</tr>
<tr>
<td>Phase C Final Design and Fabrication</td>
<td>To complete the detailed design of the system (and its associated subsystems, including its operations systems), fabricate hardware, and code software. Generate final designs for each system structure end product.</td>
<td>End product detailed designs, end product component fabrication, and software development</td>
</tr>
<tr>
<td>Phase D System Assembly, Integration and Test, Launch</td>
<td>To assemble and integrate the products to create the system, meanwhile developing confidence that it will be able to meet the system requirements. Launch and prepare for operations. Perform system end product implementation, assembly, integration and test, and transition to use.</td>
<td>Operations-ready system end product with supporting related enabling products</td>
</tr>
<tr>
<td>Phase E Operations and Sustainment</td>
<td>To conduct the mission and meet the initially identified need and maintain support for that need. Implement the mission operations plan.</td>
<td>Desired system</td>
</tr>
<tr>
<td>Phase F Closeout</td>
<td>To implement the systems decommissioning/disposal plan developed in Phase E and perform analyses of the returned data and any returned samples.</td>
<td>Product closeout</td>
</tr>
</tbody>
</table>
2.0 Fundamentals of Systems Engineering

However, that would involve a great deal of time and money. There may be a higher level or tier of product than circuit board level that would enable designers to accurately determine the feasibility of accomplishing the project (purpose of Pre-Phase A).

During Phase A, the recursive use of the SE engine is continued, this time taking the concepts and draft key requirements that were developed and validated during Pre-Phase A and fleshing them out to become the set of baseline system requirements and Concept of Operations (ConOps). During this phase, key areas of high risk might be simulated or prototyped to ensure that the concepts and requirements being developed are good ones and to identify verification and validation tools and techniques that will be needed in later phases.

During Phase B, the SE engine is applied recursively to further mature requirements for all products in the developing product tree, develop ConOps preliminary designs, and perform feasibility analysis of the verification and validation concepts to ensure the designs will likely be able to meet their requirements.

Phase C again uses the left side of the SE engine to finalize all requirement updates, finalize ConOps, develop the final designs to the lowest level of the product tree, and begin fabrication. Phase D uses the right side of the SE engine to recursively perform the final implementation, integration, verification, and validation of the end product, and at the final pass, transition the end product to the user. The technical management processes of the SE engine are used in Phases E and F to monitor performance; control configuration; and make decisions associated with the operations, sustaining engineering, and closeout of the system. Any new capabilities or upgrades of the existing system would reenter the SE engine as new developments.

2.3.1 Detailed Example

Since it is already well known, the NASA Space Transportation System (STS) will be used as an example to look at how the SE engine would be used in Phase A. This example will be simplified to illustrate the application of the SE processes in the engine, but will in no way be as detailed as necessary to actually build the highly complex vehicle. The SE engine is used recursively to drive out more and more detail with each pass. The icon shown in Figure 2.3-1 will be used to keep track of the applicable place in the SE engine. The numbers in the icon correspond to the numbered processes within the SE engine as shown in Figure 2.1-1. The various layers of the product hierarchy will be called “tiers.” Tiers are also called “layers,” or “levels.” But basically, the higher the number of the tier or level, the lower in the product hierarchy the product is going and the more detailed the product is becoming (e.g., going from boxes, to circuit boards, to components).

2.3.2 Example Premise

NASA decides that there is a need for a transportation system that will act like a “truck” to carry large pieces of equipment and crew into Low Earth Orbit (LEO). Referring back to the project life cycle, the project first enters the Pre-Phase A. During this phase, several concept studies are performed, and it is determined that it is feasible to develop such a “space truck.” This is determined through combinations of simulations, mockups, analyses, or other like means. For simplicity, assume feasibility will be proven through concept models. The processes and framework of the SE engine will be used to design and implement these models. The project would then enter the Phase A activities to take the Pre-Phase A concepts and refine them and define the system requirements for the end product. The detailed example will begin in Phase A and show how the SE engine is used. As described in the overview, a similar process is used for the other project phases.

2.3.2.1 Example Phase A System Design Passes

First Pass

Taking the preliminary concepts and drafting key system requirements developed during the Pre-Phase A activities, the SE engine is entered at the first process and used to determine who the product (i.e., the STS) stakeholders are and what they want. During Pre-Phase A these needs and expectations were pretty general ideas, probably just saying the Agency needs a “space truck” that will carry X tons of payload into LEO, accommodate a payload of so-and-so size,
2.3 Example of Using the SE Engine

Next, using the requirements and the ConOps previously developed, logical decomposition models/diagrams are built up to help bring the requirements into perspective and to show their relationship. Finally, these diagrams, requirements, and ConOps documents are used to develop one or more feasible design solutions. Note that at this point, since this is only the first pass through the SE engine, these design solutions are not detailed enough to actually build anything. Consequently, the design solutions might be summarized as, “To accomplish this transportation system, the best option in our trade studies is a three-part system: a reusable orbiter for the crew and cargo, a large external tank to hold the propellants, and two solid rocket boosters to give extra power for liftoff that can be recovered, refurbished, and reused.” (Of course, the actual design solution would be much more descriptive and detailed.) So, for this first pass, the first tier of the product hierarchy might look like Figure 2.3-2. There would also be other enabling products that might appear in the product tree, but for simplicity only, the main products are shown in this example.

Now, obviously design solution is not yet at a detailed enough level to actually build the prototypes or models of any of these products. The requirements, ConOps, functional diagrams, and design solutions are still at a pretty high, general level. Note that the SE processes on the right side (i.e., the product realization processes) of the SE engine have yet to be addressed. The design must first be at a level that something can actually be built, coded, or reused before that side of the SE engine can be used. So, a second pass of the left side of the SE engine will be started.

Second Pass

The SE engine is completely recursive. That is, each of the three elements shown in the tier 1 diagram can now be considered a product of its own and the SE engine is therefore applied to each of the three elements separately. For example, the external tank is considered an end product and the SE engine resets back to the first processes. So now, just focusing on the external tank, who are the stakeholders and what they expect of the external tank is determined. Of course, one of the main stakeholders will be the owners of the tier 1 requirements and the STS as an end product, but there will also be other new stakeholders. A new ConOps for how the external tank would operate is generated. The tier 1 requirements that are applicable (allocated) to the external tank would be “flowed down” and validated. Usually, some of these will be too general to implement into a design, so the requirements will have to be detailed out. To these derived requirements, there will also be added new requirements that are generated from the stakeholder expectations, and other applicable standards for workmanship, safety, quality, etc.

Next, the external tank requirements and the external tank ConOps are established, and functional diagrams are developed as was done in the first pass with the STS product. Finally, these diagrams, requirements, and ConOps documents are used to develop some feasible design solutions for the external tank. At this pass, there

---

Tier 0
Space Transportation System

Tier 1
- External Tank
- Orbiter
- Solid Rocket Booster

Figure 2.3-2 Product hierarchy, tier 1: first pass through the SE engine

---

carry a crew of seven, etc. During this Phase A pass, these general concepts are detailed out and agreed to. The ConOps (sometimes referred to as operational concept) generated in Pre-Phase A is also detailed out and agreed to to ensure all stakeholders are in agreement as to what is really expected of the product—in this case the transportation system. The detailed expectations are then converted into good requirement statements. (For more information on what constitutes a good requirement, see Appendix C.) Subsequent passes and subsequent phases will refine these requirements into specifications that can actually be built. Also note that all of the technical management processes (SE engine processes numbered 10 through 17) are also used during this and all subsequent passes and activities. These ensure that all the proper planning, control, assessment, and decisions are used and maintained. Although for simplification they will not be mentioned in the rest of this example, they will always be in effect.
will also not be enough detail to actually build or prototype the external tank. The design solution might be summarized as, “To build this external tank, since our trade studies showed the best option was to use cryogenic propellants, a tank for the liquid hydrogen will be needed as will another tank for the liquid oxygen, instrumentation, and an outer structure of aluminum coated with foam.” Thus, the tier 2 product tree for the external tank might look like Figure 2.3-3.

In a similar manner, the orbiter would also take another pass through the SE engine starting with identifying the stakeholders and their expectations, and generating a ConOps for the orbiter element. The tier 1 requirements that are applicable (allocated) to the orbiter would be “flowed down” and validated; new requirements derived from them and any additional requirements (including interfaces with the other elements) would be added.

Next, the orbiter requirements and the ConOps are taken, functional diagrams are developed, and one or more feasible design solutions for the orbiter are generated. As with the external tank, at this pass, there will not be enough detail to actually build or do a complex model of the orbiter. The orbiter design solution might be summarized as, “To build this orbiter will require a winged vehicle with a thermal protection system; an avionics system; a guidance, navigation, and control system; a propulsion system; an environmental control system; etc.” So the tier 2 product tree for the orbiter element might look like Figure 2.3-4.

Likewise, the solid rocket booster would also be considered an end product, and a pass through the SE engine would generate a tier 2 design concept, just as was done with the external tank and the orbiter.

**Third Pass**

Each of the tier 2 elements is also considered an end product, and each undergoes another pass through the SE engine, defining stakeholders, generating ConOps, flowing down allocated requirements, generating new and derived requirements, and developing functional diagrams and design solution concepts. As an example of just the avionics system element, the tier 3 product hierarchy tree might look like Figure 2.3-5.

**Passes 4 Through n**

For this Phase A set of passes, this recursive process is continued for each product (model) on each tier down to the lowest level in the product tree. Note that in some projects it may not be feasible, given an estimated project cost and schedule, to perform this recursive process completely down to the smallest component during Phase A.
In these cases, engineering judgment must be used to determine what level of the product tier is feasible. Note that the lowest feasible level may occur at different tiers depending on the product-line complexity. For example, for one product line it may occur at tier 2; whereas, for a more complex product, it could occur at tier 8. This also means that it will take different amounts of time to reach the bottom. Thus, for any given program or project, products will be at various stages of development. For this Phase A example, Figure 2.3-6 depicts the STS product hierarchy after completely passing through the system design processes side of the SE engine. At the end of this set of passes, system requirements, ConOps, and high-level conceptual functional and physical architectures for each product in the tree would exist. Note that these would not yet be the detailed or even preliminary designs for the end prod-

Figure 2.3-5 Product hierarchy, tier 3: avionics system

Figure 2.3-6 Product hierarchy: complete pass through system design processes side of the SE engine

Note: The unshaded boxes represent bottom-level phase products.
ucts. These will come later in the life cycle. At this point, enough conceptual design work has been done to ensure that at least the high-risk requirements are achievable as will be shown in the following passes.

2.3.2.2 Example Product Realization Passes

So now that the requirements and conceptual designs for the principal Phase A products have been developed, they need to be checked to ensure they are achievable. Note that there are two types of products. The first product is the “end product”—the one that will actually be delivered to the final user. The second type of product will be called a “phase product.” A phase product is generated within a particular life-cycle phase that helps move the project toward delivering a final product. For example, while in Pre-Phase A, a foam-core mockup might be built to help visualize some of the concepts. Those mockups would not be the final “end product,” but would be the “phase product.” For this Phase A example, assume some computer models will be created and simulations performed of these key concepts to show that they are achievable. These will be the phase product for our example.

Now the focus shifts to the right side (i.e., product realization processes) of the SE engine, which will be applied recursively, starting at the bottom of the product hierarchy and moving upwards.

First Pass

Each of the phase products (i.e., our computer models) for the bottom-level product tier (ones that are unshaded in Figure 2.3-6) is taken individually and realized—that is, it is either bought, built, coded, or reused. For our example, assume the external tank product model Aa is a standard Commercial-Off-the-Shelf (COTS) product that is bought. Aba is a model that can be reused from another project, and product Abb is a model that will have to be developed with an in-house design that is to be built. Note that these models are parts of a larger model product that will be assembled or integrated on a subsequent runthrough of the SE engine. That is, to realize the model for product Ab of the external tank, models for products Aba and Abb must be first implemented and then later integrated together. This pass of the SE engine will be the realizing part. Likewise, each of the unshaded bottom-level model products is realized in this first pass. The models will help us understand and plan the method to implement the final end product and will ensure the feasibility of the implemented method.

Next, each of the realized models (phase products) are used to verify that the end product would likely meet the requirements as defined in the Technical Requirements Definition Process during the system design pass for this product. This shows the product would likely meet the “shall” statements that were allocated, derived, or generated for it by method of test, analysis, inspection, or demonstration—that it was “built right.” Verification is performed for each of the unshaded bottom-level model products. Note that during this Phase A pass, this process is not the formal verification of the final end product. However, using analysis, simulation, models, or other means shows that the requirements are good (verifiable) and that the concepts will most likely satisfy them. This also allows draft verification procedures of key areas to be developed. What can be formally verified, however, is that the phase product (the model) meets the requirements for the model.

After verification and validation of the phase product (models) has been verified and used for planning the end product verification, the models are then used for validation. That is, additional test, analysis, inspection, or demonstrations are conducted to ensure that the proposed conceptual designs will likely meet the expectations of the stakeholders for this phase product and for the end product. This will track back to the ConOps that was mutually developed with the stakeholders during the Stakeholder Expectations Definition Process of the system design pass for this product. This will help ensure that the project has “built the right” product at this level.

After the phase product (models) is realized, it is time to prepare the model for transition to the next
level up. Depending on complexity, where the model will be transitioned, security requirements, etc., transition may involve crating and shipment, transmitting over a network, or hand carrying over to the next lab. Whatever is appropriate, each model for the bottom-level product is prepared and handed to the next level up for further integration.

**Second Pass**

Now that all the models (phase products) for the bottom-level end products are realized, verified, validated, and transitioned, it is time to start integrating them into the next higher level product. For example, for the external tank, realized tier 4 models for product Aba and Abb are integrated to form the model for the tier 3 product Ab. Note that the Product Implementation Process only occurs at the bottommost product. All subsequent passes of the SE engine will employ the Product Integration Process since already realized products will be integrated to form the new higher level products. Integrating the lower tier phase products will result in the next-higher-tier phase product. This integration process can also be used for planning the integration of the final end products.

After the new integrated phase product (model) has been formed (tier 3 product Ab for example), it must now be proven that it meets its requirements. These will be the allocated, derived, or generated requirements developed during the Technical Requirements Definition Process during the system design pass for the model for this integrated product. This ensures that the integrated product was built (assembled) right. Note that just verifying the component parts (i.e., the individual models) that were used in the integration is not sufficient to assume that the integrated product will work right. There are many sources of problems that could occur— incomplete requirements at the interfaces, wrong assumptions during design, etc. The only sure way of knowing if an integrated product is good is to perform verification and validation at each stage. The knowledge gained from verifying this integrated phase product can also be used for planning the verification of the final end products.

Likewise, after the integrated phase product is verified, it needs to be validated to show that it meets the expectations as documented in the ConOps for the model of the product at this level. Even though the component parts making up the integrated product will have been validated at this point, the only way to know that the project has built the “right” integrated product is to perform validation on the integrated product itself. Again, this information will help in the planning for the validation of the end products.

The model for the integrated phase product at this level (tier 3 product Ab for example) is now ready to be transitioned to the next higher level (tier 2 for the example). As with the products in the first pass, the integrated phase product is prepared according to its needs/requirements and shipped or handed over. In the example, the model for the external tank tier 3 integrated product Ab is transitioned to the owners of the model for the tier 2 product A. This effort with the phase products will be useful in planning for the transition of the end products.

**Passes 3 Through n**

In a similar manner as the second pass, the tier 3 models for the products are integrated together, realized, verified, validated, and transitioned to the next higher tier. For the example, the realized model for external tank tier 3 integrated phase product Ab is integrated with the model for tier 3 realized phase product Aa to form the tier 2 phase product A. Note that tier 3 product Aa is a bottom-tier product that has yet to go through the integration process. It may also have been realized some time ago and has been waiting for the Ab product line to become realized. Part of its transition might have been to place it in secure storage until the Ab product line became available. Or it could be that Aa was the long-lead item and product Ab had been completed some time ago and was waiting for the tranche to purchase to arrive before they could be integrated together.
The length of the branch of the product tree does not necessarily translate to a corresponding length of time. This is why good planning in the first part of a project is so critical.

Final Pass

At some point, all the models for the tier 1 phase products will each have been used to ensure the system requirements and concepts developed during this Phase A cycle can be implemented, integrated, verified, validated, and transitioned. The elements are now defined as the external tank, the orbiter, and the solid rocket boosters. One final pass through the SE engine will show that they will likely be successfully implemented, integrated, verified, and validated. The final of these products—in the form of the baseline system requirements, ConOps, conceptual functional and physical designs—are made to provide inputs into the next life-cycle phase (B) where they will be further matured. In later phases, the products will actually be built into physical form. At this stage of the project, the key characteristics of each product are passed downstream in key SE documentation, as noted.

2.3.2.3 Example Use of the SE Engine in Phases B Through D

Phase B begins the preliminary design of the final end product. The recursive passes through the SE engine are repeated in a similar manner to that discussed in the detailed Phase A example. At this phase, the phase product might be a prototype of the product(s). Prototypes could be developed and then put through the planned verification and validation processes to ensure the design will likely meet all the requirements and expectations prior to the build of the final flight units. Any mistakes found on prototypes are much easier and less costly to correct than if not found until the flight units are built and undergoing the certification process.

Whereas the previous phases dealt with the final product in the form of analysis, concepts, or prototypes, Phases C and D work with the final end product itself. During Phase C, we recursively use the left side of the SE engine to develop the final design. In Phase D, we recursively use the right side of the SE engine to realize the final product and conduct the formal verification and validation of the final product. As we come out of the last pass of the SE engine in Phase D, we have the final fully realized end product, the STS, ready to be delivered for launch.

2.3.2.4 Example Use of the SE Engine in Phases E and F

Even in Phase E (Operations and Sustainment) and Phase F (Closeout) of the life cycle, the technical management processes in the SE engine are still being used. During the operations phase of a project, a number of activities are still going on. In addition to the day-to-day use of the product, there is a need to monitor or manage various aspects of the system. This is where the key Technical Performance Measures (TPMs) that were defined in the early stages of development continue to play a part. (TPMs are described in Subsection 6.7.2.) These are great measures to monitor to ensure the product continues to perform as designed and expected. Configurations are still under control, still executing the Configuration Management Process. Decisions are still being made using the Decision Analysis Process. Indeed, all of the technical management processes still apply. For this discussion, the term “systems management” will be used for this aspect of operations. In addition to systems management and systems operation, there may also be a need for periodic refurbishment, repairing broken parts, cleaning, sparring, logistics, or other activities. Although other terms are used, for the purposes of this discussion the term “sustaining engineering” will be used for these activities. Again, all of the technical management processes still apply to these
2.3 Example of Using the SE Engine

Figure 2.3-7 represents these three activities occurring simultaneously and continuously throughout the operational lifetime of the final product. Some portions of the SE processes need to continue even after the system becomes nonoperational to handle retirement, decommissioning, and disposal. This is consistent with the basic SE principle of handling the full system life cycle from “cradle to grave.”

However, if at any point in this phase a new product, a change that affects the design or certification of a product, or an upgrade to an existing product is needed, the development processes of the SE engine are reentered at the top. That is, the first thing that is done for an upgrade is to determine who the stakeholders are and what they expect. The entire SE engine is used just as for a newly developed product. This might be pictorially portrayed as in Figure 2.3-8. Note that in the figure although the SE engine is shown only once, it is used recursively down through the product hierarchy for upgraded products, just as described in our detailed example for the initial product.

2.4 Distinctions Between Product Verification and Product Validation

From a process perspective, the Product Verification and Product Validation Processes may be similar in nature, but the objectives are fundamentally different. Verification of a product shows proof of compliance with requirements—that the product can meet each “shall” statement as proven through performance of a test, analysis, inspection, or demonstration. Validation of a product shows that the product accomplishes the intended purpose in the intended environment—that it meets the expectations of the customer and other stakeholders as shown through performance of a test, analysis, inspection, or demonstration.

Verification testing relates back to the approved requirements set and can be performed at different stages in the product life cycle. The approved specifications, drawings, parts lists, and other configuration documentation establish the configuration baseline of that product, which may have to be modified at a later time. Without a verified baseline and appropriate configuration controls, later modifications could be costly or cause major performance problems.

Validation relates back to the ConOps document. Validation testing is conducted under realistic conditions (or simulated conditions) on end products for the purpose of determining the effectiveness and suitability of the product for use in mission operations by typical users.

The selection of the verification or validation method is based on engineering judgment as to which is the most effective way to reliably show the product’s conformance to requirements or that it will operate as intended and described in the ConOps.
2.5 Cost Aspect of Systems Engineering

The objective of systems engineering is to see that the system is designed, built, and operated so that it accomplishes its purpose safely in the most cost-effective way possible considering performance, cost, schedule, and risk.

A cost-effective and safe system must provide a particular kind of balance between effectiveness and cost: the system must provide the most effectiveness for the resources expended, or equivalently, it must be the least expensive for the effectiveness it provides. This condition is a weak one because there are usually many designs that meet the condition. Think of each possible design as a point in the tradeoff space between effectiveness and cost. A graph plotting the maximum achievable effectiveness of designs available with current technology as a function of cost would, in general, yield a curved line such as the one shown in Figure 2.5-1. (In the figure, all the dimensions of effectiveness are represented by the ordinate (y axis) and all the dimensions of cost by the abscissa (x axis).) In other words, the curved line represents the envelope of the currently available technology in terms of cost-effectiveness.

Points above the line cannot be achieved with currently available technology; that is, they do not represent feasible designs. (Some of those points may be feasible in the future when further technological advances have been made.) Points inside the envelope are feasible, but are said to be dominated by designs whose combined cost and effectiveness lie on the envelope line. Designs represented by points on the envelope line are called cost-effective (or efficient or nondominated) solutions.

Design trade studies, an important part of the systems engineering process, often attempt to find designs that provide a better combination of the various dimensions of cost and effectiveness. When the starting point for a design trade study is inside the envelope, there are alternatives that either reduce costs with change to the overall effectiveness or alternatives that improve effectiveness without a cost increase (i.e., moving closer to the envelope curve). Then, the systems engineer’s decision is easy. Other than in the sizing of subsystems, such “win-win” design trades are uncommon, but by no means rare. When the alternatives in a design trade study require trading cost for effectiveness, or even one dimension of effectiveness for another at the same cost (i.e., moving parallel to the envelope curve), the decisions become harder.

**Figure 2.5-1 The enveloping surface of nondominated designs**
The process of finding the most cost-effective design is further complicated by uncertainty, which is shown in Figure 2.5-2. Exactly what outcomes will be realized by a particular system design cannot be known in advance with certainty, so the projected cost and effectiveness of a design are better described by a probability distribution than by a point. This distribution can be thought of as a cloud that is thickest at the most likely value and thinnest farthest away from the most likely point, as is shown for design concept A in the figure. Distributions resulting from designs that have little uncertainty are dense and highly compact, as is shown for concept B. Distributions associated with risky designs may have significant probabilities of producing highly undesirable outcomes, as is suggested by the presence of an additional low-effectiveness/high-cost cloud for concept C. (Of course, the envelope of such clouds cannot be a sharp line such as is shown in the figure, but must itself be rather fuzzy. The line can now be thought of as representing the envelope at some fixed confidence level, that is, a specific, numerical probability of achieving that effectiveness.)

Both effectiveness and cost may require several descriptors. Even the Echo balloons (circa 1960), in addition to their primary mission as communications satellites, obtained scientific data on the electromagnetic environment and atmospheric drag. Furthermore, Echo was the first satellite visible to the naked eye, an unquantifiable—but not unrecognized at the beginning of the space race—aspect of its effectiveness. Sputnik (circa 1957), for example, drew much of its effectiveness from the fact that it was a “first.” Costs, the expenditure of limited resources, may be measured in the several dimensions of funding, personnel, use of facilities, and so on. Schedule may appear as an attribute of effectiveness or cost, or as a constraint. A mission to Mars that misses its launch window has to wait about two years for another opportunity—a clear schedule constraint.

In some contexts, it is appropriate to seek the most effectiveness possible within a fixed budget and with a fixed risk; in other contexts, it is more appropriate to seek the least cost possible with specified effectiveness and risk. In these cases, there is the question of what level of effectiveness to specify or what level of costs to fix. In practice, these may be mandated in the form of performance or cost requirements. It then becomes appropriate to ask whether a slight relaxation of requirements could produce a significantly cheaper system or whether a few more resources could produce a significantly more effective system.

The technical team must choose among designs that differ in terms of numerous attributes. A variety of methods have been developed that can be used to help uncover preferences between attributes and to quantify subjective assessments of relative value. When this can be done, trades between attributes can be assessed quantitatively. Often, however, the attributes seem to be truly incommensurate: decisions need to be made in spite of this multiplicity.
3.0 NASA Program/Project Life Cycle

One of the fundamental concepts used within NASA for the management of major systems is the program/project life cycle, which consists of a categorization of everything that should be done to accomplish a program or project into distinct phases, separated by Key Decision Points (KDPs). 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). Phase boundaries are defined so that they provide more or less natural points for Go or No-Go decisions. Decisions to proceed may be qualified by liens that must be removed within an agreed to time period. A program or project that fails to pass a KDP may be allowed to “go back to the drawing board” to try again later—or it may be terminated.

All systems start with the recognition of a need or the discovery of an opportunity and proceed through various stages of development to a final disposition. While the most dramatic impacts of the analysis and optimization activities associated with systems engineering are obtained in the early stages, decisions that affect millions of dollars of value or cost continue to be amenable to the systems approach even as the end of the system lifetime approaches.

Decomposing the program/project life cycle into phases organizes the entire process into more manageable pieces. The program/project life cycle should provide managers with incremental visibility into the progress being made at points in time that fit with the management and budgetary environments.

NPR 7120.5, NASA Space Flight Program and Project Management Requirements defines the major NASA life-cycle phases as Formulation and Implementation. For Flight Systems and Ground Support (FS&GS) projects, the NASA life-cycle phases of Formulation and Implementation divide into the following seven incremental pieces. The phases of the project life cycle are:

- Pre-Phase A: Concept Studies (i.e., identify feasible alternatives)
- Phase A: Concept and Technology Development (i.e., define the project and identify and initiate necessary technology)
- Phase B: Preliminary Design and Technology Completion (i.e., establish a preliminary design and develop necessary technology)
- Phase C: Final Design and Fabrication (i.e., complete the system design and build/code the components)
- Phase D: System Assembly, Integration and Test, Launch (i.e., integrate components, and verify the system, prepare for operations, and launch)
- Phase E: Operations and Sustainment (i.e., operate and maintain the system)
- Phase F: Closeout (i.e., disposal of systems and analysis of data)

Figure 3.0-1 (NASA program life cycle) and Figure 3.0-2 (NASA project life cycle) identify the KDPs and reviews that characterize the phases. Sections 3.1 and 3.2 contain narrative descriptions of the purposes, major activities, products, and KDPs of the NASA program life-cycle phases. Sections 3.3 to 3.9 contain narrative descriptions of the purposes, major activities, products, and KDPs of the NASA project life-cycle phases. Section 3.10 describes the NASA budget cycle within which program/project managers and systems engineers must operate.

3.1 Program Formulation

The program Formulation phase establishes a cost-effective program that is demonstrably capable of meeting Agency and mission directorate goals and objectives. The program Formulation Authorization Document (FAD) authorizes a Program Manager (PM) to initiate the planning of a new program and to perform the analyses required to formulate a sound program plan. Major reviews leading to approval at KDP I are the P/SRR, P/SDR, PAR, and governing Program Management Council (PMC) review. (See full list of reviews in the program and project life cycle figures on the next page.) A summary of the required gate products for the pro-
3.0 NASA Program/Project Life Cycle

NASA Life-Cycle Phases

Key Decision Points

- KDP 0
- KDP I
- KDP II
- KDP III
- KDP IV
- KDP n

Formulation Approval Implementation

Uncoupled & Loosely Coupled Programs

- PSR
- P/SDR

Single-Project & Tightly Coupled Programs

- PDR
- CDR
- SIR
- TRR
- ORR
- FRR
- PLAR
- CERR
- PSR

Figure 3.0-1 NASA program life cycle

- CDR: Critical Design Review
- CERR: Critical Events Readiness Review
- DR: Decommissioning Review
- FRR: Flight Readiness Review
- KDP: Key Decision Point
- MCR: Mission Concept Review
- MDR: Mission Definition Review
- ORR: Operational Readiness Review
- PDR: Preliminary Design Review
- PFAR: Post-Flight Assessment Review
- PIR: Program Implementation Review
- PLAR: Post-Launch Assessment Review
- PRR: Production Readiness Review
- P/SDR: Program/System Definition Review
- P/SRR: Program/System Requirements Review
- ORR: Operational Readiness Review
- SIR: System Integration Review
- SRR: System Requirements Review
- SIR: System Integration Review
- TRR: Test Readiness Review

Figure 3.0-2 NASA project life cycle

- Pre-Phase A: Concept Studies
- Phase A: Preliminary Design & Technology Development
- Phase B: Final Design & Fabrication
- Phase C: System Assembly, Integration & Test, Launch
- Phase D: Operations & Sustainment
- Phase E: 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
- TRR
- SAR
- ORR
- FRR
- PLAR
- CERR
- PFAR
- DR

Robotic Mission Reviews

- MCR
- SRR
- MDR
- PDR
- CDR/PRR
- TRR
- SAR
- ORR
- FRR
- PLAR
- CERR
- PFAR
- DR

Supporting Reviews

- Peer Reviews, Subsystem Reviews, and System Reviews

PSRs, PIRs, and KDPs are conducted ~ every 2 years
3.1 Program Formulation

Program Formulation

Purpose
To establish a cost-effective program that is demonstrably capable of meeting Agency and mission directorate goals and objectives

Typical Activities and Their Products
- Develop program requirements and allocate them to initial projects
- Define and approve program acquisition strategies
- Develop interfaces to other programs
- Start development of technologies that cut across multiple projects within the program
- Derive initial cost estimates and approve a program budget
- Perform required program Formulation technical activities defined in NPR 7120.5
- Satisfy program Formulation reviews’ entrance/success criteria detailed in NPR 7123.1

Reviews
- P/SRR
- P/SDR

For uncoupled and loosely coupled programs, the Implementation phase only requires PSRs and PIRs to assess the program’s performance and make a recommendation on its authorization at KDPs approximately every two years. Single-project and tightly coupled programs are more complex. For single-project programs, the Implementation phase program reviews shown in Figure 3.0-1 are synonymous (not duplicative) with the project reviews in the project life cycle (see Figure 3.0-2) through Phase D. Once in operations, these programs usually have biennial KDPs preceded by attendant PSRs/PIRs. Tightly coupled programs during implementation have program reviews tied to the project reviews to ensure the proper integration of projects into the larger system. Once in operations, tightly coupled programs also have biennial PSRs/PIRs/KDPs to assess the program’s performance and authorize its continuation.

Program Implementation

Purpose
To execute the program and constituent projects and ensure the program continues to contribute to Agency goals and objectives within funding constraints

Typical Activities and Their Products
- Initiate projects through direct assignment or competitive process (e.g., Request for Proposal (RFP), Announcement of Opportunity (AO))
- Monitor project’s formulation, approval, implementation, integration, operation, and ultimate decommissioning
- Adjust program as resources and requirements change
- Perform required program Implementation technical activities from NPR 7120.5
- Satisfy program Implementation reviews’ entrance/success criteria from NPR 7123.1

Reviews
- PSR/PIR (uncoupled and loosely coupled programs only)
- Reviews synonymous (not duplicative) with the project reviews in the project life cycle (see Figure 3.0-2) through Phase D (single-project and tightly coupled programs only)

gram Formulation phase can be found in NPR 7120.5. Formulation for all program types is the same, involving one or more program reviews followed by KDP I where a decision is made approving a program to begin implementation. Typically, there is no incentive to move a program into implementation until its first project is ready for implementation.

3.2 Program Implementation

During the program Implementation phase, the PM works with the Mission Directorate Associate Administrator (MDAA) and the constituent project managers to execute the program plan cost effectively. Program reviews ensure that the program continues to contribute to Agency and mission directorate goals and objectives within funding constraints. A summary of the required gate products for the program Implementation phase can be found in NPR 7120.5. The program life cycle has two different implementation paths, depending on program type. Each implementation path has different types of major reviews.
The purpose of this phase, which is usually performed more or less continually by concept study groups, is to devise various feasible concepts from which new projects (programs) can be selected. Typically, this activity consists of loosely structured examinations of new ideas, usually without central control and mostly oriented toward small studies. Its major product is a list of suggested projects, based on the identification of needs and the discovery of opportunities that are potentially consistent with NASA’s mission, capabilities, priorities, and resources.

Advanced studies may extend for several years and may be a sequence of papers that are only loosely connected. These studies typically focus on establishing mission goals and formulating top-level system requirements and ConOps. Conceptual designs are often offered to demonstrate feasibility and support programmatic estimates. The emphasis is on establishing feasibility and desirability rather than optimality. Analyses and designs are accordingly limited in both depth and number of options.

### Pre-Phase A: Concept Studies

**Purpose**
To produce a broad spectrum of ideas and alternatives for missions from which new programs/projects can be selected

**Typical Activities and Products**
(Note: AO projects will have defined the deliverable products.)
- Identify missions and architecture consistent with charter
- Identify and involve users and other stakeholders
- Identify and perform tradeoffs and analyses
- Identify requirements, which include:
  - Mission,
  - Science, and
  - Top-level system.
- Define measures of effectiveness and measures of performance
- Identify top-level technical performance measures
- Perform preliminary evaluations of possible missions
- Prepare program/project proposals, which may include:
  - Mission justification and objectives;
  - Possible ConOps;
  - High-level WBSs;
  - Cost, schedule, and risk estimates; and
  - Technology assessment and maturation strategies.
- Prepare preliminary mission concept report
- Perform required Pre-Phase A technical activities from NPR 7120.5
- Satisfy MCR entrance/success criteria from NPR 7123.1

**Reviews**
- MCR
- Informal proposal review

### 3.4 Project Phase A: Concept and Technology Development

During Phase A, activities are performed to fully develop a baseline mission concept and begin or assume responsibility for the development of needed technologies. This work, along with interactions with stakeholders, helps establish a mission concept and the program requirements on the project.

In Phase A, a team—often associated with a program or informal project office—readdresses the mission concept to ensure that the project justification and practicality are sufficient to warrant a place in NASA’s budget. The team’s effort focuses on analyzing mission requirements and establishing a mission architecture. Activities become formal, and the emphasis shifts toward establishing optimality rather than feasibility. The effort addresses more depth and considers many alternatives. Goals and objectives are solidified, and the project develops more definition in the system requirements, top-level system architecture, and ConOps. Conceptual designs are developed and exhibit more engineering detail than in advanced studies. Technical risks are identified in more detail, and technology development needs become focused.

In Phase A, the effort focuses on allocating functions to particular items of hardware, software, personnel, etc. System functional and performance requirements, along with architectures and designs, become firm as system tradeoffs and subsystem tradeoffs iterate back and forth.
3.4 Project Phase A: Concept and Technology Development

**Phase A: Concept and Technology Development**

**Purpose**
To determine the feasibility and desirability of a suggested new major system and establish an initial baseline compatibility with NASA’s strategic plans.

**Typical Activities and Their Products**
- Prepare and initiate a project plan
- Develop top-level requirements and constraints
- Define and document system requirements (hardware and software)
- Allocate preliminary system requirements to next lower level
- Define system software functionality description and requirements
- Define and document internal and external interface requirements
- Identify integrated logistics support requirements
- Develop corresponding evaluation criteria and metrics
- Document the ConOps
- Baseline the mission concept report
- Demonstrate that credible, feasible design(s) exist
- Perform and archive trade studies
- Develop mission architecture
- Initiate environmental evaluation/National Environmental Policy Act process
- Develop initial orbital debris assessment (NASA Safety Standard 1740.14)
- Establish technical resource estimates
- Define life-cycle cost estimates and develop system-level cost-effectiveness model
- Define the WBS
- Develop SOWs
- Acquire systems engineering tools and models
- Baseline the SEMP
- Develop system risk analyses
- Prepare and initiate a risk management plan
- Prepare and initiate a configuration management plan
- Prepare and initiate a data management plan
- Prepare engineering specialty plans (e.g., contamination control plan, electromagnetic interference/electromagnetic compatibility control plan, reliability plan, quality control plan, parts management plan)
- Prepare a safety and mission assurance plan
- Prepare a software development or management plan (see NPR 7150.2)
- Prepare a technology development plan and initiate advanced technology development
- Establish human rating plan
- Define verification and validation approach and document it in verification and validation plans
- Perform required Phase A technical activities from NPR 7120.5
- Satisfy Phase A reviews’ entrance/success criteria from NPR 7123.1

**Reviews**
- SRR
- MDR (robotic mission only)
- SDR (human space flight only)
in the effort to seek out more cost-effective designs. (Trade studies should precede—rather than follow—system design decisions.) Major products to this point include an accepted functional baseline for the system and its major end items. The effort also produces various engineering and management plans to prepare for managing the project’s downstream processes, such as verification and operations, and for implementing engineering specialty programs.

### 3.5 Project Phase B: Preliminary Design and Technology Completion

During Phase B, activities are performed to establish an initial project baseline, which (according to NPR 7120.5 and NPR 7123.1) includes “a formal flow down of the project-level performance requirements to a complete set of system and subsystem design specifications for both flight and ground elements” and “corresponding preliminary designs.” The technical requirements should be sufficiently detailed to establish firm schedule and cost estimates for the project. It also should be noted, especially for AO-driven projects, that Phase B is where the top-level requirements and the requirements flowed down to the next level are finalized and placed under configuration control. While the requirements should be baselined in Phase A, there are just enough changes resulting from the trade studies and analyses in late Phase A and early Phase B that changes are inevitable. However, by mid-Phase B, the top-level requirements should be finalized.

Actually, the Phase B baseline consists of a collection of evolving baselines covering technical and business aspects of the project: system (and subsystem) requirements and specifications, designs, verification and operations plans, and so on in the technical portion of the baseline, and schedules, cost projections, and management plans in the business portion. Establishment of baselines implies the implementation of configuration management procedures. (See Section 6.5.)

In Phase B, the effort shifts to establishing a functionally complete preliminary design solution (i.e., a functional baseline) that meets mission goals and objectives. Trade studies continue. Interfaces among the

---

**Phase B: Preliminary Design and Technology Completion**

**Purpose**
To define the project in enough detail to establish an initial baseline capable of meeting mission needs

**Typical Activities and Their Products**
- Baseline the project plan
- Review and update documents developed and baselined in Phase A
- Develop science/exploration operations plan based on matured ConOps
- Update engineering specialty plans (e.g., contamination control plan, electromagnetic interference/ electromagnetic compatibility control plan, reliability plan, quality control plan, parts management plan)
- Update technology maturation planning
- Report technology development results
- Update risk management plan
- Update cost and schedule data
- Finalize and approve top-level requirements and flowdown to the next level of requirements
- Establish and baseline design-to specifications (hardware and software) and drawings, verification and validation plans, and interface documents at lower levels
- Perform and archive trade studies’ results
- Perform design analyses and report results
- Conduct engineering development tests and report results
- Select a baseline design solution
- Baseline a preliminary design report
- Define internal and external interface design solutions (e.g., interface control documents)
- Define system operations as well as PI/contract proposal management, review, and access and contingency planning
- Develop appropriate level safety data package
- Develop preliminary orbital debris assessment
- Perform required Phase B technical activities from NPR 7120.5
- Satisfy Phase B reviews’ entrance/success criteria from NPR 7123.1

**Reviews**
- PDR
- Safety review
major end items are defined. Engineering test items may be developed and used to derive data for further design work, and project risks are reduced by successful technology developments and demonstrations. Phase B culminates in a series of PDRs, containing the system-level PDR and PDRs for lower level end items as appropriate. The PDRs reflect the successive refinement of requirements into designs. (See the doctrine of successive refinement in Subsection 4.4.1.2 and Figure 4.4-2.) Design issues uncovered in the PDRs should be resolved so that final design can begin with unambiguous design-to specifications. From this point on, almost all changes to the baseline are expected to represent successive refinements, not fundamental changes. Prior to baselining, the system architecture, preliminary design, and ConOps must have been validated by enough technical analysis and design work to establish a credible, feasible design in greater detail than was sufficient for Phase A.

3.6 Project Phase C: Final Design and Fabrication

During Phase C, activities are performed to establish a complete design (allocated baseline), fabricate or produce hardware, and code software in preparation for integration. Trade studies continue. Engineering test units more closely resembling actual hardware are built and tested to establish confidence that the design will function in the expected environments. Engineering specialty analysis results are integrated into the design, and the manufacturing process and controls are defined and validated. All the planning initiated back in Phase A for the testing and operational equipment, processes and analysis, integration of the engineering specialty analysis, and manufacturing processes and controls is implemented. Configuration management continues to track and control design changes as detailed interfaces are defined. At each step in the successive refinement of the final design, corresponding integration and verification activities are planned in greater detail. During this phase, technical parameters, schedules, and budgets are closely tracked to ensure that undesirable trends (such as an unexpected growth in spacecraft mass or increase in its cost) are recognized early enough to take corrective action. These activities focus on preparing for the CDR, PRR (if required), and the SIR.

Phase C contains a series of CDRs containing the system-level CDR and CDRs corresponding to the different levels of the system hierarchy. A CDR for each end item should be held prior to the start of fabrication/production for hardware and prior to the start of coding of deliverable software products. Typically, the sequence of CDRs reflects the integration process that will occur in the next phase—that is, from lower level CDRs to the system-level CDR. Projects, however, should tailor the sequencing of the reviews to meet the needs of the project. If there is a production run of products, a PRR will be performed to ensure the production plans, facilities, and personnel are ready to begin production. Phase C culminates with an SIR. The final product of this phase is a product ready for integration.

3.7 Project Phase D: System Assembly, Integration and Test, Launch

During Phase D, activities are performed to assemble, integrate, test, and launch the system. These activities focus on preparing for the FRR. Activities include assembly, integration, verification, and validation of the system, including testing the flight system to expected environments within margin. Other activities include the initial training of operating personnel and implementation of the logistics and spares planning. For flight projects, the focus of activities then shifts to prelaunch integration and launch. Although all these activities are conducted in this phase of a project, the planning for these activities was initiated in Phase A. The planning for the activities cannot be delayed until Phase D begins because the design of the project is too advanced to incorporate requirements for testing and operations. Phase D concludes with a system that has been shown to be capable of accomplishing the purpose for which it was created.
Phase C: Final Design and Fabrication

Purpose
To complete the detailed design of the system (and its associated subsystems, including its operations systems), fabricate hardware, and code software

Typical Activities and Their Products
- Update documents developed and baselined in Phase B
- Update interface documents
- Update mission operations plan based on matured ConOps
- Update engineering specialty plans (e.g., contamination control plan, electromagnetic interference/electromagnetic compatibility control plan, reliability plan, quality control plan, parts management plan)
- Augment baselined documents to reflect the growing maturity of the system, including the system architecture, WBS, and project plans
- Update and baseline production plans
- Refine integration procedures
- Baseline logistics support plan
- Add remaining lower level design specifications to the system architecture
- Complete manufacturing and assembly plans and procedures
- Establish and baseline build-to specifications (hardware and software) and drawings, verification and validation plans, and interface documents at all levels
- Baseline detailed design report
- Maintain requirements documents
- Maintain verification and validation plans
- Monitor project progress against project plans
- Develop verification and validation procedures
- Develop hardware and software detailed designs
- Develop the system integration plan and the system operation plan
- Develop the end-to-end information system design
- Develop spares planning
- Develop command and telemetry list
- Prepare launch site checkout and operations plans
- Prepare operations and activation plan
- Prepare system decommissioning/disposal plan, including human capital transition, for use in Phase F
- Finalize appropriate level safety data package
- Develop preliminary operations handbook
- Perform and archive trade studies
- Fabricate (or code) the product
- Perform testing at the component or subsystem level
- Identify opportunities for preplanned product improvement
- Baseline orbital debris assessment
- Perform required Phase C technical activities from NPR 7120.5
- Satisfy Phase C reviews’ entrance/success criteria from NPR 7123.1

Reviews
- CDR
- PRR
- SIR
- Safety review
### Phase D: System Assembly, Integration and Test, Launch

**Purpose**  
To assemble and integrate the products and create the system, meanwhile developing confidence that it will be able to meet the system requirements; conduct launch and prepare for operations.

**Typical Activities and Their Products**
- Integrate and verify items according to the integration and verification plans, yielding verified components and (sub-)systems.
- Monitor project progress against project plans.
- Refine verification and validation procedures at all levels.
- Perform system qualification verifications.
- Perform system acceptance verifications and validation(s) (e.g., end-to-end tests encompassing all elements (i.e., space element, ground system, data processing system).
- Perform system environmental testing.
- Assess and approve verification and validation results.
- Resolve verification and validation discrepancies.
- Archive documentation for verifications and validations performed.
- Baseline verification and validation report.
- Baseline “as-built” hardware and software documentation.
- Update logistics support plan.
- Document lessons learned.
- Prepare and baseline operator’s manuals.
- Prepare and baseline maintenance manuals.
- Approve and baseline operations handbook.
- Train initial system operators and maintainers.
- Train on contingency planning.
- Finalize and implement spares planning.
- Confirm telemetry validation and ground data processing.
- Confirm system and support elements are ready for flight.
- Integrate with launch vehicle(s) and launch, perform orbit insertion, etc., to achieve a deployed system.
- Perform initial operational verification(s) and validation(s).
- Perform required Phase D technical activities from NPR 7120.5.
- Satisfy Phase D reviews’ entrance/success criteria from NPR 7123.1.

**Reviews**
- TRR (at all levels).
- SAR (human space flight only).
- ORR.
- FRR.
- System functional and physical configuration audits.
- Safety review.
3.8 Project Phase E: Operations and Sustainment

During Phase E, activities are performed to conduct the prime mission and meet the initially identified need and maintain support for that need. The products of the phase are the results of the mission. This phase encompasses the evolution of the system only insofar as that evolution does not involve major changes to the system architecture. Changes of that scope constitute new “needs,” and the project life cycle starts over. For large flight projects, there may be an extended period of cruise, orbit insertion, on-orbit assembly, and initial shakedown operations. Near the end of the prime mission, the project may apply for a mission extension to continue mission activities or attempt to perform additional mission objectives.

3.9 Project Phase F: Closeout

During Phase F, activities are performed to implement the systems decommissioning/disposal planning and analyze any returned data and samples. The products of the phase are the results of the mission.

Phase F deals with the final closeout of the system when it has completed its mission; the time at which this occurs depends on many factors. For a flight system that returns to Earth with a short mission duration, closeout may require little more than deintegration of the hardware and its return to its owner. On flight projects of long duration, closeout may proceed according to established plans or may begin as a result of unplanned events, such as failures. Refer to NPD 8010.3, Notification of Intent to Decommission or Terminate Operating Space Systems and Terminate Missions for terminating an operating mission. Alternatively, technological advances may make it uneconomical to continue operating the system either in its current configuration or an improved one.

---

**Phase E: Operations and Sustainment**

**Purpose**
To conduct the mission and meet the initially identified need and maintain support for that need

**Typical Activities and Their Products**
- Conduct launch vehicle performance assessment
- Conduct in-orbit spacecraft checkout
- Commission and activate science instruments
- Conduct the intended prime mission(s)
- Collect engineering and science data
- Train replacement operators and maintainers
- Train the flight team for future mission phases (e.g., planetary landed operations)
- Maintain and approve operations and maintenance logs
- Maintain and upgrade the system
- Address problem/failure reports
- Process and analyze mission data
- Apply for mission extensions, if warranted, and conduct mission activities if awarded
- Prepare for deactivation, disassembly, decommissioning as planned (subject to mission extension)
- Complete post-flight evaluation reports
- Complete final mission report
- Perform required Phase E technical activities from NPR 7120.5
- Satisfy Phase E reviews’ entrance/success criteria from NPR 7123.1

**Reviews**
- PLAR
- CERR
- PFAR (human space flight only)
- System upgrade review
- Safety review

---

**Phase F: Closeout**

**Purpose**
To implement the systems decommissioning/disposal plan developed in Phase C and analyze any returned data and samples

**Typical Activities and Their Products**
- Dispose of the system and supporting processes
- Document lessons learned
- Baseline mission final report
- Archive data
- Begin transition of human capital (if applicable)
- Perform required Phase F technical activities from NPR 7120.5
- Satisfy Phase F reviews’ entrance/success criteria from NPR 7123.1

**Reviews**
- DR
To limit space debris, NPR 8715.6, NASA Procedural Requirements for Limiting Orbital Debris provides guidelines for removing Earth-orbiting robotic satellites from their operational orbits at the end of their useful life. For Low Earth Orbiting (LEO) missions, the satellite is usually deorbited. For small satellites, this is accomplished by allowing the orbit to slowly decay until the satellite eventually burns up in the Earth’s atmosphere. Larger, more massive satellites and observatories must be designed to demise or deorbited in a controlled manner so that they can be safely targeted for impact in a remote area of the ocean. The Geostationary (GEO) satellites at 35,790 km above the Earth cannot be practically deorbited, so they are boosted to a higher orbit well beyond the crowded operational GEO orbit.

In addition to uncertainty as to when this part of the phase begins, the activities associated with safe closeout of a system may be long and complex and may affect the system design. Consequently, different options and strategies should be considered during the project’s earlier phases along with the costs and risks associated with the different options.

### 3.10 Funding: The Budget Cycle

NASA operates with annual funding from Congress. This funding results, however, from a continuous rolling process of budget formulation, budget enactment, and finally, budget execution. NASA’s Financial Management Requirements (FMR) Volume 4 provides the concepts, the goals, and an overview of NASA’s budget system of resource alignment referred to as Planning, Programming, Budgeting, and Execution (PPBE) and establishes guidance on the programming and budgeting phases of the PPBE process, which are critical to budget formulation for NASA. Volume 4 includes strategic budget planning and resources guidance, program review, budget development, budget presentation, and justification of estimates to the Office of Management and Budget (OMB) and to Congress. It also provides detailed descriptions of the roles and responsibilities for key players in each step of the process. It consolidates current legal, regulatory, and administrative policies and procedures applicable to NASA. A highly simplified representation of the typical NASA budget cycle is shown in Figure 3.10-1.
NASA typically starts developing its budget each February with economic forecasts and general guidelines as identified in the most recent President's budget. By late August, NASA has completed the planning, programming, and budgeting phases of the PPBE process and prepares for submittal of a preliminary NASA budget to the OMB. A final NASA budget is submitted to the OMB in September for incorporation into the President's budget transmittal to Congress, which generally occurs in January. This proposed budget is then subjected to congressional review and approval, culminating in the passage of bills authorizing NASA to obligate funds in accordance with congressional stipulations and appropriating those funds. The congressional process generally lasts through the summer. In recent years, however, final bills have often been delayed past the start of the fiscal year on October 1. In those years, NASA has operated on continuing resolution by Congress.

With annual funding, there is an implicit funding control gate at the beginning of every fiscal year. While these gates place planning requirements on the project and can make significant replanning necessary, they are not part of an orderly systems engineering process. Rather, they constitute one of the sources of uncertainty that affect project risks, and they are essential to consider in project planning.
4.0 System Design

This chapter describes the activities in the system design processes listed in Figure 2.1-1. The chapter is separated into sections corresponding to steps 1 to 4 listed in Figure 2.1-1. The processes within each step are discussed in terms of inputs, activities, and outputs. Additional guidance is provided using examples that are relevant to NASA projects. The system design processes are four interdependent, highly iterative and recursive processes, resulting in a validated set of requirements and a validated design solution that satisfies a set of stakeholder expectations. The four system design processes are to develop stakeholder expectations, technical requirements, logical decompositions, and design solutions.

Figure 4.0-1 illustrates the recursive relationship among the four system design processes. These processes start with a study team collecting and clarifying the stakeholder expectations, including the mission objectives, constraints, design drivers, operational objectives, and criteria for defining mission success. This set of stakeholder expectations and high-level requirements is used to drive an iterative design loop where a strawman architecture/design, the concept of operations, and derived requirements are developed. These three products must be consistent with each other and will require iterations and design decisions to achieve this consistency. Once consistency is achieved, analyses allow the project team to validate the design against the stakeholder expectations. A simplified validation asks the questions: Does the system work? Is the system safe and reliable? Is the system achievable within budget and schedule constraints? If the answer to any of these questions is no,
then changes to the design or stakeholder expectations will be required, and the process started again. This process continues until the system—architecture, ConOps, and requirements—meets the stakeholder expectations.

The depth of the design effort must be sufficient to allow analytical verification of the design to the requirements. The design must be feasible and credible when judged by a knowledgeable independent review team and must have sufficient depth to support cost modeling.

Once the system meets the stakeholder expectations, the study team baselines the products and prepares for the next phase. Often, intermediate levels of decomposition are validated as part of the process. In the next level of decomposition, the baselined derived (and allocated) requirements become the set of high-level requirements for the decomposed elements and the process begins again. These system design processes are primarily applied in Pre-Phase A and continue through Phase C.

The system design processes during Pre-Phase A focus on producing a feasible design that will lead to Formulation approval. During Phase A, alternative designs and additional analytical maturity are pursued to optimize the design architecture. Phase B results in a preliminary design that satisfies the approval criteria. During Phase C, detailed, build-to designs are completed.

This has been a simplified description intended to demonstrate the recursive relationship among the system design processes. These processes should be used as guidance and tailored for each study team depending on the size of the project and the hierarchical level of the study team. The next sections describe each of the four system design processes and their associated products for a given NASA mission.

---

### System Design Keys

- Successfully understanding and defining the mission objectives and operational concepts are keys to capturing the stakeholder expectations, which will translate into quality requirements over the life cycle of the project.
- Complete and thorough requirements traceability is a critical factor in successful validation of requirements.
- Clear and unambiguous requirements will help avoid misunderstanding when developing the overall system and when making major or minor changes.
- Document all decisions made during the development of the original design concept in the technical data package. This will make the original design philosophy and negotiation results available to assess future proposed changes and modifications against.
- The design solution verification occurs when an acceptable design solution has been selected and documented in a technical data package. The design solution is verified against the system requirements and constraints. However, the validation of a design solution is a continuing recursive and iterative process during which the design solution is evaluated against stakeholder expectations.
4.1 Stakeholder Expectations Definition

The Stakeholder Expectations Definition Process is the initial process within the SE engine that establishes the foundation from which the system is designed and the product is realized. The main purpose of this process is to identify who the stakeholders are and how they intend to use the product. This is usually accomplished through use-case scenarios, Design Reference Missions (DRMs), and ConOps.

4.1.1 Process Description

Figure 4.1-1 provides a typical flow diagram for the Stakeholder Expectations Definition Process and identifies typical inputs, outputs, and activities to consider in addressing stakeholder expectations definition.

4.1.1.1 Inputs

Typical inputs needed for the Stakeholder Expectations Definition Process would include the following:

- **Upper Level Requirements and Expectations**: These would be the requirements and expectations (e.g., needs, wants, desires, capabilities, constraints, external interfaces) that are being flowed down to a particular system of interest from a higher level (e.g., program, project, etc.).
- **Identified Customers and Stakeholders**: The organization or individual who has requested the product(s) and those who are affected by or are in some way accountable for the product’s outcome.

4.1.1.2 Process Activities

**Identifying Stakeholders**

Advocacy for new programs and projects may originate in many organizations. These include Presidential directives, Congress, NASA Headquarters (HQ), the NASA Centers, NASA advisory committees, the National Academy of Sci-
ences, the National Space Council, and many other groups in the science and space communities. These organizations are commonly referred to as stakeholders. A stakeholder is a group or individual who is affected by or is in some way accountable for the outcome of an undertaking.

Stakeholders can be classified as customers and other interested parties. Customers are those who will receive the goods or services and are the direct beneficiaries of the work. Examples of customers are scientists, project managers, and subsystems engineers.

Other interested parties are those who affect the project by providing broad, overarching constraints within which the customers' needs must be achieved. These parties may be affected by the resulting product, the manner in which the product is used, or have a responsibility for providing life-cycle support services. Examples include Congress, advisory planning teams, program managers, users, operators, maintainers, mission partners, and NASA contractors. It is important that the list of stakeholders be identified early in the process, as well as the primary stakeholders who will have the most significant influence over the project.

**Identifying Stakeholder Expectations**

Stakeholder expectations, the vision of a particular stakeholder individual or group, result when they specify what is desired as an end state or as an item to be produced and put bounds upon the achievement of the goals. These bounds may encompass expenditures (resources), time to deliver, performance objectives, or other less obvious quantities such as organizational needs or geopolitical goals.

Figure 4.1-2 shows the type of information needed when defining stakeholder expectations and depicts how the information evolves into a set of high-level requirements. The yellow paths depict validation paths. Examples of the types of information that would be defined during each step are also provided.

Defining stakeholder expectations begins with the mission authority and strategic objectives that the mission is meant to achieve. *Mission authority* changes depending on the category of the mission. For example, science missions are usually driven by NASA Science Mission Directorate strategic plans; whereas the exploration missions may be driven by a Presidential directive.

An early task in defining stakeholder expectations is understanding the objectives of the mission. Clearly describing and documenting them helps ensure that the project team is working toward a common goal. These objectives form the basis for developing the mission, so they need to be clearly defined and articulated.

Defining the objectives is done by eliciting the needs, wants, desires, capabilities, external interfaces, assumptions, and constraints from the stakeholders. Arriving at an agreed-to set of objectives can be a long and arduous task. The proactive iteration with the stakeholders throughout the systems engineering process is the way

---

**Figure 4.1-2 Product flow for stakeholder expectations**

- **Mission Authority**
  - Agency Strategic Plans
  - Announcements of Opportunity
  - Road Maps
  - Directed Missions

- **Mission Objectives**
  - Science Objectives
  - Exploration Objectives
  - Technology Demonstration Objectives
  - Technology Development Objectives
  - Programmatic Objectives

- **Operational Objectives**
  - Operational Drivers
    - Integration and Test
    - Launch
    - On-Orbit
    - Transfer
    - Surface
    - Science Data Distribution
    - ...

- **Success Criteria**
  - Measurements
    - What measurements?
    - How well?

- **Design Drivers**
  - Mission Drivers
    - Launch Date
    - Mission Duration
    - Orbit
    - ...

---

Please purchase PDF Split-Merge on www.verypdf.com to remove this watermark.
that all parties can come to a true understanding of what should be done and what it takes to do the job. It is important to know who the primary stakeholders are and who has the decision authority to help resolve conflicts.

The project team should also identify the constraints that may apply. A constraint is a condition that must be met. Sometimes a constraint is dictated by external factors such as orbital mechanics or the state of technology; sometimes constraints are the result of the overall budget environment. It is important to document the constraints and assumptions along with the mission objectives.

Operational objectives also need to be included in defining the stakeholder expectations. The operational objectives identify how the mission must be operated to achieve the mission objectives.

The mission and operational success criteria define what the mission must accomplish to be successful. This will be in the form of a measurement concept for science missions and exploration concept for human exploration missions. The success criteria also define how well the concept measurements or exploration activities must be accomplished. The success criteria capture the stakeholder expectations and, along with programmatic requirements and constraints, are used within the high-level requirements.

The design drivers will be strongly dependent upon the ConOps, including the operational environment, orbit, and mission duration requirements. For science missions, the design drivers may include, at a minimum, the mission launch date, duration, and orbit. If alternative orbits are to be considered, a separate concept is needed for each orbit. Exploration missions must consider the destination, the duration, the operational sequence (and system configuration changes), and the in situ exploration activities that allow the exploration to succeed.

The end result of this step is the discovery and delineation of the system’s goals, which generally express the agreements, desires, and requirements of the eventual users of the system. The high-level requirements and success criteria are examples of the products representing the consensus of the stakeholders.

4.1.1.3 Outputs

Typical outputs for capturing stakeholder expectations would include the following:

- **Top-Level Requirements and Expectations:** These would be the top-level requirements and expectations (e.g., needs, wants, desires, capabilities, constraints, and external interfaces) for the product(s) to be developed.
- **ConOps:** This describes how the system will be operated during the life-cycle phases to meet stakeholder expectations. It describes the system characteristics from an operational perspective and helps facilitate an understanding of the system goals. Examples would be the ConOps document or a DRM.

4.1.2 Stakeholder Expectations Definition Guidance

4.1.2.1 Concept of Operations

The ConOps is an important component in capturing stakeholder expectations, requirements, and the architecture of a project. It stimulates the development of the requirements and architecture related to the user elements of the system. It serves as the basis for subsequent definition documents such as the operations plan, launch and early orbit plan, and operations handbook and provides the foundation for the long-range operational planning activities such as operational facilities, staffing, and network scheduling.

The ConOps is important for all projects. For science projects, the ConOps describes how the systems will be operated to achieve the measurement set required for a

**Note:** It is extremely important to involve stakeholders in all phases of a project. Such involvement should be built in as a self-correcting feedback loop that will significantly enhance the chances of mission success. Involving stakeholders in a project builds confidence in the end product and serves as a validation and acceptance with the target audience.
successful mission. They are usually driven by the data volume of the measurement set. The ConOps for exploration projects is likely to be more complex. There are typically more operational phases, more configuration changes, and additional communication links required for human interaction. For human spaceflight, functions and objectives must be clearly allocated between human operators and systems early in the project.

The ConOps should consider all aspects of operations including integration, test, and launch through disposal. Typical information contained in the ConOps includes a description of the major phases; operation timelines; operational scenarios and/or DRM; end-to-end communications strategy; command and data architecture; operational facilities; integrated logistic support (resupply, maintenance, and assembly); and critical events. The operational scenarios describe the dynamic view of the systems’ operations and include how the system is perceived to function throughout the various modes and mode transitions, including interactions with external interfaces. For exploration missions, multiple DRMs make up a ConOps. The design and performance analysis leading to the requirements must satisfy all of them. Figure 4.1-3

![Figure 4.1-3 Typical ConOps development for a science mission](image_url)

![Figure 4.1-4 Example of an associated end-to-end operational architecture](image_url)
illustrates typical information included in the ConOps for a science mission, and Figure 4.1-4 is an example of an end-to-end operational architecture. For more information about developing the ConOps, see ANSI/AIAA G-043-1992, *Guide for the Preparation of Operational Concept Documents.*

The operation timelines provide the basis for defining system configurations, operational activities, and other sequenced related elements necessary to achieve the mission objectives for each operational phase. It describes the activities, tasks, and other sequenced related elements necessary to achieve the mission objectives in each of the phases. Depending on the type of project (science, exploration, operational), the timeline could become quite complex.

The timeline matures along with the design. It starts as a simple time-sequenced order of the major events and matures into a detailed description of subsystem operations during all major mission modes or transitions. An example of a lunar sortie timeline and DRM early in the life cycle are shown in Figures 4.1-5a and b, respectively. An example of a more detailed, integrated time-line later in the life cycle for a science mission is shown in Figure 4.1-6.

An important part of the ConOps is defining the operational phases, which will span project Phases D, E, and F. The operational phases provide a time-sequenced
4.0 System Design

structure for defining the configuration changes and operational activities needed to be carried out to meet the goals of the mission. For each of the operational phases, facilities, equipment, and critical events should also be included. Table 4.1-1 identifies some common examples of operational phases for a NASA mission.

![Timeline Diagram]

Figure 4.1-6 Example of a more detailed, integrated timeline later in the life cycle for a science mission
### Table 4.1-1 Typical Operational Phases for a NASA Mission

<table>
<thead>
<tr>
<th>Operational Phase</th>
<th>Description</th>
</tr>
</thead>
</table>
| Integration and test operations          | **Project Integration and Test:** During the latter period of project integration and test, the system is tested by performing operational simulations during functional and environmental testing. The simulations typically exercise the end-to-end command and data system to provide a complete verification of system functionality and performance against simulated project operational scenarios.  
**Launch Integration:** The launch integration phase may repeat integration and test operational and functional verification in the launch-integrated configuration. |
| Launch operations                        | **Launch:** Launch operation occurs during the launch countdown, launch ascent, and orbit injection. Critical event telemetry is an important driver during this phase.  
**Deployment:** Following orbit injection, spacecraft deployment operations reconfigure the spacecraft to its orbital configuration. Typically, critical events covering solar array, antenna, and other deployments and orbit trim maneuvers occur during this phase.  
**In-Orbit Checkout:** In-orbit checkout is used to perform a verification that all systems are healthy. This is followed by on-orbit alignment, calibration, and parameterization of the flight systems to prepare for science operations. |
| Science operations                       | The majority of the operational lifetime is used to perform science operations.                                                                                                                                 |
| Safe-hold operations                     | As a result of on-board fault detection or by ground command, the spacecraft may transition to a safe-hold mode. This mode is designed to maintain the spacecraft in a power positive, thermally stable state until the fault is resolved and science operations can resume. |
| Anomaly resolution and maintenance operations | Anomaly resolution and maintenance operations occur throughout the mission. They may require resources beyond established operational resources.                                                              |
| Disposal operations                      | Disposal operations occur at the end of project life. These operations are used to either provide a controlled reentry of the spacecraft or a repositioning of the spacecraft to a disposal orbit. In the latter case, the dissipation of stored fuel and electrical energy is required. |
The Technical Requirements Definition Process transforms the stakeholder expectations into a definition of the problem and then into a complete set of validated technical requirements expressed as “shall” statements that can be used for defining a design solution for the Product Breakdown Structure (PBS) model and related enabling products. The process of requirements definition is a recursive and iterative one that develops the stakeholders’ requirements, product requirements, and lower level product/component requirements (e.g., PBS model products such as systems or subsystems and related enabling products such as external systems that provide or consume data). The requirements should enable the description of all inputs, outputs, and required relationships between inputs and outputs. The requirements documents organize and communicate requirements to the customer and other stakeholders and the technical community.

Technical requirements definition activities apply to the definition of all technical requirements from the program, project, and system levels down to the lowest level product/component requirements document.

### 4.2.1 Process Description

Figure 4.2-1 provides a typical flow diagram for the Technical Requirements Definition Process and identifies typical inputs, outputs, and activities to consider in addressing technical requirements definition.

---

**Figure 4.2-1 Technical Requirements Definition Process**

---

Please purchase PDF Split-Merge on www.verypdf.com to remove this watermark.
4.2.1.1 Inputs

Typical inputs needed for the requirements process would include the following:

- **Top-Level Requirements and Expectations**: These would be the agreed-to top-level requirements and expectations (e.g., needs, wants, desires, capabilities, constraints, external interfaces) for the product(s) to be developed coming from the customer and other stakeholders.

- **Concept of Operations**: This describes how the system will be operated during the life-cycle phases to meet stakeholder expectations. It describes the system characteristics from an operational perspective and helps facilitate an understanding of the system goals. Examples would be a ConOps document or a DRM.

4.2.1.2 Process Activities

The top-level requirements and expectations are initially assessed to understand the technical problem to be solved and establish the design boundary. This boundary is typically established by performing the following activities:

- Defining constraints that the design must adhere to or how the system will be used. The constraints are typically not able to be changed based on tradeoff analyses.

- Identifying those elements that are already under design control and cannot be changed. This helps establish those areas where further trades will be performed to narrow potential design solutions.

- Establishing physical and functional interfaces (e.g., mechanical, electrical, thermal, human, etc.) with which the system must interact.

- Defining functional and behavioral expectations for the range of anticipated uses of the system as identified in the ConOps. The ConOps describes how the system will be operated and the possible use-case scenarios.

With an overall understanding of the constraints, physical/functional interfaces, and functional/behavioral expectations, the requirements can be further defined by establishing performance criteria. The performance is expressed as the quantitative part of the requirement to indicate how well each product function is expected to be accomplished.

Finally, the requirements should be defined in acceptable “shall” statements, which are complete sentences with a single “shall” per statement. See Appendix C for guidance on how to write good requirements and Appendix E for validating requirements. A well-written requirements document provides several specific benefits to both the stakeholders and the technical team, as shown in Table 4.2-1.

4.2.1.3 Outputs

Typical outputs for the Technical Requirements Definition Process would include the following:

- **Technical Requirements**: This would be the approved set of requirements that represents a complete description of the problem to be solved and requirements that have been validated and approved by the customer and stakeholders. Examples of documentation that capture the requirements are a System Requirements Document (SRD), Project Requirements Document (PRD), Interface Requirements Document (IRD), etc.

- **Technical Measures**: An established set of measures based on the expectations and requirements that will be tracked and assessed to determine overall system or product effectiveness and customer satisfaction. Common terms for these measures are Measures of Effectiveness (MOEs), Measures of Performance (MOPs), and Technical Performance Measures (TPMs). See Section 6.7 for further details.

4.2.2 Technical Requirements Definition Guidance

4.2.2.1 Types of Requirements

A complete set of project requirements includes the functional needs requirements (what functions need to be performed), performance requirements (how well these functions must be performed), and interface requirements (design element interface requirements). For space projects, these requirements are decomposed and allocated down to design elements through the PBS.

Functional, performance, and interface requirements are very important but do not constitute the entire set of requirements necessary for project success. The space segment design elements must also survive and continue to perform in the project environment. These environmental drivers include radiation, thermal, acoustic, mechanical loads, contamination, radio frequency, and others. In addition, reliability requirements drive design choices in design robustness, failure tolerance, and redundancy. Safety requirements drive design choices in providing diverse functional redundancy. Other spe-
4.0 System Design

Table 4.2-1 Benefits of Well-Written Requirements

<table>
<thead>
<tr>
<th>Benefit</th>
<th>Rationale</th>
</tr>
</thead>
<tbody>
<tr>
<td>Establish the basis for agreement between the stakeholders and the developers on what the product is to do</td>
<td>The complete description of the functions to be performed by the product specified in the requirements will assist the potential users in determining if the product specified meets their needs or how the product must be modified to meet their needs. During system design, requirements are allocated to subsystems (e.g., hardware, software, and other major components of the system), people, or processes.</td>
</tr>
<tr>
<td>Reduce the development effort because less rework is required to address poorly written, missing, and misunderstood requirements</td>
<td>The Technical Requirements Definition Process activities force the relevant stakeholders to consider rigorously all of the requirements before design begins. Careful review of the requirements can reveal omissions, misunderstandings, and inconsistencies early in the development cycle when these problems are easier to correct thereby reducing costly redesign, remanufacture, recoding, and retesting in later life-cycle phases.</td>
</tr>
<tr>
<td>Provide a basis for estimating costs and schedules</td>
<td>The description of the product to be developed as given in the requirements is a realistic basis for estimating project costs and can be used to evaluate bids or price estimates.</td>
</tr>
<tr>
<td>Provide a baseline for validation and verification</td>
<td>Organizations can develop their validation and verification plans much more productively from a good requirements document. Both system and subsystem test plans and procedures are generated from the requirements. As part of the development, the requirements document provides a baseline against which compliance can be measured. The requirements are also used to provide the stakeholders with a basis for acceptance of the system.</td>
</tr>
<tr>
<td>Facilitate transfer</td>
<td>The requirements make it easier to transfer the product to new users or new machines. Stakeholders thus find it easier to transfer the product to other parts of their organization, and developers find it easier to transfer it to new stakeholders or reuse it.</td>
</tr>
<tr>
<td>Serve as a basis for enhancement</td>
<td>The requirements serve as a basis for later enhancement or alteration of the finished product.</td>
</tr>
</tbody>
</table>

Functional requirements also may affect design choices. These may include producibility, maintainability, availability, upgradeability, human factors, and others. Unlike functional needs requirements, which are decomposed and allocated to design elements, these requirements are levied across major project elements. Designing to meet these requirements requires careful analysis of design alternatives. Figure 4.2-2 shows the characteristics of functional, operational, reliability, safety, and specialty requirements. Top-level mission requirements are generated from mission objectives, programmatic constraints, and assumptions. These are normally grouped into function and performance requirements and include the categories of requirements in Figure 4.2-2.

**Functional Requirements**

The functional requirements need to be specified for all intended uses of the product over its entire lifetime. Functional analysis is used to draw out both functional and performance requirements. Requirements are partitioned into groups, based on established criteria (e.g., similar functionality, performance, or coupling, etc.), to facilitate and focus the requirements analysis. Functional and performance requirements are allocated to functional partitions and subfunctions, objects, people, or processes. Sequencing of time-critical functions is considered. Each function is identified and described in terms of inputs, outputs, and interface requirements from the top down so that the decomposed functions are recognized as part of larger functional groupings. Functions are arranged in a logical sequence so that any specified operational usage of the system can be traced in an end-to-end path to indicate the sequential relationship of all functions that must be accomplished by the system.

It is helpful to walk through the ConOps and scenarios asking the following types of questions: what functions need to be performed, where do they need to be performed, how often, under what operational and environ-