The Berlin Airlift

A systems engineering case study

Dr. Carlee Bishop
carlee.bishop@gtri.gatech.edu
Tutorial Outline

- Motivation
  - SE Education
  - SE Experiential Learning
  - SE Case Studies

- Berlin Airlift Case Study
  - Vehicle for training SE Leadership and Management
  - Case Study Learning Principals
    - Applied Systems Thinking
    - Organizational Behaviors
    - Leadership and Decision Making
    - Requirements and System Architecting
    - Project Management for Complex Systems
Berlin Airlift Case Study Author

Dr. Carlee Bishop
carlee.bishop@gtri.gatech.edu
Georgia Tech Research Center
(404)407-6335

- Executive Director, Professional Masters Degree in Applied Systems Engineering
- Principal Research Engineer, GTRI
- Project Director / Program Manager, GTRI research programs
- Adjunct Associate Prof, GT School of Aerospace Engineering
- Former C-141 Pilot, Instructor, Evaluator
- Former Professor, Air Force Academy
- PhD, Auburn
- MSEE, Georgia Tech
- BSEE, Air Force Acad.
- Certified Systems Engineering Professional (CSEP)
Additional Credits

This tutorial draws material from a number of short courses and masters degree courses taught at the Georgia Institute of Technology. The following Georgia Tech research faculty contributed to the development of this course material:

- Tom McDermott (Course Director - Leading SE Teams)
- Marty Broadwell (Instructor – Leading SE Teams)
- Tommer Ender (Course Director - SOS & Architecture; Instructor – Leading SE Teams, Fundamentals of Modern Systems Engineering)
- Jack Zentner (Course Director - Advanced Problem Solving; Instructor – Fundamentals of Modern Systems Engineering)
- Dennis Folds (Course Director – Human Systems Integration)

Additional details on each of these courses is available at:

www.pmase.gatech.edu and
www.pe.gatech.edu/subjects/systems-engineering
Motivation

- Case Studies in SE
- AFIT Case Studies
- Berlin Airlift - Why
- Case Study Learning Principals
Why is this Important?

- System complexity is increasing, affecting more around us
- Issues of Systems of Systems (SoS) and complex systems are pervading all of engineering (not just DoD, but also commercial networks, energy, sustainability, etc.)
- SE education is lacking engineering fundamentals - too much process (management), not enough engineering rigor
- SE research has fallen behind in the need to address complex system problems
Complexity in Systems Engineering

- Multiple, often inversely related requirements
- Ambiguous and competing visions of solutions
- Constraints in tension: cost, schedule, performance…
- Many sources of information, expertise, & innovation
  - No source has all
  - Almost all sources are required
- Organizational dissonance among participants/stakeholders
  - Conflicting goals (including implicit)
  - Varying levels of commitment/investment
  - Varying levels of risk tolerance
  - Missing or Inadequate resources
Why SE Case Studies

Case studies in engineering:
- Used to introduce students to real programs and real problems
- Presents open ended problems that student teams work and then compare to actual outcomes
- Allow instructors to introduce topics too difficult to convey through just lectures and homework

Systems engineering (SE) case studies:
- Special Category of Engineering Case Studies Focus on Applied SE
- Air Force Institute of Technology (AFIT) Cases: http://www.afit.edu/cse/cases.cfm
- Extend Applied Systems Engineering to Berlin Airlift

The Berlin Airlift:
- Provides forum for Experiential treatment of SE concepts
- Promotes innovative, interdisciplinary SE education
- Melds theory & experience.
- Advances systems thinking & practice further into technological future
Berlin Airlift Case Study Objectives

- Experience Learning by Doing
- Identify conditions that foster good SE practices.
- Identify long term consequences of the SE and programmatic decisions on program success.
- Exercise Team Leadership
- Develop a “System” Architecture
- Exercise your Systems Thinking.
Basic Functions of Systems Engineering

- Systems Engineering Processes
- Systems Engineering Methods and Tools
- Systems Management
- System Product or Service Under Development
- System Development Team
Growing Functions of Systems Engineering

- Maturity in Systems Thinking, Complexity
  - Systems Engineering Methods and Tools
  - Systems Engineering Processes
  - Systems Management
  - System Product or Service Under Development
  - System Development Team
  - Foundation in Leadership
Disciplines of the Systems Engineer

- **System Design**: Creating the integrated set of interrelated components that interact in an organized fashion toward a common objective.
- **Systems Engineering**: Creating and executing the process to ensure the stakeholder’s needs are fully satisfied throughout the system’s life cycle.
- **Systems Management**: Managing the system’s life cycle and the processes that contribute to its development and use.
- **Systems Thinking**: Taking a “big picture” or holistic view of large-scale and complex problems and their proposed solutions.
A Model of Systems Thinking & Management

- Technical
- Project
- Enterprise

Political Factors

Program Requirements

User Requirements

System Design & Analysis

Concurrent Engineering

Available Technical Skills

Team Structure

Development Models

Development Processes

Organizational Structure

Team Leadership

System

• Technical
• Project
• Enterprise
Applied Systems Thinking

- SE Leadership/Management Model
- Experiential Learning
- Berlin Airlift Application
Keys to Systems Thinking & Management

- Leadership in a Complex Environment
- Organization and Culture
- Team Capabilities
- Lifecycle Management
- Business Planning
- Risk Management
- Stakeholders
- Processes
- Management Methods & Feedback
There is consensus on primary mechanisms that enable systems thinking development in engineers

1. Experiential learning
2. Individual characteristics
3. Supportive environment

Experiential Learning

- Center of Learning is Experience

- Students can enter the Learning Cycle at any point based on their Experiences and Learning Styles

- We use Case Studies to facilitate Experiential Learning

Experience Based Learning Systems, Inc
http://learningfromexperience.com/
Your Viewpoint

- **Hard systems methods:**
  - Thinking about the system: components, interfaces, processes, technology, engineering
  - Quantitative analysis and evaluation

- **Soft systems methods:**
  - Thinking *from* the system: policy, governance, enterprise, behavior, utility
  - Insight into problem definition and usefulness of solution

- Systems thinking combines both of these

- The combined process of **Synthesis** (putting things together) and **Analysis** (breaking things down) is enabled by **Inquiry**, *the human process of investigation via dialogue and directed discussion of outcomes*. The combination of the three constitute the discipline of Systems Thinking (Ackoff 1999, Senge 2006)
Understanding & Synthesizing a System

Boundaries
- Scope: Boundary, Interior, and Exterior

Inter-relationships
- Function: Inputs, Outputs, Transformations
- Structure: Hierarchy, Openness, Emergence
- Governance: Command, Control, Communication

Perspective
- Process: Wholes, Parts, Relationships
- Vision: Variety, Economy, Harmony

Systems Thinking in Practice

- Take the Broad View
- Find the Root Cause
- Lead
- Listen
# System Thinking Tools for Orientation: The Problem Spectrum

<table>
<thead>
<tr>
<th>Tame Problems</th>
<th>Solvable Problems</th>
<th>Wicked Problems</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Formulation</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Situation/Need Clearly Defined</td>
<td>Situation/Need Can be Defined</td>
<td>Situation/Need Poorly Understood or Ill-Defined</td>
</tr>
<tr>
<td><strong>Solution</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- Understood</td>
<td>Not Understood, Difficult</td>
<td>Not Possible</td>
</tr>
<tr>
<td><strong>Toolset</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
| - Analysis Tools  
  Equations/Algorithms  
  Process Flows  
  Models & Simulations | Thinking Tools  
  Mind Maps  
  Logic Models  
  Causal Models |                |
Sample Tools for Systems Understanding

- **SWOT** (Strengths, Weaknesses, Opportunities, Threats) Analysis – Weirich: A process for determining internal and external factors key to achieving a chosen objective
- **OODA** (Observe, Orient Decide, Act) – Boyd: an approach to create situational awareness around system behaviors to aid in decision making
- **Logic Model** – an approach to aid in understanding structure & process. Links outcomes (both short- and long-term) with program activities/processes and the theoretical assumptions/principles of the program
Exercise: Berlin Airlift Application

Introduction and Set Up
SWOT Analysis
Identify SMEs Needed
Video Clip

- [http://www.youtube.com/watch?v=UOsqxp1ZDts](http://www.youtube.com/watch?v=UOsqxp1ZDts)
Operations Vittles

Setting the Stage: At the conclusion of WWII, the Soviets, Americans, British and French divided Germany into occupation zones. A delicate balance of power surfaced between the once united allies. Although Berlin was located in the Soviet zone, it was also divided among the four powers. As western Germany was rebuilding and preparing to govern itself, the political tension between the Soviets and their former allies was escalating. By 1948 the Soviets cut off all ground travel into and out of Berlin essentially isolated it from the rest of the world. Airlift was the only way to supply West Berlin and its people. Berlin became a symbol of the United States resolve to stand up to the Soviet threat of expansion without being forced into a direct conflict.

The Mission: The official U.S. mission directive from the commanding general, United States Air Forces Europe (USAFE), to the project commander of the USAFE Berlin Airlift Operation was to: "Insure that the maximum number of missions are flown and that optimum overall efficiency of the operation is maintained."
Operation Vittles Concept Brief

- Your mission, should you decide to accept it, is to build the concept briefing for “Operation Vittles”.

- Audience:
  - Brigadier General Joseph Smith, Commander of the Wiesbaden Military Post, Task Force Commander, Operation Vittles

- Include:
  - Development Plan
  - Risks and Mitigation Plan
  - Organization and Team

- Your planning/briefing team consists of the team leader and subject matter experts to be identified
Berlin Airlift Case Study Deliverables

- Identify the project constraints
  - You might use a SWOT analysis here
  - What Subject Matter Experts do you need
- Identify Stakeholders (who leads, who benefits, who supports)
- Assign Roles within Organization
- Lifecycle Selection and Baseline Development
- Document team/project vision & purpose, goals, and values
- Identify the critical success factors & measures of success
- Develop the use cases and concept of operations
- Identify driving requirements
- Develop an architectural view
- Create your development plan/strategy
- Identify risks and mitigation plans
- Provide an answer to the General!
Identify Project Constraints

- Strengths?
- Opportunity?
- Weakness?
- Threats?

What Subject Matter Expertise do you need?
Organizational Behaviors

- Organizing for SE
- Baseline Development and SE Effort
- Berlin Airlift Application
Organizational Roles

Organizational Level

Executive Management Team
Support Services
Project Team

Management Team
Business Unit 1
Process Team
Project Team
Project Team

Management Team
Business Unit 2
Process Team
Project Team
Project Team

Everything serves the Business Unit

Project Level
Understanding Organizations - Valuable Read #1

- Fundamental Concepts of Centralized and Decentralized Organizations
- Emerging Culture of Decentralization, Empowered by Internet
Centralized Versus Decentralized

◆ **Centralization**
  - There’s someone in charge
  - There are headquarters
  - **If you thump it on the head, it dies**
  - There’s a clear division of roles
  - **If you take out a unit, the organization is harmed**
  - Knowledge and power are concentrated
  - The organization is rigid
  - Units are funded by the organization
  - You can count the participants
  - Working groups communicate through intermediaries

◆ **Decentralization**
  - There’s no one in charge
  - There are no headquarters
  - **If you thump it on the head, it survives**
  - There’s an amorphous division of roles
  - **If you take out a unit, the organization is unharmed**
  - Knowledge and power are distributed
  - The organization is flexible
  - Units are self-funded
  - You cannot count the participants
  - Working groups communicate with each other directly
Hierarchical versus Team Structures

Hierarchical Organizations
- Group People with Similar Tasks and Skills
- Clearly Define Employee Roles
- Promote Shared Knowledge & Efficiency Across the Skill Set
- Have a Well-Defined Management Hierarchy
- Assign Accountability to Unit Managers – Who Primarily Direct the Activities of the Unit
- Formulate Business Strategy at the Top of the Organization, Control the Strategy in the Middle
- See Innovation & Improvement Primarily Within the Functions
- Promote Career Growth Upward Within a Function
- Train People in Functional Skills

Team-Based Organizations
- Group People with Skills Required by the Project
- Focus all Employees on the Project
- Promote Shared Accountability for the Project
- Move Management into the Team – Requires Broader Business & Management Skills
- Assign Accountability to Project Managers – Who Primarily Create an Environment for Project Success
- Encourage Shared Ownership in Business Strategy
- See Innovation and Improvement via Diversity of Perspective and Opinion
- Promote Career via Expertise in Broad Skill Sets
- Cross-train
Organizational Factors to Team Success

- Organizational Support
  - Visible management support to the team structure
  - Employee processes for “managing the matrix”

- Process Focus
  - Employees must adopt team processes - can’t just organize into teams

- Clear Role Definitions
  - Purpose of the team
  - Responsibilities of the team

- Continuous Learning
  - Employees learn and develop broad skills
Systems Engineering is an Integrating Function

Project Management Team
- Project Manager
- Systems Manager
- Chief Engineer
- Production Manager
  etc.

Systems Support Team
- Quality Engineer
- Reliability Engineer
- Safety Engineer
  etc.

Systems Engineering & Integration Team Lead
- Systems Engineers

Subsystem 1 Team Lead
- Electronics Engineers
- SW Engineers
- Systems Engineer

Subsystem 2 Team Lead
- Electronics Engineers
- Mechanical Engineers
- SW Engineers
- Systems Engineer

Subsystem 3 Team Lead
- SW Engineers
- Systems Engineer
Summary

- Strong organizational systems engineering discipline is critical for today’s complexity
- The systems engineer has a critical role
  - Demonstrate leadership and team skills
  - Critical thinking tools for requirements/design trades and for understanding complexity
Baseline Development and Management

- The main point of Baseline Management is to establish a starting point and implement procedures to **Control Changes**!
Simple Life Cycle Baseline Development

Tailoring of the life cycle reviews and control gates depends on program size, complexity and scope.
So how do we develop these baseline then??
- Via SE processes
- Via Life Cycle selection
- Via SE tools
Mapping DAU to INCOSE Processes

Technical Processes

Requirements Analysis → System Analysis & Control (Balance) → Design Synthesis → Functional Analysis & Allocation

Technical Mgt Processes

- Project Planning
- Project Assessment and Control
- Decision Management
- Risk Management
- Configuration Management
- Information Management
- Measurement
SE through the Life Cycle and Baseline Development

System Engineering - Decomposition and Definition
## Baseline Levels of SE Effort

<table>
<thead>
<tr>
<th>Technical Processes</th>
<th>Concept BL</th>
<th>System BL</th>
<th>Functional BL</th>
<th>Design-To BL</th>
<th>Build-To BL</th>
<th>As-Built BL</th>
<th>As-Deployed BL</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Technical Management Processes</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 Project Planning</td>
<td>2</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2 Project Assessment and Control</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2a. Requirements Management</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>2b. Interface Management</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2c. Technical Assessment</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>3 (Technical) Risk Management</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>4 Configuration Mangement</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>5</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>5 (Technical) Data Management</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>6 Decision Analysis</td>
<td>2</td>
<td>5</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>7 Measurement</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>2</td>
</tr>
</tbody>
</table>

**Notes:**
- AOA/CR: Analysis of Alternatives & Concept Review
- SRR: Systems Requirements Review
- SDR: System Definition Review
- PDR: Preliminary Design Review
- CDR: Critical Design Review
- SAR: System Acceptance Review
- ORR: Ops Readiness Review

*Modified from: Applied Space Systems Engineering*
Tools & Methods Enable the SE Process

**Quality Function Deployment**

- Requirements Analysis
- Functional Analysis & Allocation
- System Analysis & Control (Balance)
- Verification
- Design Loop
- Design Synthesis
- Morphological Analysis
- Modeling & Simulation Tools

- Qualitative Selection Methods (Pugh)
- Quantitative Selection Methods (Multi-Attribute Decision Making)
- Robust Design
Exercise: Berlin Airlift Application

Organization of Operational Units, Stakeholders and Roles, Lifecycle and Baseline Development
Now that it has become clear that the airlift will continue for significantly longer than the original 3 weeks, Lt. General William Tunner of the Military Air Transport Service (MATS) will take over operations. General Tunner has significant experience in commanding and organizing the airlift over The Hump. Among other measures, he institutes 3 rules; Instrument Flight Rules will be in effect at all times, regardless of actual visibility; each sortie will have only one chance to land in Berlin, returning to its base if it missed its chance; aircrew can not leave their aircraft for any reason while in Berlin. He is working to improve living conditions for the aircrews and ground crews. He is recruiting former Luftwaffe aircraft mechanics to help with maintenance and established a school at Malmstrom AFB to train pilots in procedures specific to the airlift. All C-47s are replaced with the more capable C-54s.
Berlin Airlift Case Study Deliverables

- Identify the project constraints
  - You might use a SWOT analysis here
  - What Subject Matter Experts do you need
- Identify Stakeholders (who leads, who benefits, who supports)
- Assign Roles within Organization
- Lifecycle Selection and Baseline Development
- Document team/project vision & purpose, goals, and values
- Identify the critical success factors & measures of success
- Develop the use cases and concept of operations
- Identify driving requirements
- Develop an architectural view
- Create your development plan/strategy
- Identify risks and mitigation plans
- Provide an answer to the General!
Stakeholders

- Who Leads?
- Who Benefits?
- Who Supports?
Operational Units

- Airlift
- Airfield Operations
- Logistics and Cargo
- Maintenance and Servicing
Organization & Lifecycle

◆ Organization
  – Centralizes vs Decentralized?
  – Hierarchy vs Team Based?
  – What are the “business” units?

◆ Lifecycle
  – Baseline development?
  – Development lifecycle?
Team Organization

- Roles?
  - XXX

- Organization?
  - XXX
Leadership and Decision Making

- Leadership Concepts
- Decision Support Tools
- Berlin Airlift Application
Leadership Roles

Organizational Level

Executive Management Team

Management Team

Support Services

Management Team

Business Unit 1

Process Team

Project Team

Business Unit 2

Process Team

Project Team

Everything serves the Business Unit

Management Team

Management Team

Integrating Team

Integrating Team

Team Manager

Team Manager

Technical Leads

Technical Leads

Team Lead

Project Level
NASA found the behaviors of highly effective system engineers were very consistent:

1. **Leadership**
2. **Attitudes and attributes**
3. **Communication**
4. **Problem solving & systems thinking**
5. **Technical acumen**

1. Leadership Competencies

NASA Systems Engineering Behavior Competency Model*

- Appreciates/Recognizes Others
- Builds Team Cohesion
- Understands the Human Dynamics of a Team
- Creates Vision and Direction
- Ensures System Integrity
- Possesses Influencing Skills
- Sees Situations Objectively
- Coaches and Mentors
- Delegates
- Ensures Resources are Available

2. Attitudes & Attributes

NASA Systems Engineering Behavior Competency Model*

- Remains Inquisitive and Curious
- Seeks Information and Uses the Art of Questioning
- **Advances Ideas**
- **Gains Respect Credibility, and Trust**
- **Possesses Self-Confidence**
- Has a **Comprehensive View**
- Positive Attitude; Dedication to Mission Success
- Aware of Personal Limitations
- Adapts to Change and Uncertainty
- Uses Intuition/ Sensing
- Able to Deal with **Politics, Financial Issues, Customer Needs**

3. Communication

NASA Systems Engineering Behavior Competency Model*

- Listens effectively and translations information
  - Excellent listener (listens for recurring themes)
  - A translator; Often clarifies & summarizes

- Communicates through personal Interaction
  - Daily, hourly interaction
  - Face to face, rather than email
  - Facilitates personal interactions of the team

- Facilitates environment of open & honest communication
  - Creates atmosphere of freedom to express opinions
  - Everyone gets heard
  - Demonstrates approachability

## Team Leadership Spectrum

<table>
<thead>
<tr>
<th>Days/Weeks</th>
<th>Months</th>
<th>1 Year</th>
<th>3-5 Years</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>P</strong>rotect the Team</td>
<td><strong>P</strong>rogress</td>
<td><strong>P</strong>ursue Goals</td>
<td><strong>P</strong>rovide the Vision</td>
</tr>
<tr>
<td>• “Problem Solver”</td>
<td>• “Manager”</td>
<td>• “Ruler”</td>
<td>• “Leader”</td>
</tr>
<tr>
<td>• Roles: Solve problems &amp; remove obstacles</td>
<td>• Roles: Provide information, &amp; track performance</td>
<td>• Roles: set boundaries &amp; norms of behavior</td>
<td>• Grow and enable the team purpose and shared vision</td>
</tr>
</tbody>
</table>

**Governance is at the center; Leadership is at either end.**
Purpose, Mission, Vision

Creating and documenting these provides the team with a shared view of its future and reason to get there.
Concepts Applied to Leadership & Organization - Senge’s Five Disciplines

- **Systems Thinking**
  - The understanding of complex systems, the ability to see patterns in complexity, and the tools to support such understanding.

- **Personal Mastery**
  - "continually clarifying what is important to us, and continually learning to see current reality more clearly"

- **Mental Modeling**
  - “the art of reflection and inquiry, leading to models that influence how we understand the organization and how we take action”

- **Building Shared Vision**
  - “hold a shared picture of the future we seek to create"

- **Team Learning**
  - “teams, not individuals, are the fundamental learning unit in modern organizations"

Concepts Applied to Decision Making

- Understanding Causes, Effects, Symptoms
  - Collaborative, multiple perspectives
  - Experimental
  - Open
  - Contextual
- Aligned with greater vision
- Development and follow through
Decision Making is a Collaborative Process

- Successful goals and objectives are achieved through decisions that:
  - Are data based
  - Manage expectations
  - Capitalize on the creativity, skills and resources available
  - Build and maintain relationships

The challenge of the Systems Engineer is to present the trade space in a form that is both understandable to high level decision makers and that contains an actionable set of data.
Note how orientation shapes observation, shapes decision, shapes action, and in turn is shaped by the feedback and other phenomena coming into our sensing or observing window.

Also note how the entire “loop” (not just orientation) is an ongoing many-sided implicit cross-referencing process of projection, empathy, correlation, and rejection.


### The Life Cycle of a Judgment Call

<table>
<thead>
<tr>
<th>Observe</th>
<th>Orient</th>
<th>Decide</th>
<th>Act</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Preparation Phase</strong></td>
<td><strong>Call Phase</strong></td>
<td><strong>Execution Phase</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Good Leader</strong></td>
<td><strong>Identifies important stakeholders</strong></td>
<td><strong>Makes a clear yes/no call</strong></td>
<td><strong>Stays involved during execution</strong></td>
</tr>
<tr>
<td>Picks up on signals in the environment</td>
<td></td>
<td>Makes a clear yes/no call</td>
<td>Stays involved during execution</td>
</tr>
<tr>
<td>Is energized about the future</td>
<td>Engages and energizes stakeholders</td>
<td>Thoroughly explains the call</td>
<td>Supports others who are involved</td>
</tr>
<tr>
<td></td>
<td>Taps best ideas from anywhere</td>
<td></td>
<td>Sets clear milestones</td>
</tr>
<tr>
<td><strong>Preparation Phase</strong></td>
<td><strong>Call Phase</strong></td>
<td><strong>Execution Phase</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Poor Leader</strong></td>
<td><strong>Does not set clear expectations</strong></td>
<td><strong>Dillydallies when it’s time to make a call</strong></td>
<td><strong>Walks away once the call is made</strong></td>
</tr>
<tr>
<td>Cannot read the environment</td>
<td></td>
<td>Does not understand how issues intersect and how the call will play out</td>
<td>Does not measure outcomes</td>
</tr>
<tr>
<td>Fails to acknowledge reality</td>
<td>Does not define the ultimate goal</td>
<td>Fails to understand important information</td>
<td>Does not respond to resistance in the organization</td>
</tr>
<tr>
<td>Does not follow gut instincts</td>
<td>Remains stuck in an old paradigm</td>
<td>Does not understand what good execution requires</td>
<td>Lacks operating mechanisms to make necessary changes</td>
</tr>
</tbody>
</table>

---

*Tichy, Noel M. and Bennis, Warren; *Judgment, How Winning Leaders Make Great Calls*; Portfolio Hardcover, Nov 2007*
The Message

- System engineering is increasingly difficult.
  - Increasingly complex systems
  - Increasingly more participants, stakeholders, & influences

- Leadership is fundamental for successful systems engineering.

- Leadership skills must be developed by practice.
SE Tools for Decision Making

- Quality Function Deployment (QFD)
- Use Cases
- Morphological Matrix of Alternatives
- Modeling and Simulation
- SWOT Analysis

- Multi-Attribute Decision Making (MADM)
  - Methods for handling multiple and conflicting objectives
    - Pugh, AHP, and TOPSIS common techniques
    - Introducing Design Difficulty vs Resources Analysis
The Need for Metrics

To evaluate the results of the solution generation phase, a set of metrics must be created to evaluate one alternative vs. another.

Typically called Measures of Effectiveness and/or Measures of Performance

The metrics should be directly associated with the specific objectives of the solutions.

Generally the metrics should be prioritized according to their operations effectiveness.

Two Metrics - Universal Metrics

- Design Difficulty – captures the feasibility of the design
- Required Resources – captures the viability of the design
- These two metrics can be used to assess the risk of project failure
- These metrics allow the engineer to evaluate any project on its location on the Design Difficulty vs. Resource plane
- “Metrics and Case Studies for Evaluating Engineering Designs” has 33 different design projects evaluated on the DD vs. R plane
The Design Difficulty vs. Resources Plane

- Star Wars
- Moon Landing
- Consumer Products
- Seven Wonders of the Ancient World
Design Difficulty Categories (Suggested)

1) Design type
2) Knowledge complexity
3) Number of process steps to create system
4) Desired quality level
5) Process complexity
6) Selling price goals

◆ Note – these are the suggested categories, additional categories can be added as necessary
## Design Difficulty Scoring

<table>
<thead>
<tr>
<th>Categories</th>
<th>Typical Scoring</th>
<th>Ordinal Scoring</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Design type</strong></td>
<td>14 or 15 points for a breakthrough design effort.</td>
<td>7 – 13 points for original innovative design</td>
</tr>
<tr>
<td></td>
<td>0 – 6 points for continuous improvement</td>
<td></td>
</tr>
<tr>
<td><strong>Knowledge complexity</strong></td>
<td>9 – 10 points for undiscovered knowledge found only by specialists.</td>
<td>6 – 8 points for complex knowledge held by a few people</td>
</tr>
<tr>
<td></td>
<td>3 – 5 points for complex knowledge held by a numerous people</td>
<td>0 – 2 points for common knowledge held by a many people</td>
</tr>
<tr>
<td><strong>Number of process steps to create system</strong></td>
<td>9 – 10 points for systems with more than 10,000 steps or components</td>
<td>5 – 8 points for systems with 500 but less than 10,000</td>
</tr>
<tr>
<td></td>
<td>3 – 4 points for systems with up to 500 steps or components</td>
<td>0 – 2 points for systems with less than 50 steps or components</td>
</tr>
<tr>
<td><strong>Desired quality level</strong></td>
<td>7 – 10 points for system whose developer places high emphasis on quality related programs / techniques</td>
<td>4 – 6 points for medium level of focus on quality related programs and techniques</td>
</tr>
<tr>
<td></td>
<td>0 – 3 points for developer that puts little to no emphasis on implementing or continuing quality related programs or techniques.</td>
<td></td>
</tr>
<tr>
<td><strong>Process complexity</strong></td>
<td>5 points for highly complex manufacturing processes for producing products to meet a large national market share.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4 points for high manufacturing complexity for moderate national market share or moderate manufacturing complexity for large national market share</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3 points for high manufacturing complexity and small market share, moderate manufacturing complexity and moderate share or low manufacturing complexity and large market share</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 points for moderate complexity and small market share or low complexity and moderate market share.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 point for low complexity to produce low quantities (greater than one)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 points for low complexity that only produce one system.</td>
<td></td>
</tr>
<tr>
<td><strong>Selling price goals</strong></td>
<td>4 – 5 points for very challenging unit price goals or high market competition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2 – 3 points for moderate unit price goals and or market competition</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 – 1 points for little or no unit price goals or market competition</td>
<td></td>
</tr>
</tbody>
</table>
Resources Categories (Suggested)

1. Cost
2. Time
3. Infrastructure

Note – these are the suggested categories, additional categories can be added as necessary
– E.g. – Manpower
# Resources Scoring

<table>
<thead>
<tr>
<th>Categories</th>
<th>Typical Scoring</th>
<th>Ordinal Scoring</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost</td>
<td>14 – 15 points for systems that require massive financial sacrifices</td>
<td>9 – 13 points for very expensive system that are rarely developed</td>
</tr>
<tr>
<td></td>
<td>9 – 13 points for very expensive system that are rarely developed</td>
<td>3 – 8 points for moderately expensive systems</td>
</tr>
<tr>
<td></td>
<td>3 – 8 points for moderately expensive systems</td>
<td>0 – 2 points for affordable systems</td>
</tr>
<tr>
<td>Time</td>
<td>10 points for projects requiring more than 8 years</td>
<td>8 – 9 points for projects lasting 5 to 8 years</td>
</tr>
<tr>
<td></td>
<td>8 – 9 points for projects lasting 5 to 8 years</td>
<td>4 – 7 points for projects lasting 1 to 5 years</td>
</tr>
<tr>
<td></td>
<td>4 – 7 points for projects lasting 1 to 5 years</td>
<td>3 points for a six month to one year effort</td>
</tr>
<tr>
<td></td>
<td>3 points for a six month to one year effort</td>
<td>2 points for a three to six month effort</td>
</tr>
<tr>
<td></td>
<td>2 points for a three to six month effort</td>
<td>1 point for one to three months</td>
</tr>
<tr>
<td></td>
<td>1 point for one to three months</td>
<td>0 points for less than one month</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>9 – 10 points for massive infrastructure requiring major portions of the available workforce and available equipment</td>
<td>6 – 8 points for large, complex infrastructures requiring large portions of the cost of entire project</td>
</tr>
<tr>
<td></td>
<td>6 – 8 points for large, complex infrastructures requiring large portions of the cost of entire project</td>
<td>3 – 5 points for moderate infrastructures requiring people on the project to support it.</td>
</tr>
<tr>
<td></td>
<td>3 – 5 points for moderate infrastructures requiring people on the project to support it.</td>
<td>0 – 2 points given for common, low cost infrastructure (e.g. clean tap water in the U.S.)</td>
</tr>
</tbody>
</table>

Copyright © Georgia Tech. All Rights Reserved.
Calculating DD-R Plane Scores

- Values for Design Difficulty and Resources are computed by summing scores for their individual parts.

- Each constituent part is an ordinal ranking within the category.

- Extreme examples may not fit the ranking methodology, scale as necessary to pass a reasonable test.
DD – R Plane for Case Studies

The diagram illustrates a design difficulty and risk matrix for various case studies. The x-axis represents resources on a scale from 0 to 35, and the y-axis represents design difficulty on a scale from 0 to 50. The cases are plotted according to their design difficulty and risk levels. For example:

- **High Technological Risk**: Projects like **C3PO**, **Breakthrough Battery**, **Electric Vehicle**, **Manhattan Project**, **777**, **Polaris**, **Apollo**, **Supercollider**, **Hubble Telescope**, **Pyramid**.
Exercise: Berlin Airlift Application

Team/Project Vision, Purpose, Goal
Critical Success Factors and Measures of Success
Berlin Airlift Case Study Deliverables

- Identify the project constraints
  - You might use a SWOT analysis here
  - What Subject Matter Experts do you need
- Identify Stakeholders (who leads, who benefits, who supports)
- Assign Roles within Organization
- Lifecycle Selection and Baseline Development

- Document team/project vision & purpose, goals, and values
- Identify the critical success factors & measures of success

- Develop the use cases and concept of operations
- Identify driving requirements
- Develop an architectural view
- Create your development plan/strategy
- Identify risks and mitigation plans
- Provide an answer to the General!
Berlin Airlift

- **Vision?**:
  - XXX

- **Purpose?**:
  - XXX

- **Goal?**:
  - XXX

- **Values?**:
  - XXX
Measures of Success

- Critical Success Factors?
  - XXX
  - XXX
  - XXX

- Measures of Success?
  - XXX
  - XXX
  - XXX

- Design Difficulty vs Resources Evaluation?
Part 1 Summary and Break

- Why SE Case Studies
- Berlin Airlift Case Study – Experiential Learning and Systems Thinking
  - Applied systems thinking
  - Organizational Behaviors
  - Leadership and Decision Making
- Deliverables:
  - Project constraints
  - Stakeholders
  - Roles within Organization
  - Lifecycle Selection and Baseline Development
  - Team/project vision & purpose, goals, and values
  - Critical success factors & measures of success
Part 2 Overview

- Requirements and System Architecting
  - Use Cases
  - Logic Models
  - Concept of Operations
  - Berlin Airlift Application

- Project Management for Complex Systems
  - Project Planning
  - Risk Management
  - Berlin Airlift Application

- Brief the General!
Requirements and System Architecting

– Use Cases
– Logic Models
– Concept of Operations
– System Architecting
– Berlin Airlift Application
What is a Use Case?

A Use Case is
- a **set of scenarios** that describe the behavior (or desired behavior) of a system and its users
- at a superficial level of detail
- with “sunny-day” and “rainy-day” scenarios
- with some generalization of the roles and activities
- a set of activities within a system

A Use Case is
- the **set of scenarios** that provides positive value to one or more external actors
  - actors are the people and/or computer systems that are outside the system under development
  - scenarios are dialogs between actors and the system
  - no information about the internal design
### Use Case Fundamentals

**Step 1: Create a list of Actors**

Customer  
Support hotline person  
Administrator  
Repair person

**Step 2: Create a list of Goals**

**Web-based music distribution system:**
- UC1: Customer downloads a song
- UC2: Customer searches music directory
- UC3: Administrator adds a new user
- UC4: Administrator updates directory
- UC5: Support hotline person investigates a Customer problem
- UC6: Support hotline person authorizes Customer refund
- UC7: Repair person runs diagnostics

**Step 3: Write simple use cases with only sunny-day scenarios**

**UC1: Customer downloads a song**  
**Precondition:** Song file is on a server  
**Main scenario:**
1. Customer chooses song  
2. System checks availability and price; prompts Customer for payment  
3. Customer enters credit card info  
4. System sends credit card transaction to Bank  
5. Bank returns transaction number  
6. System transmits the song to Customer’s computer

**Step 4: Review the use cases with customer (or customer surrogate)**
Step 5: Identify failure conditions

2a. Song is not available
3a. Customer quits without entering credit card info
4a. Link to Bank is down
5a. Credit card is rejected by Bank
6a. Server fails during transmission
6b. Customer cancels during transmission

Step 6: Write a selected set of failure scenarios and alternatives

5a. Credit card is rejected by Bank:
   5a1. System reports failure to the Customer, prompts Customer for a different credit card
   5a2. Customer enters card info
   5a3. go to step 4

Step 7: Internal review

- Review the scenarios and failure branches with testers, developers, project managers

Ongoing: make links to other requirements, update use case model as needed

- Define the business rules and non-functional requirements (in text documents, with links to the use case model)
- Add new use cases and new scenarios for new actors and goals; new variations for existing use cases
Logic Models

- Roots of program evaluation theory and methods can be traced to industrial psychology and “scientific” management methods from the 1920’s and 1930’s.
  - Concept of intervention to address a problem
  - Hawthorne effect
- Logic Models identify interventions and intermediate, measurable outcomes to achieve long-term goals
Specifying the Logic Model

- Identify the desired long-term outcomes
- Identify the constructs involved in the model
  - Latent variables (cannot be directly observed)
  - Manifest variables (can be observed or measured)
- Specify the causal relationships among the constructs
  - Direct and indirect causes
- Specify factors that influence the causal relationships
  - Moderating and mediating variables
Desired Effect and Interventions

You might have to act on other causes (e.g. reduce barriers) in order to achieve the desired effect.
Start at the End

- Logic models must address what outcomes (effects) are desired.
- The desired outcomes are usually affected by factors beyond the interventions introduced by the program.
- If you don’t know where you want to go, you’ll never know when you get there!
Jump to the Beginning

- Describe the current situation
  - What factors contribute to the effect of interest?
  - What factors interfere with the effect of interest?
- Identify needs / gaps where there is opportunity to influence the effect
- Consider strengths, weaknesses, opportunities, threats (SWOT analysis)
Fill in the Middle

- Given the desired effect, specify the interventions (program actions) that will be performed, and the rational for how those interventions will influence the desired effect.

- The interventions can directly produce the desired effect, or can indirectly produce the effect by acting on other causes of the effect.
Create Concept of Operations

- Create, visualize and discuss use scenarios in complex environments; Used as a strategic planning tool to reduce chance of overlooking important factors; provides balanced perspective
- Explore scenarios for clear understanding of operational needs and performance requirement rationale
Concept of Operations (CONOPS)

- A user oriented document that describes system characteristics of the to-be-delivered system from the user’s viewpoint.

- Used to communicate overall quantitative and qualitative system characteristics to the user, buyer, developer, and other organizational elements (e.g., training, facilities, staffing, and maintenance).

- Describes the user organization(s), mission(s), and organizational objectives from an integrated systems point of view.

Source: IEEE Std 1362-1998 (R2007)
The Role of the System Architect

- The System Architect is more a leadership and management role than a technical role.
- Architects need experience, and a blend of management and leadership disciplines.
- Communication and vision require leadership capacity.
  - The architect holds the architectural vision, often their own.
  - The architect makes high-level design decisions around interfaces, functional partitioning, and interactions.
  - The architect must communicate these effectively, often visually.
- The architect’s primary tasks are rule-setting.
  - The architect must direct technical standards, including design standards, tools, or platforms.
  - These should be based on business goals rather than to place arbitrary restrictions on the choices of developers.
Leadership Competencies

◆ Experience and judgment
  – The architect must balance the customer’s view of the system with their organization’s business view of the system

◆ Communications
  – The architecture is presented in visuals to all stakeholders
  – The architecture is used to derive written guidelines and design rules for the team

◆ Leadership and Systems Thinking
  – The architecture is the high-level vision of the system
  – The architecture is defined more by heuristics than requirements
  – The architecture definition contains a number of soft requirements that have to be evaluated in collaborative groups

◆ Management
  – The architect ensures the design team follows design standards
Architecture Summary

- Develop use cases with potential or targeted customers
- Develop Architectural Views
- Develop the functional architecture: allocation of functions within the higher level architectural goals
- With the customer and team, define the quality requirements
- Select or create design guidance for the team
- This is the earliest part of requirements development, and the requirements document captures the result of this process in order to inform the derived requirements
Techniques for Architecture and Design

- Use cases and usage scenarios, functional requirements, non-functional requirements, technological requirements, the target deployment environment, and other constraints produce:

- A list of Architecturally Significant Use Cases

- These feed a scenario-based evaluation process
Exercise: Berlin Airlift Application

Use Cases

Concept of Operations

Architecture Views
Berlin Airlift Case Study Deliverables

- Identify the project constraints
  - You might use a SWOT analysis here
  - What Subject Matter Experts do you need
- Identify Stakeholders (who leads, who benefits, who supports)
- Assign Roles within Organization
- Lifecycle Selection and Baseline Development
- Document team/project vision & purpose, goals, and values
- Identify the critical success factors & measures of success
- Develop the use cases and concept of operations
- Identify driving requirements
- Develop an architectural view
- Create your development plan/strategy
- Identify risks and mitigation plans
- Provide an answer to the General!
Example Berlin Airlift Use Case

“System” encompasses:
- Aircraft
- Cargo
- Airfields
- Service

Cargo Delivery Scenarios:
- Deliver Cargo to Airfield
- Distribute Cargo to Berlin Citizens
- Return to Base w/out Cargo Delivery

Actors:
- Pilots
- Berlin Citizens
- Soviets
- Wx
Concept of Operations

- ??
- ??
Driving Requirements

- XXX
- XXX

- Can we meet these requirements?
Architecture Views

- Airfield Operations?
- Logistics and Cargo?
- Maintenance and Servicing?
- Overall Mission Architecture?
Project Management for Complex Systems

- Project Planning
- Risk Management
- Berlin Airlift Application
Risk, Uncertainty, and Opportunity

Risk = KNOWN-UNKNOWN

Uncertainty = UNKNOWN-UNKNOWN

As we know,
There are known knowns.
There are things we know we know.
We also know
There are known unknowns.
That is to say
We know there are some things
We do not know.
But there are also unknown unknowns,
The ones we don't know
We don't know.

—Feb. 12, 2002, Department of Defense news briefing

The Unknown
by Donald Rumsfeld

As we know,
There are known knowns.
There are things we know we know.
We also know
There are known unknowns.
That is to say
We know there are some things
We do not know.
But there are also unknown unknowns,
The ones we don't know
We don't know.

—Feb. 12, 2002, Department of Defense news briefing
Project Plan: The Iron Triangle

Performance

Novelty

Environment

Cost

Schedule

Pace

Risks

Technology

Complexity
A New Reference

- Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation, by Aaron J. Shenhar and Dov Dvir
  - A model for evaluating your project management approach versus project complexity
  - Useful guidance to evaluating the project management disciplines selected versus 4 dimensions of complexity
Basic Product Development

Investigate

Need
Market
Idea
Technology

Determine Feasibility

Concept
Dev. Plan
Risks
Business Case

Develop

Design
IPPD
Mfg. Plan.
Schedule
Cost

Pilot/Test

Int’n
Test Plan
Test Market
V & V

Launch

Eng’g
Support
Sales
Plan
Quality

Deploy

Support
Plan
Training

Retire

Disposal
Plan
Next Product

User Requirements

Product Requirements

Product Design

Product

Product Support

Product Replacement
The Fuzzy Front End

- Initial drivers to classify a project:
  - The need or idea: who, what, why, when?
  - The business goal: what is the exact outcome or product? What are the business drivers?
  - The market/customers: what is the exact work that needs to be done? What is the complexity?
  - The environment: what are the other factors driving the project? Business, market, technology, industry, economics, policies, organization, people skills, process?
The Fuzzy Front End

- Who needs it?
- Why do they need it?
- When do they need it?
- How will they use it?
- What will they use if they don’t have it?
- How many would use it?
- What might they pay for it?

- What best meets the need?
- How easy is it to use?
- When will it be delivered?
- How will it be made, delivered, supported?
- Who will provide it?

- Does it fit current architecture?
- Does it meet timeline?
- Is risk manageable?
- What is the expected return?

Novelty  Technology  Complexity  Pace
Managing Uncertainty

- Traditional project management discipline is based on relatively predictable models.

- As project complexity increases, project management becomes more about managing uncertainty:
  - Market uncertainty: the novelty of approach leading to uncertainty in requirements
  - Technology uncertainty: maturity of technology leading to uncertainty in design
  - Complexity: system is difficult to understand or predict, unpredictable behaviors in market or project teams
  - Pace: decisions and behaviors must be adapted to meet hard deadlines

Aaron J. Shenhar and Dov Dvir, Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation
The Diamond or NTCP Model

Complexity

Technology

Super-high-tech

High-tech

Medium-tech

Low-tech

Novelty

Regular

Array

System

Assembly

Fast/competitive

Derivative

Platform

Breakthrough

Time-critical

Blitz

Pace

Aaron J. Shenhar and Dov Dvir, Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation
The Impact of NTCP Dimensions on Project Management

Complexity

Technology

Novelty

Complex organization

Formality

Later design freeze

More design cycles

Less market data

Later requirements freeze

Autonomy

Pace

Aaron J. Shenhar and Dov Dvir,
Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation
Project Management Tools

◆ Planning a complex project
  1. Identify the business objectives and customer needs
  2. Simplify objectives, allow structure to be defined; determine system and project organizational architecture
  3. Develop work breakdown and high level scheduling, then details of work teams and tasks
  4. Analyze the complexity of the resultant project, adapt planning to suit: The diamond or NTCP model
  5. Select project management approach; determine evolutionary development framework

◆ Managing a complex project
  – Use agile development techniques
  – Develop team-based learning
  – Monitor based on risk
### Business Objectives and Models

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Consumer</th>
<th>Enterprise</th>
<th>Public</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Objective</td>
<td>Volume, Market Share</td>
<td>Long-term Provider</td>
<td>Long-term Relationship</td>
</tr>
<tr>
<td>Project Focus</td>
<td>Cost, Quality, Novelty</td>
<td>Cost, Service</td>
<td>Performance, Service</td>
</tr>
<tr>
<td>Project Pace</td>
<td>Time to market</td>
<td>Time to delivery</td>
<td>Focus on long-term</td>
</tr>
<tr>
<td>Product</td>
<td>Defined by marketing</td>
<td>Defined w/customer involvement</td>
<td>Defined by customer</td>
</tr>
<tr>
<td>Project Plan</td>
<td>Defined by producer</td>
<td>Defined by producer with customer</td>
<td>Defined by or with customer</td>
</tr>
<tr>
<td>Contract</td>
<td>No contract</td>
<td>Either</td>
<td>Contracted</td>
</tr>
<tr>
<td>Reviews/Milestones</td>
<td>Internal</td>
<td>Internal/external</td>
<td>Customer driven</td>
</tr>
<tr>
<td>Production Readiness</td>
<td>Mass production</td>
<td>Tailored to customer</td>
<td>Limited quantity</td>
</tr>
</tbody>
</table>

Aaron J. Shenhar and Dov Dvir, Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation
Identify your project type

- **Novelty**
  - Derivative, platform, breakthrough

- **Technology**
  - Low, medium, high, super-high tech

- **Complexity**
  - Assembly, system, array

- **Pace**
  - Regular, fast/competitive, time-critical, blitz

- **Other**
  - Strategic (might take more risk)
  - Internally or externally driven
Define Where you fit on the NTCP Model

Aaron J. Shenhar and Dov Dvir, Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation
Project Planning

- WBS
  - Project Tasks
  - Project Organization
- Communication (how you will track it)
- Development process
  - Major phases, gates, milestones and what will happen at each
- Define 3-5 relevant success criteria, and what can go wrong with each
## Project Uncertainty and its Impact

<table>
<thead>
<tr>
<th>Uncertainty level</th>
<th>Quantitative Level</th>
<th>Novelty</th>
<th>Technology</th>
<th>Number of Iterations</th>
<th>Number of Prototypes</th>
<th>Time &amp; Budget Reserves</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>1</td>
<td>Derivative</td>
<td>Low</td>
<td>Few (1-2)</td>
<td>None</td>
<td>5%</td>
</tr>
<tr>
<td>Medium</td>
<td>2</td>
<td>Platform</td>
<td>Medium</td>
<td>Several (2-3)</td>
<td>Few (1-2)</td>
<td>5-10%</td>
</tr>
<tr>
<td>High</td>
<td>3</td>
<td>Breakthrough</td>
<td>High</td>
<td>Many (3-4)</td>
<td>Many (3-4)</td>
<td>10-25%</td>
</tr>
<tr>
<td>Super-high</td>
<td>4</td>
<td>Super-high</td>
<td>Multiple*</td>
<td>Multiple*</td>
<td>Multiple*</td>
<td>25-50%</td>
</tr>
</tbody>
</table>

*Multiple = multiple cycles with multiple prototypes each
2 Dimensions of Work Package Mgmt.

- **Type of outcome, type of work**
  - Tangible outcomes: physical artifacts
  - Intangible outcomes: information, including SW (not manufactured)

- **Type of work**
  - Inventive: result of creative input, exploratory in nature
  - Engineering: science & engineering to produce outcomes
  - Craft: repetitive tasks around work that has been done before

- **These drive how you define your scheduling model and approach**
Use Agile Project Planning

- The project plan seldom sticks to its original
- Plan your work, work, and replan
- Planning detail at the point in the high level plan you are sitting on today and 3 months further (rolling waves)

Laufer, Alex; “Simultaneous Management;” 3 hierarchical plans instead of 1 integrated plan:
  - Highest level – looks over the entire project life
    » Major milestones identified
  - Middle level – 4-6 months, medium level or focused events
  - Detailed work plan – 1-2 months
Managing Uncertainty

- Uncertainty level of the project is the maximal between Novelty and Technology
- Risk and Uncertainty are not always related
  - Known Unknowns versus Unknown Unknowns
- Use the diamond model for risk
- Evaluate risk types for your project: Novelty, Technology, Complexity, Pace. Where does your project sit? Address risk consequence based on maximal points in NTCP model.
The Relationship Between the NTCP Model and Project Risk

Project risk = a(N) + b(T) + c(C) + d(P)

Aaron J. Shenhar and Dov Dvir, Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation
Exercise: Berlin Airlift Application

Development Strategy, Risks and Mitigation
Berlin Airlift Case Study Deliverables

- Identify the project constraints
  - You might use a SWOT analysis here
  - What Subject Matter Experts do you need
- Identify Stakeholders (who leads, who benefits, who supports)
- Assign Roles within Organization
- Lifecycle Selection and Baseline Development
- Document team/project vision & purpose, goals, and values
- Identify the critical success factors & measures of success
- Develop the use cases and concept of operations
- Identify driving requirements
- Develop an architectural view
- Create your development plan/strategy
- Identify risks and mitigation plans
- Provide an answer to the General!
Development Strategy

- **Highest level** (looks over the entire project life)
  - Major milestones identified
  - Milestone 1 XXX
  - Milestone 2 XXX

- **Middle level** (4-6 months - medium level or focused events)
  - XXX

- **Detailed work plan** (1-2 months)
  - XXX
Define Where you fit on the NTCP Model

Technology

Super-high-tech
High-tech
Medium-tech
Low-tech

Complexity
Array
System
Assembly

Pace
Fast/competitive
Time-critical
Blitz

Novelty
Regular
Derivative
Platform
Breakthrough

Aaron J. Shenhar and Dov Dvir, Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation.
The Relationship Between the NTCP Model and Project Risk

Project risk = a(N) + b(T) + c(C) + d(P)

Aaron J. Shenhar and Dov Dvir, Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation
Risks and Mitigations

- **Risk**
  - XXX
  - XXX

- **Mitigation**
  - XXX
  - XXX
Summary and Conclusions

Bring all deliverables together for the Concept Briefing
Discussion

- Did you find yourself approaching this “project” differently than you would have before this seminar?
  - If so – how? If not – why?
- Did you recognize the “systems” aspect of this study?
  - What aspects of the seminar helped you the most when dealing with this large, complex system of systems challenge?
- What additional “resources” did you need at the front end of this planning exercise?
- What “team based” organizational issues did you have to address? Centralized vs Decentralized?
- How did you identify the risks?
- How about requirements? Biggest driver?
- Will your lifecycle help manage risk? Anything else?
- How did you handle incomplete data?
- Other Techniques? Mindmapping? QFD? Functional Decomp?
- What about your planning team?
  - Did it work? Why or why not? Forest or trees?
Conclusions and Summary

- Systems engineering (SE) case studies:
  - Extension of traditional engineering case studies
  - Expose students to open ended problems
  - Enable Experiential Learning
  - Foster Systems Thinking
  - Focus on Applied Systems Engineering

- Air Force Institute of Technology (AFIT) Cases:
  - Wealth of resources
  - Extend to other exercises & SE labs

- The Berlin Airlift:
  - Experience Learning by Doing
  - Exercise Team Building & Leadership
  - Develop a “System” Architecture
  - Exercise your Systems Thinking
References

References and Recommended Reading

## N2 on planning

<table>
<thead>
<tr>
<th></th>
<th>Tangible</th>
<th>Intangible</th>
<th>Inventive</th>
<th>Engineering</th>
<th>Craft</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Tangible</strong></td>
<td></td>
<td>Risk of forcing all development down same path</td>
<td>High risk of customer dissatisfaction</td>
<td>High risk of technology maturity issues</td>
<td>Risk of being late to market</td>
</tr>
<tr>
<td><strong>Intangible</strong></td>
<td>Use multiple development models</td>
<td></td>
<td>High risk of customer dissatisfaction</td>
<td>High risk of utility or use case issues</td>
<td>Generally low risk unless innovation is a premium</td>
</tr>
<tr>
<td><strong>Inventive</strong></td>
<td>Build several prototypes and test with customers</td>
<td>Case for incremental development with frequent customer interaction</td>
<td></td>
<td>Risk of immature requirements leading to poor use case design</td>
<td>Risk of disruptive design or process issues</td>
</tr>
<tr>
<td><strong>Engineering</strong></td>
<td>Evolutionary development approach with several fielded increments</td>
<td>Early increments focus on system use cases and utility</td>
<td>Use M&amp;S to focus customer on use cases and utility</td>
<td></td>
<td>Risk of cost or quality issues</td>
</tr>
<tr>
<td><strong>Craft</strong></td>
<td>Waterfall approach or evolutions focused on improved cost &amp; quality</td>
<td>Accelerate fielded systems to evaluate utility and maturity</td>
<td>Early prototypes to mature processes</td>
<td>Early prototypes to prove technology</td>
<td></td>
</tr>
</tbody>
</table>
AFIT Case Studies

- Hubble
- A-10
- GPS
- TBMCS
- ISS
- Global Hawk

http://www.afit.edu/cse/cases.cfm
The Tuckman Model recognizes that there is a process to building relationships between team members.

**Tuckman Model of Team Behavior**

**Forming**
- Making contact and bonding
- Developing trust
- Members dependent

**Storming**
- Expressing differences of ideas, feelings, and opinions
- Reacting to leadership
- Members independent or counterdependent

**Norming**
- Decisions are made through negotiation and consensus building
- Identifying power and control issues
- Gaining skills in communication
- Identifying resources

**Performing**
- Members work collaboratively
- Members care about each other
- The group establishes a unique identity
- Members are interdependent

**Behaviors**