Program Engineering

(Systems Engineering at the Program Level)

Bob Easter / Rick Pomphrey / Chuck Weisbin
December, 2002

Contents

◈ Part 1: Program Engineering Concepts

Contents (cont.)

- **Part 2: Program Engineering in the New Millennium Program** (Rick Pomphrey)
  - Program Engineering in Action at JPL
- **Part 3: Program Quantitative Investment Analysis** (Chuck Weisbin)
  - Setting Priorities and Making decisions

Part 1

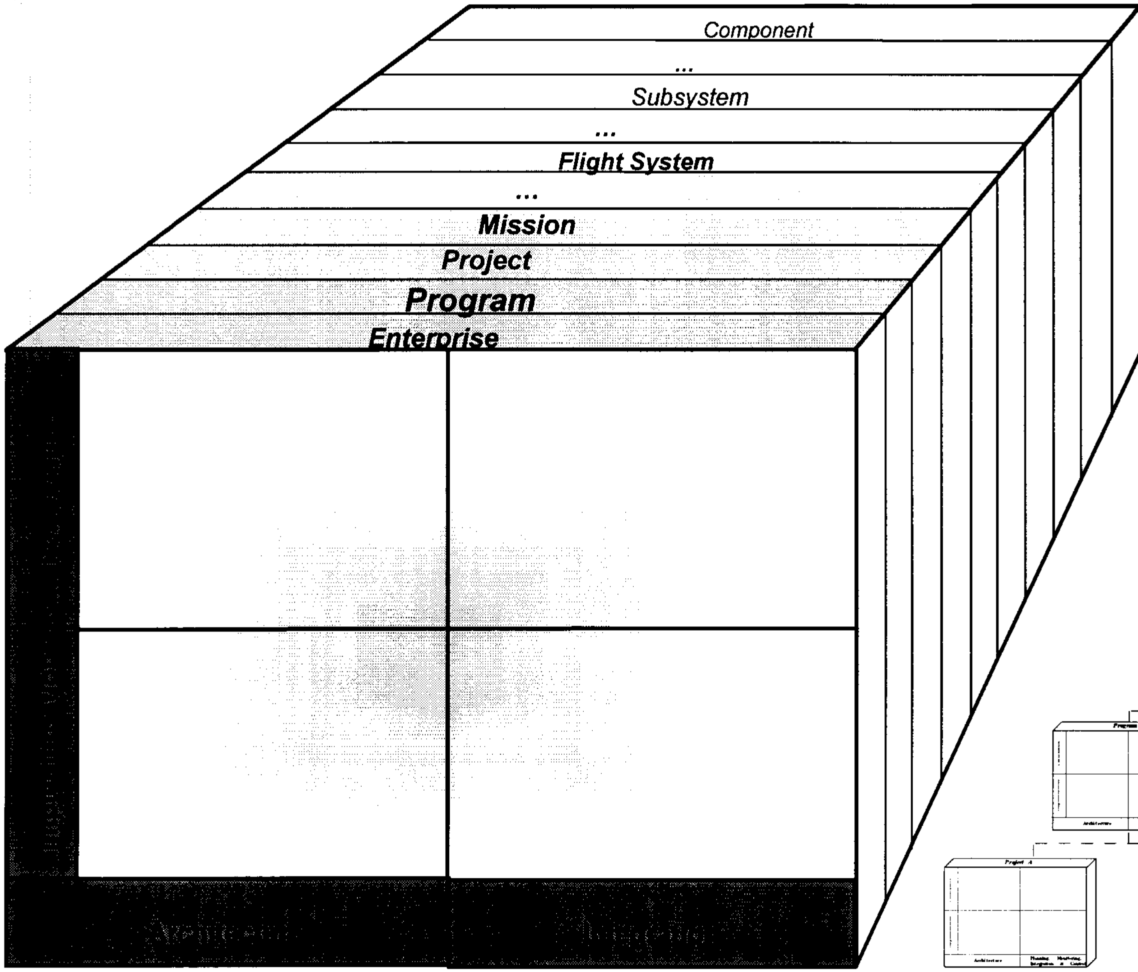
Program Engineering Concepts

Bob Easter

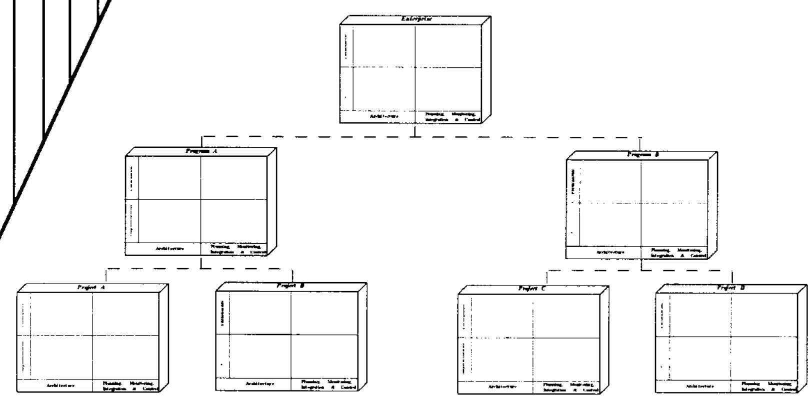
Background - What is Program Engineering?

- Application of Systems Engineering disciplines in support of Program Management.
- PROGRAM = Series or set of projects or missions aimed at an overall goal or set of goals, often involving development and utilization of new technology.
- New discipline emerging at JPL because of program management assignments from NASA.
- JPL Program System Engineering experience to date suggests need for systems engineering tool set for programs similar to but different from those needed for other levels.

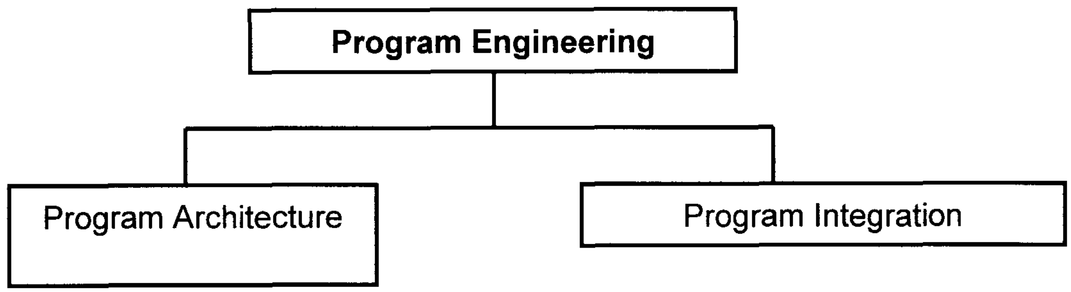
One System Engineering Model



Each level of system engineering may contain some elements of the next lower-level type. That is, the various system engineering levels are considered to be “fractal” or “nested.”



Program Engineering Simplified Functional Chart



Program Architecture

- Support Development of Program Architecture

Program Integration

- Support Development of Program Plans and Processes

Formulation Phase

Implementation Phase

Program Architecture

- Program Architecture What-ifs, e.g., Risk Rebalancing
- Flowdown of Requirements During Formulation of Additional Program Elements

Program Integration

- Program Configuration Management, incl…
  - Performance Measures, Reserves and Margins
  - Plans and Requirements
  - Budgets and Schedules
  - Risk

Program Engineering Activities-to-Date (Oct '02)

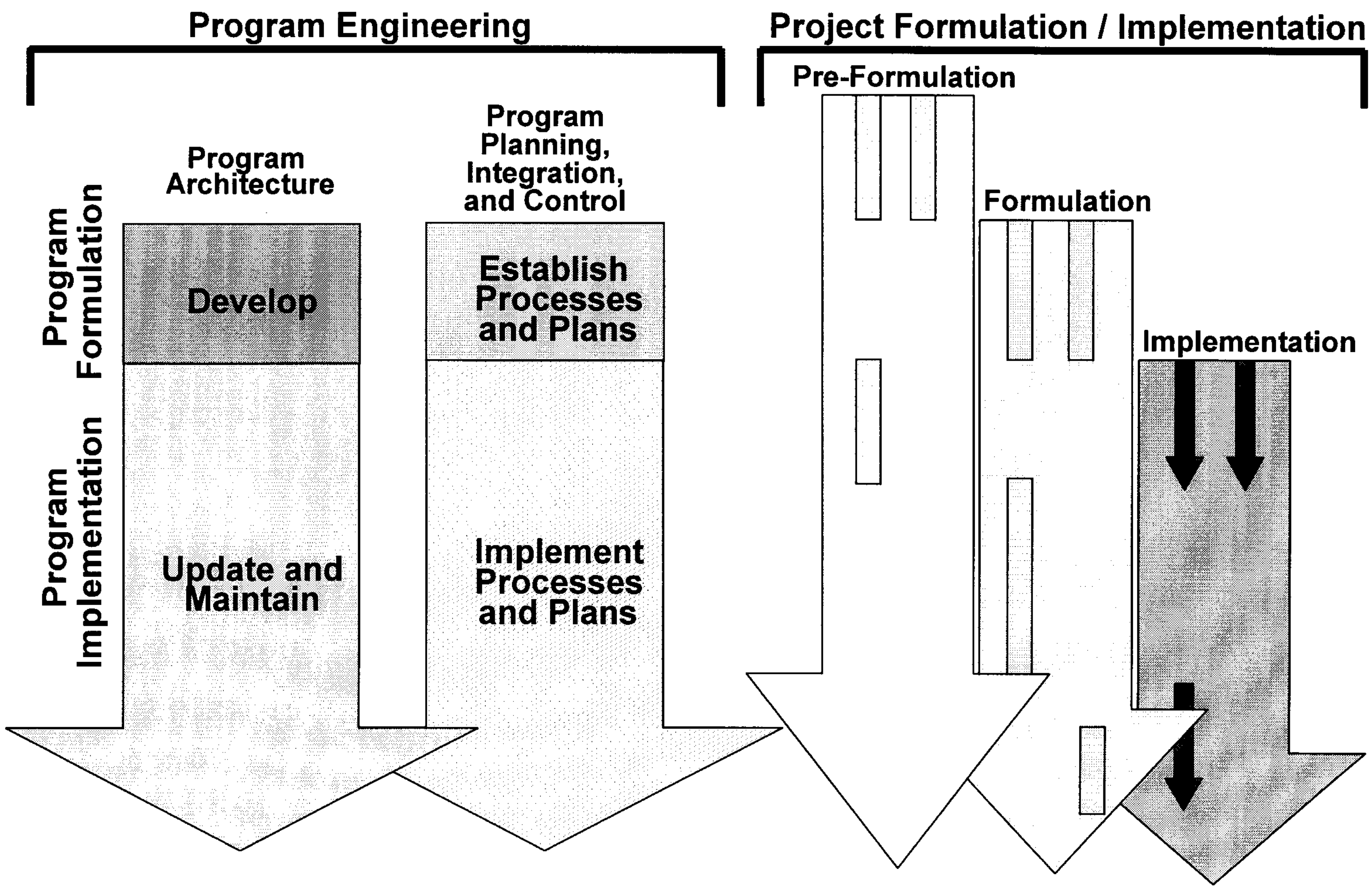
- Mars Exploration Program – Program Plans (including Program-Level Requirements Appendices), Outpost Programs Formulation including Availability Analysis, Risk Tree Program Architecture, Technology Investment Analysis, Engineering Environment Development
- Navigator Program – Program Risk-based Architecture Formulation, Program Plans and PLRAs, Program Risk Management and Review Plans
- Focused Physical Oceanography and Solid Earth Program – Program Phase II Formulation Planning
- Sun Earth Connection Theme (NASA HQ) – Proposal preparation for Program Engineering of SEC Theme, emphasizing Availability and Coverage
- NMP – Flight Options Analysis, Program Engineering Workbench, Mission Data Information System, Investment Analysis
- IR&D - Funded preliminary development of "Workbench" website, processes, and tools

Program Engineering Status

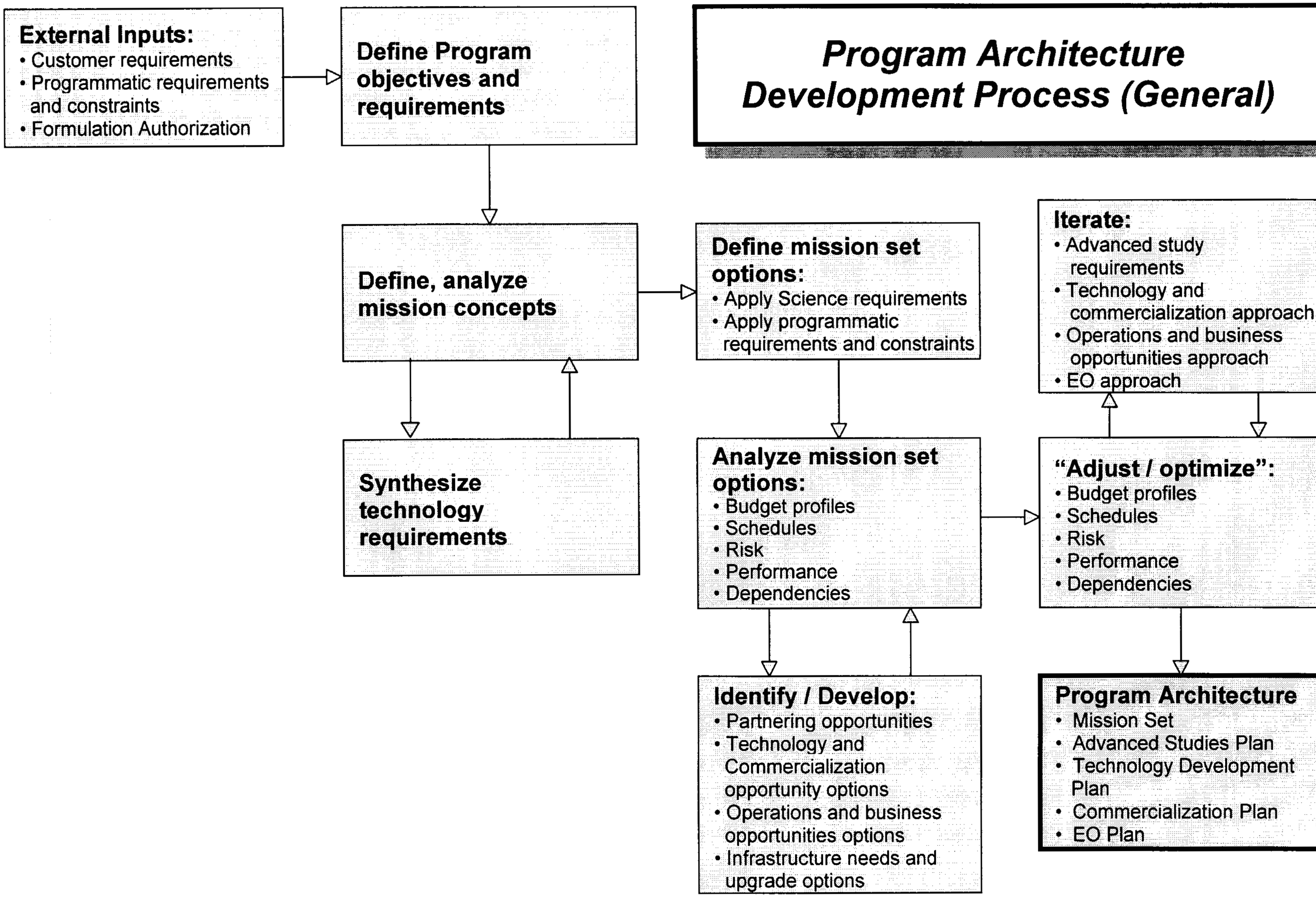
Program Architecture Formulation and Update Processes

[redacted]

Program Formulation and Implementation



Program Architecture Development Process (General)



[redacted]

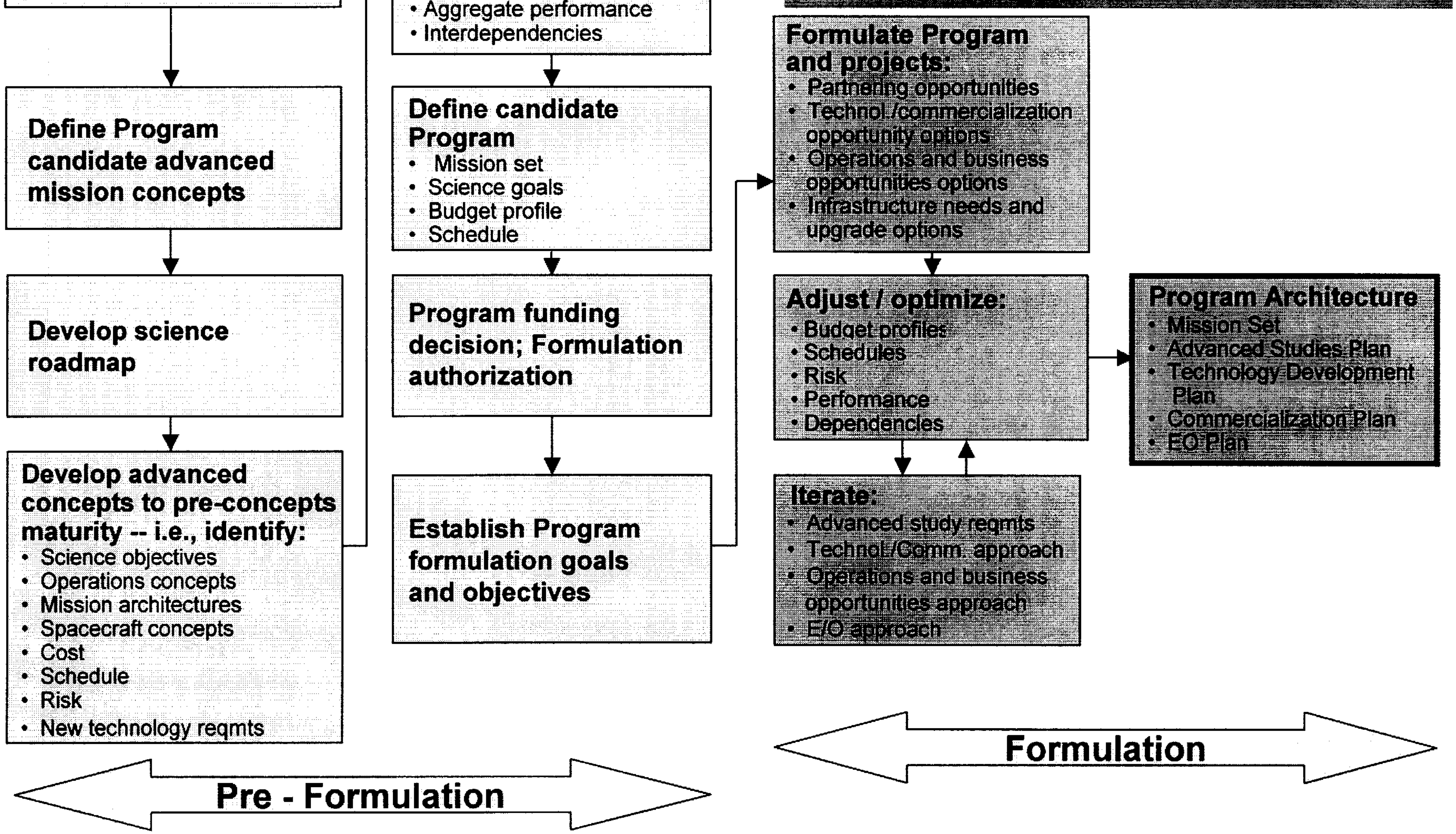
Program Architecture Formulation & Update Processes:

Four Cases

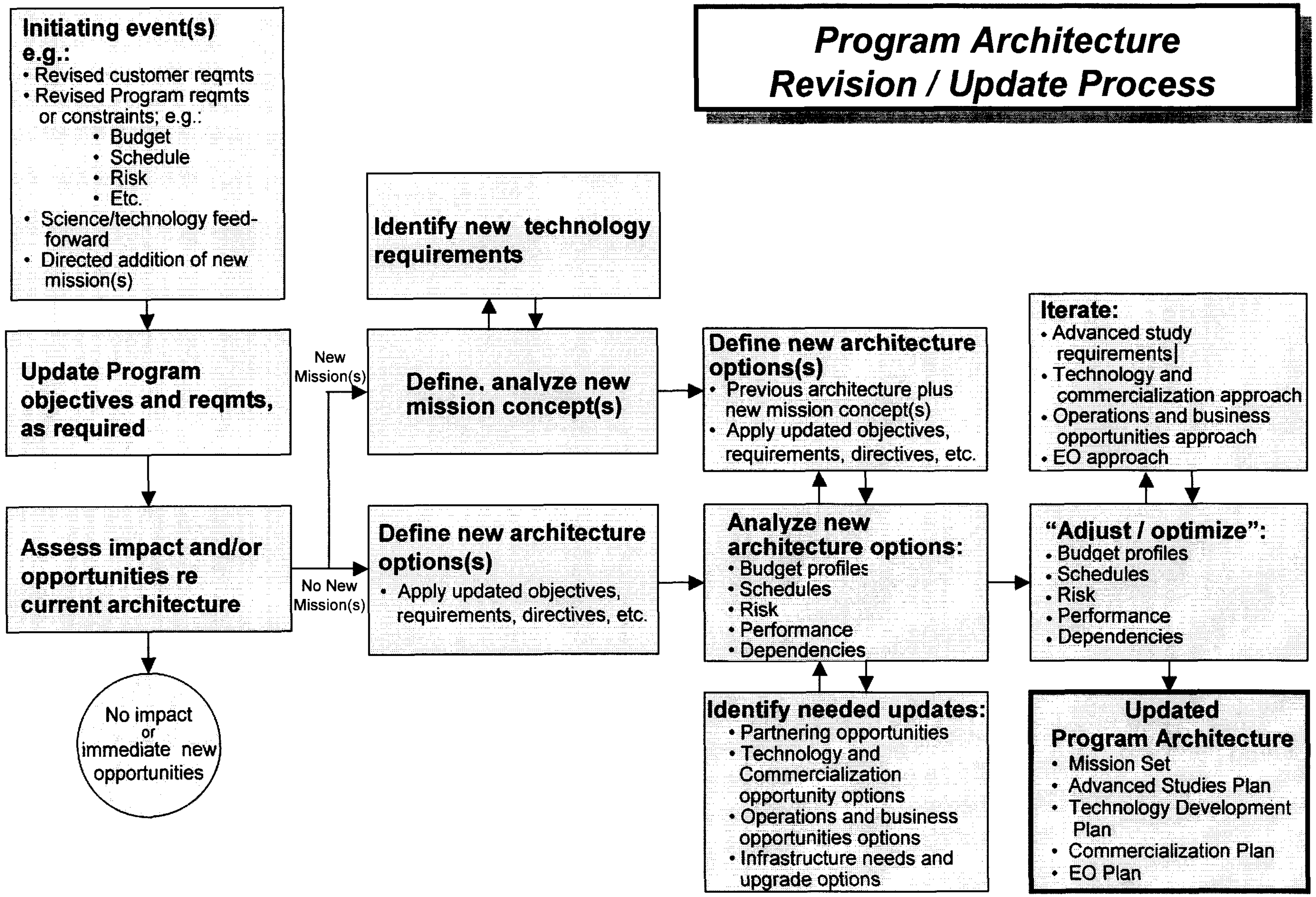
	Top-Down	Bottom-Up
New Architecture Formulation	• Starts with high-level, overarching program goal(s) • Definition of program goals and objectives precedes identification of candidate mission sets • Program defined as a set of candidate missions that lead to achievement of long-term program goal • Program formulation in part precedes project formulation • <u>Process outcome</u>: architecture for new program	• Starts with set of high-level mission concepts derived from high-priority science/technol. goals • Definition of candidate mission set precedes definition of program goals and objectives • Program defined as a collection of mission concepts salable as a set • Project and program formulation proceed in parallel • <u>Process outcome</u>: architecture for new program
Architecture Update./ Revision	• Starts with, e.g., change in program reqmts or constraints, directed addition of new activity, etc. • Assessment of impact focused on program goals and objectives • Project interdependencies key to planning response to changes • <u>Process outcome</u>: revised architecture for existing program	• Starts with, e.g., changing program reqmts or constraints, directed addition of new activity, etc. • Assessment of impact focused on individual projects • Project interdependencies secondary • <u>Process outcome</u>: revised architecture for existing program

*Program Architecture Formulation Process**

*Code S process, derived from Space Science Enterprise Management Handbook

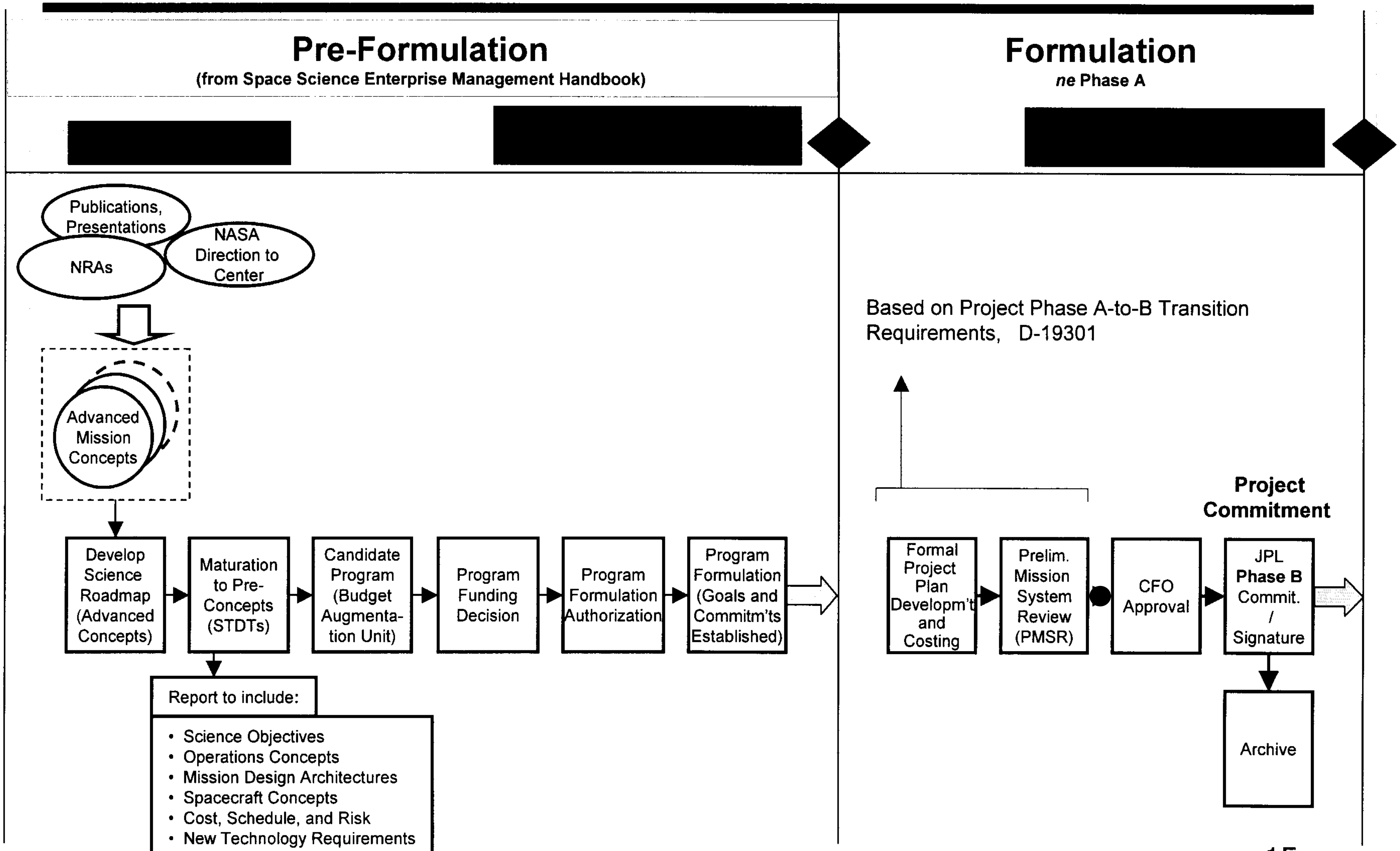


Program Architecture Revision / Update Process



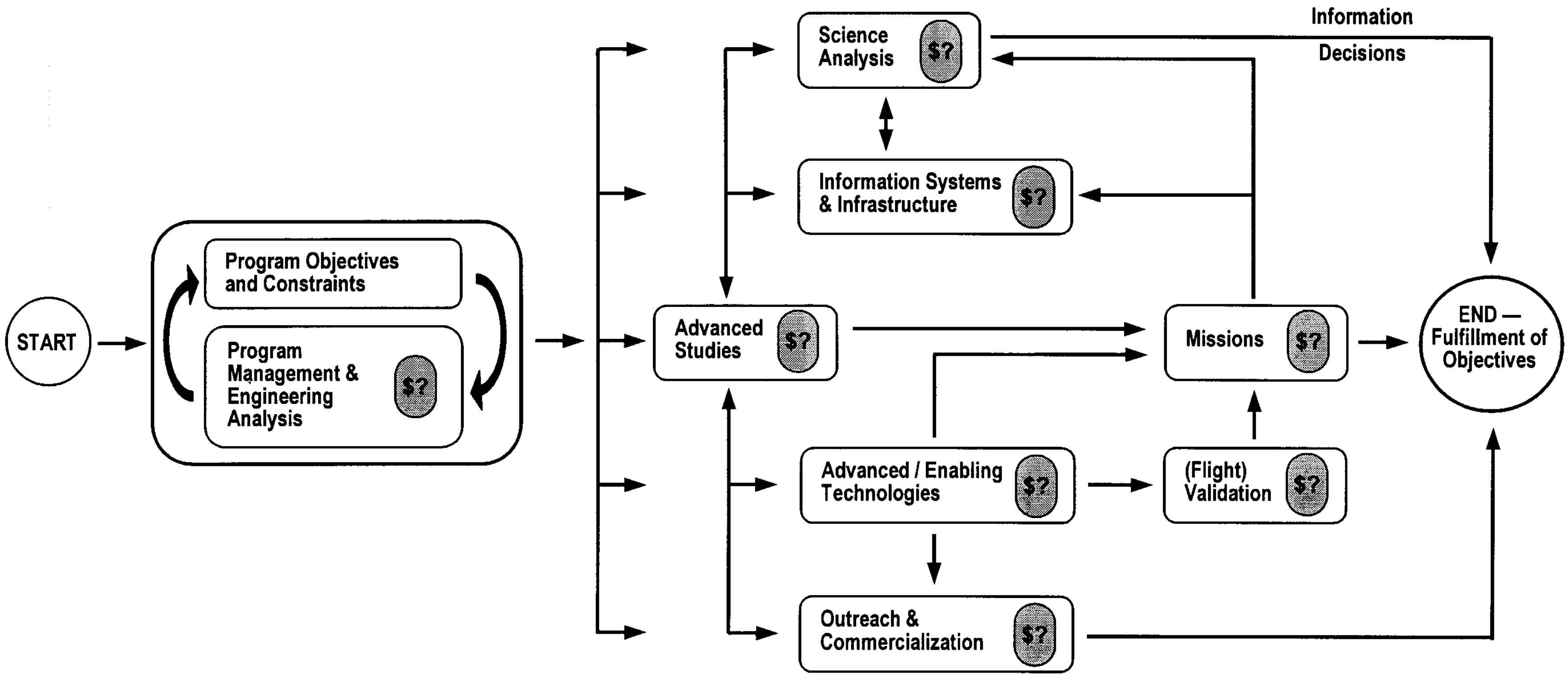
[redacted]

Proposed Commitment Process: JPL Program-Directed Projects
(for New Programs)



[redacted]

Principal Program Elements and Dependencies



Program Engineering – Some Elements of the Credo

- Understand and illuminate interrelationships among projects within a program, especially cost and risk interactions
- Emphasize what the program needs from the projects, rather than how the projects should do things
- Establish program engineering processes that make life easier for the projects, if possible
- Meet the program's needs via existing institutional project processes and moderate revisions thereto, rather than via imposition of new processes

Acknowledgements

- **Lynn Baroff**, **Warren Nogaki**, and **Dr. Kent Volkmer** have made major contributions to our growing understanding of Program Engineering.

Tools to Support Program Engineering

- ## Program Architecture Tools
  - Operational Availability Analysis, Coverage Analysis *
  - Interdependency Analysis
  - Risk / Probability of Success Analysis *
  - Technology Investment Analysis / Methodology
  - Trade-off / Option Analysis of Candidate Architectures
  - Parametric Costing / Resource Estimation & Analysis
  - High Level Program Milestone & Cost Manipulation and Analysis

- ## Program Integration Tools
  - Information Management System
  - Requirements Trace and Management
  - Budget and Schedule Reporting Standards and Rollups
  - Performance Metrics and Reporting
  - Program Plans, Policies, Processes
  - Integration Teams

What Is Availability (GAP) Analysis?

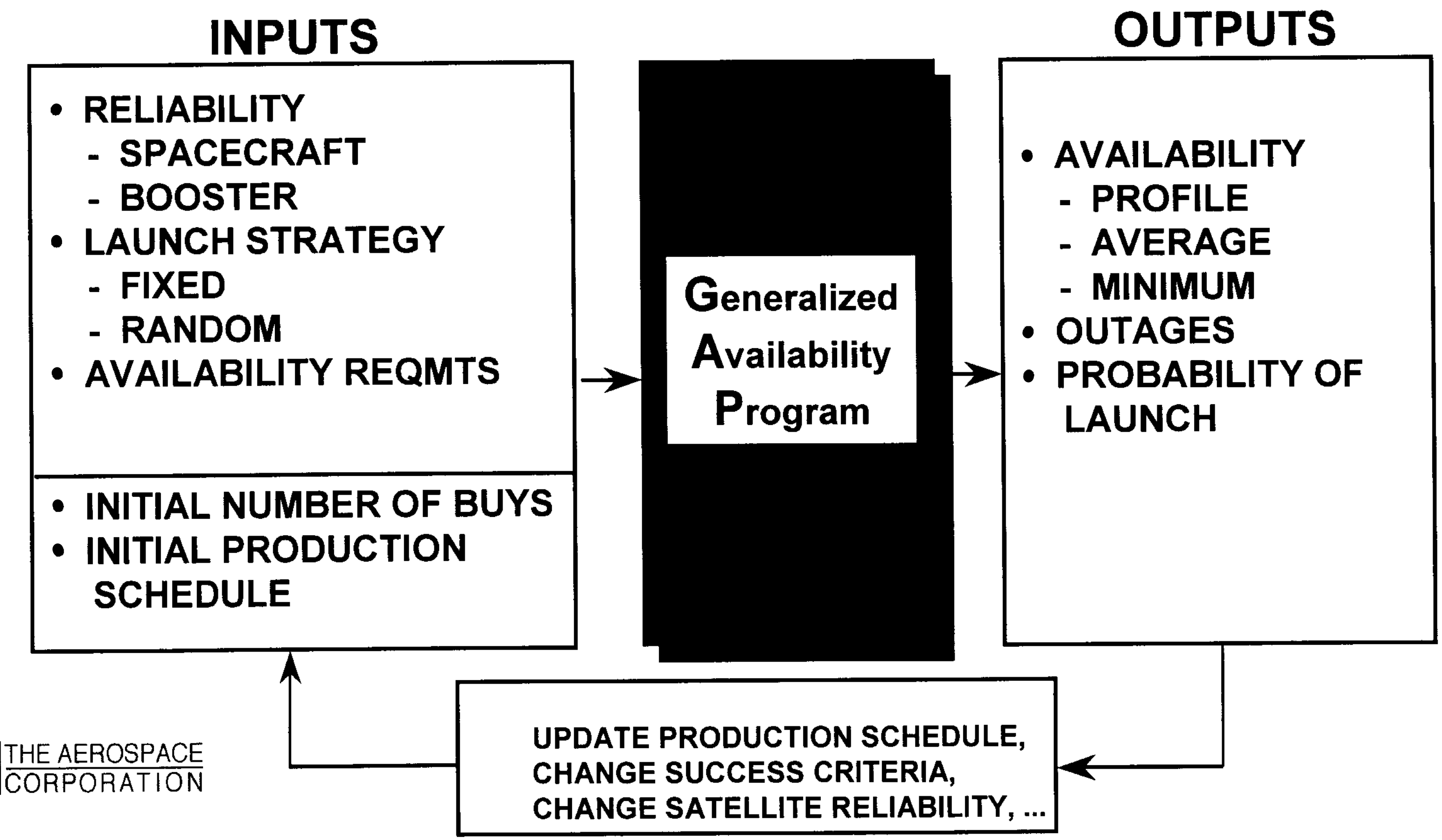
- Tool belonging to Aerospace Corp.
- Several Monte Carlo computer programs for computing replenishment and procurement uncertainties from satellite and launch reliability inputs
- Simulates mission lifetime of satellite system many times (3000) resulting in statistics to estimate procurement risk and system effectiveness
  - Probability of system outage or loss of data collection
  - Procurement need time distributions
  - Launch time distributions
  - Other outputs

THE AEROSPACE CORPORATION

What Is Availability (GAP) Analysis? (cont.)

Replenishment Sensitivity Analysis

Satellite Constellation



THE AEROSPACE CORPORATION

Previous Users for GAP

- All military navigation, communication, and weather satellite constellations including:
  - **Global Positioning System (GPS)**
  - **Defense Satellite Communications System (DSCS)**
  - **Defense Meteorological Satellite System (DMSP)**
  - **Defense Support Program (DSP)**
- Many civil and commercial communications and weather satellite constellations including:
  - **Geosynchronous Operational Environmental Satellite (NASA-GOES) System**
  - **Iridium**
  - **Teledesic**

THE AEROSPACE CORPORATION

- JPL Mars Outpost Study (surface architecture)

Constellation Coverage Analysis

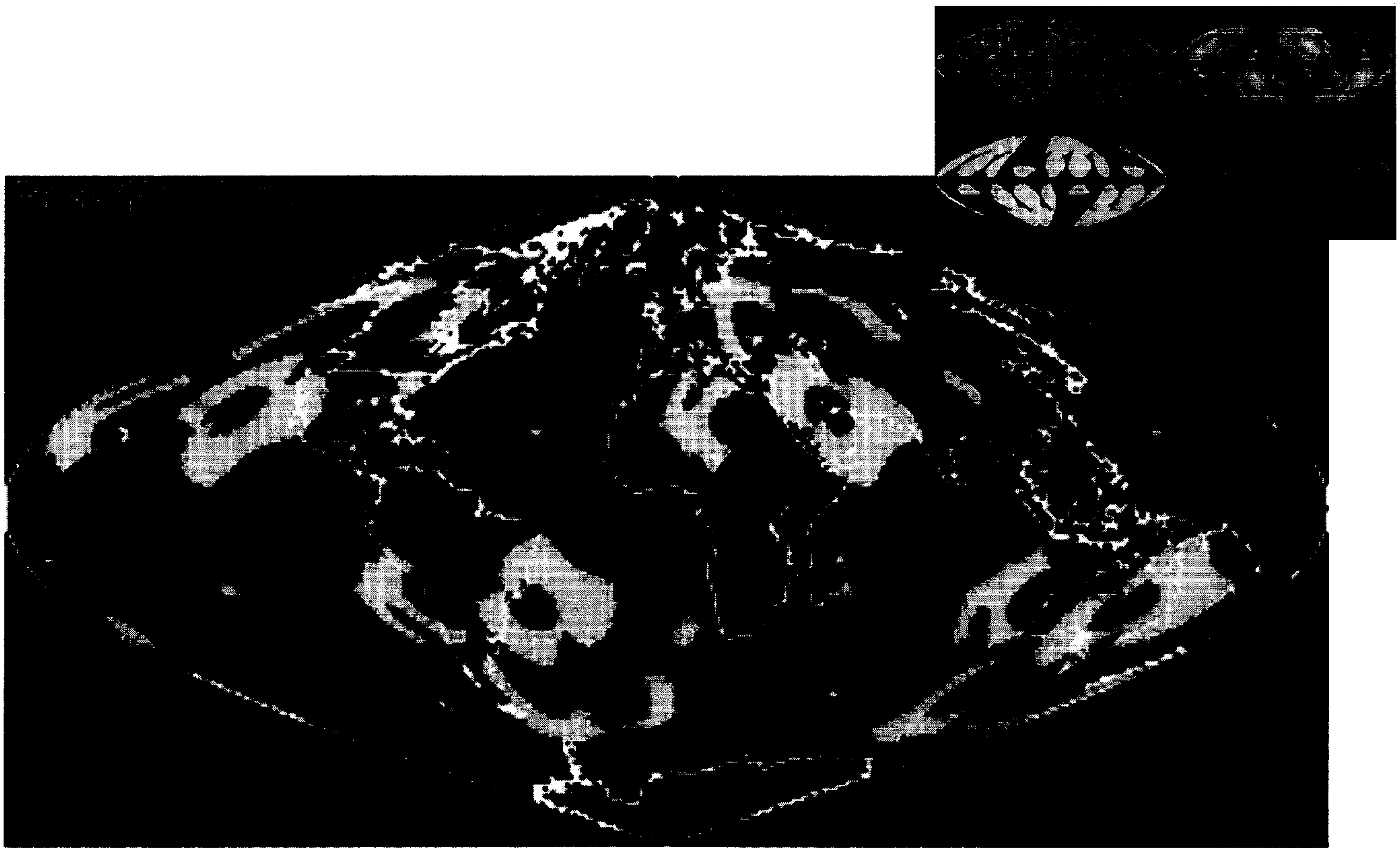
Constellation Design Capabilities

- Fast Station View Period Computation WITHOUT ORBIT PROGAPATION! (Using Ergodic Theory)
- Coverage Analysis of Complex Networks Using the "Visual Calculus"
  - Example: SAR GPS (Bistatic SAR)
    - Double Constellation
    - Complex Instrument Performance
    - Ability to Query Complex Statiscal Questions
- Formation Flight in Conic and Libration Orbits

[redacted]

Constellation Coverage Analysis

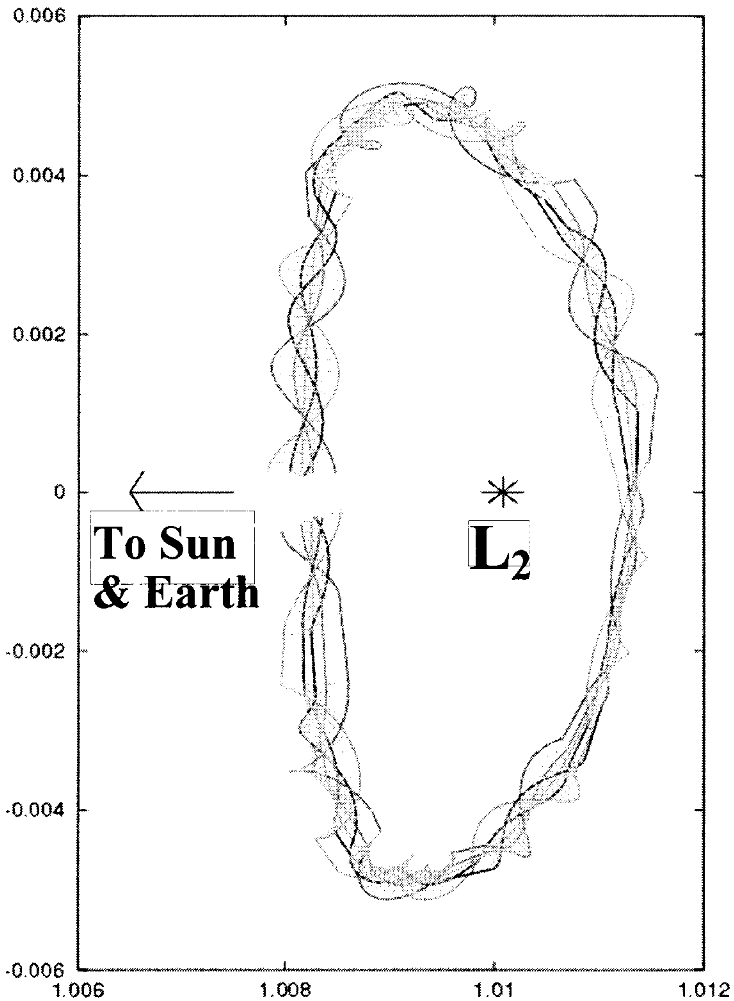
Combined Performance Pass / Fail Map

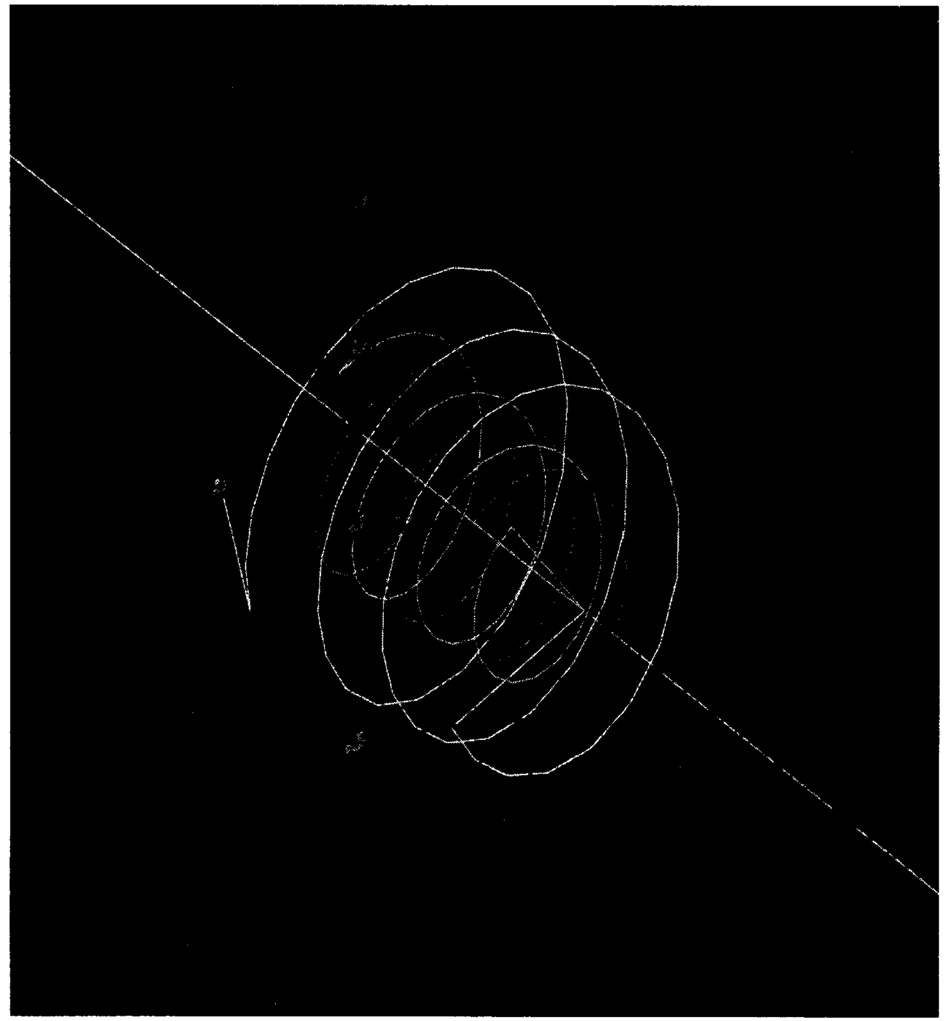


[redacted]

Constellation Coverage Analysis

TPF in Formation Flight Near L_2





[redacted]

First Order Benefit/Cost/Risk Analysis (Probability of Success)

- BCR approaches can provide a simple but powerful tool for comparing Program Alternatives
- A benefit/cost/risk ratio for mission implementation approach may be expressed
  - bcr = B x P / C
  - bcr is defined such that values of bcr less 1 are unattractive, and the larger the value, the more attactive. B is the basic benefit or payoff of accomplishing the mission's objectives; P is the probability of success; and C is the required investment or cost. In a Program Engineering context, differing implementations of given missions might be roughly compared by seeing which yields the higher bcr.

Use of Simple BCR Analysis

When, Where & Why Humans – An Example

- Assume the objective is to refurbish a spacecraft in LEO. It is estimated that the servicing can be accomplished with a probability of success of .85 by a shuttle mission costing $450M
- If this is deemed acceptable, then

  B must be $\geq$ 1 x $450M / .85 = $529M

  if the spacecraft can be replaced for an expected cost (cost/probability of success) of less than this — If, for instance, the spacecraft would cost $500M to replace — the shuttle servicing mission is not feasible, i.e. (bcr<1). If the replacement cost is $600M then the shuttle servicing mission is feasible, i.e.

  If B=$600M, then bcr_{human} = 1.13

- The bcr for an automated robotic servicing mission is

  bcr_{robot} = $600M x P_{robot} / C_{robot}

  So, for instance, if P_{robot} is .5, C_{robot} must be less than $265M if bcr_{robot} is to be greater than bcr_{human}