Project Manager Notes

Notes to assist a project manager

Richard Chambers

Version: .6 Jun 15, 2003

Things to Remember

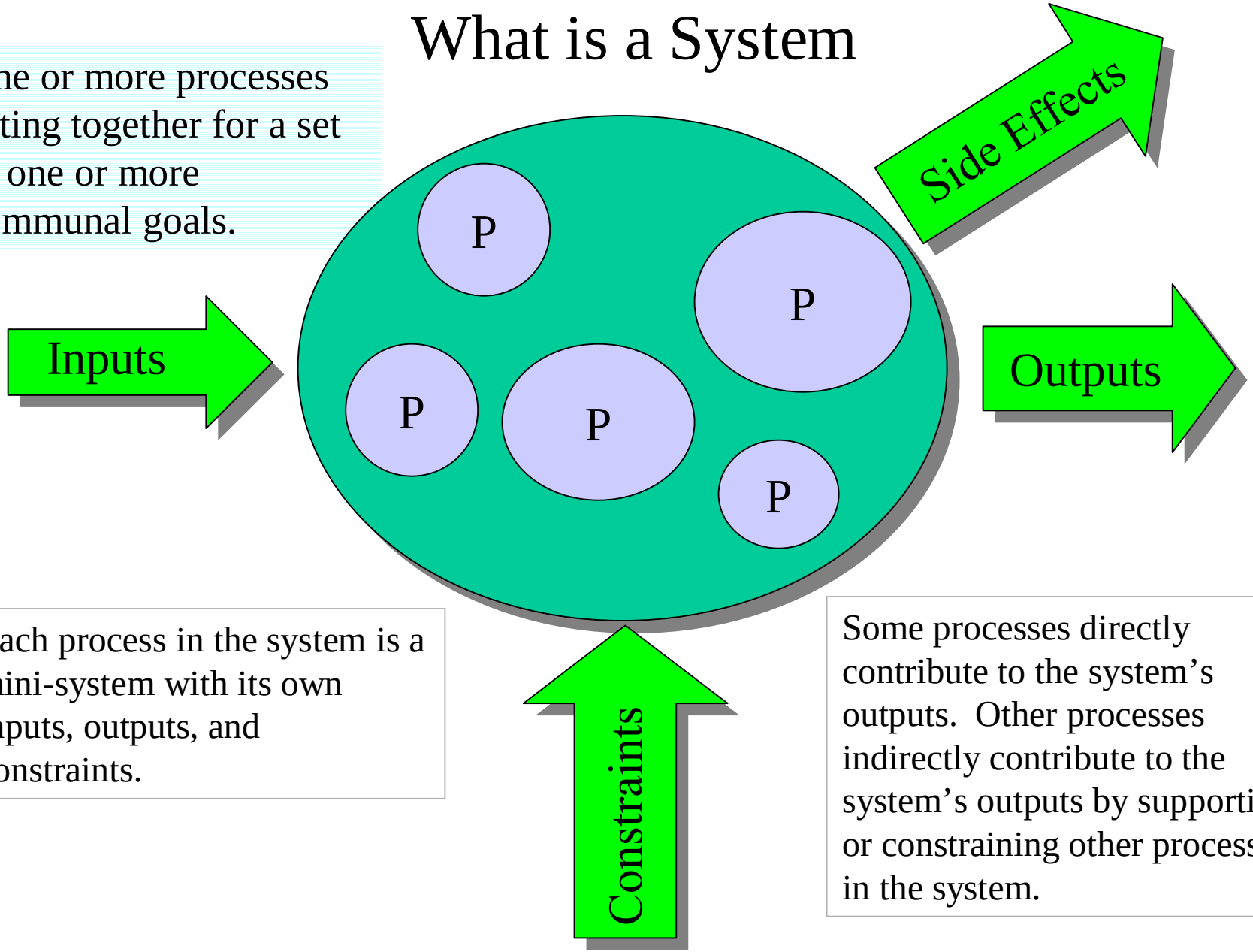
- Since all information and assumption are open to doubt, and with chance at work everywhere, the commander continually finds that things are not as he expected.
  - Carl von Clausewitz - *On War*
- Everything in war is simple, but the simplest thing is difficult. The difficulties accumulate and end by producing a kind of friction that is inconceivable unless one has experienced war.
  - Carl von Clausewitz - *On War*
- The map is not the territory.
  - Count Alfred Korzybski - *Science and Sanity*
- Rules build up fortifications behind which small minds create satrapies. A perilous state of affairs in the best of times, disastrous during crises.
  - Frank Herbert

Things to Remember - Frank Herbert

- Every symbol has hidden premises behind it. Every word carries unspoken assumptions buried in the history of the language and the conditioning experiences of the speakers. If you snatch those buried meanings out of your words, you spill a whole stream of new understanding into your awareness.
- The Zen master tells us that an omnipresent idea can be hidden by its own omnipresence - the forest lost among the trees. In our normal daily behavior we are most estranged, most in the grip of an illusory idea of the self. Every enchanting inclination of pride and its ego, of convention and its master - social training - conspires to maintain the illusion. The semanticist calls it the inertia of old premises. And this is what holds our analyses of consciousness within fixed limits.
- Everything you do, everything you sense and say is experiment. No deduction final. Nothing stops until dead and perhaps not even then, because each life creates ripples. Induction bounces within and you sensitize yourself to it. Deduction conveys illusions of absolutes. Kick the truth and shatter it. Seek the questions that form the best images. Doing this, you never think of yourself as clever, that you have the best formula to provide the solution. You remain as responsive to new questions as you do to new patterns and data. Testing, re-testing, shaping and re-shaping. A constant process, never stopping, never satisfied.

What is a System

One or more processes acting together for a set of one or more communal goals.



Each process in the system is a mini-system with its own inputs, outputs, and constraints.

Some processes directly contribute to the system's outputs. Other processes indirectly contribute to the system's outputs by supporting or constraining other processes in the system.

Characteristics of Simple/Complex Systems

- Predictable versus unpredictable behavior
  - Simple systems are predictable and intuitive due to simpler relationships, loops, and decisions between/within components
  - Complex systems are unpredictable and counterintuitive due to more subtle interactions, loops, and decisions between/within components
- Component interactions and feedback/feedforward loops
  - Simple systems have few components with less coupling between the components with few and well defined loops providing behavior changes from changes in component interactions
  - Complex systems have many components with greater coupling between components with many feedback/feedforward loops restructuring and/or modifying component interactions introducing subtle behavior changes
- Concentrated versus diffuse decision making
  - Simple systems have few decision centers with a few, easily identified and well defined control flows
  - Complex systems have many decision centers with many, hard to identify and poorly defined control flows
- Decomposable versus irreducible
  - Simple systems are easily decomposable due to the simple interactions between the components of the system
  - Complex systems are difficult to decompose or may be irreducible due to the complex interactions between the components of the system

From: Complexification
John Casti

Systemantics

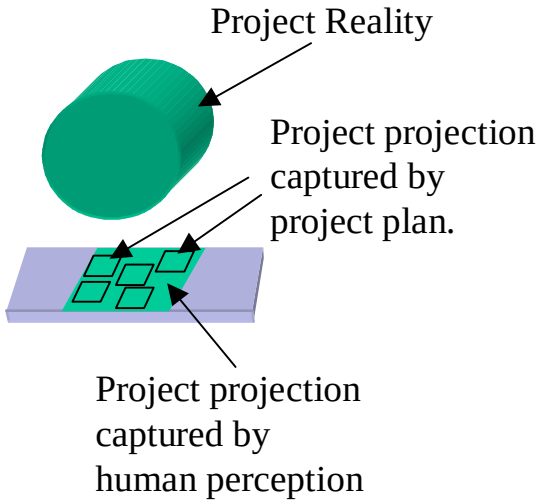
- Systems usually operate in failure mode.
  - Component failures
  - Interface failures
- Systems introduce new problems.
  - Component operational requirements
  - Side effects
- Systems tend to oppose their own proper function.

A Project is a System

A project plan is a **forecast** of:

- the activities needed
- the deliverables needed
- the budget needed
- the resources needed
- the schedule needed
- the processes needed
- the interfaces needed

To produce the project deliverables.
To meet the project goals.



Project Complexity is a measure of the complexity of the management task.

The complexity of a project springs from the number of participants, the interfaces between participants, the fuzziness of the project processes used, and the fuzziness of project goals.

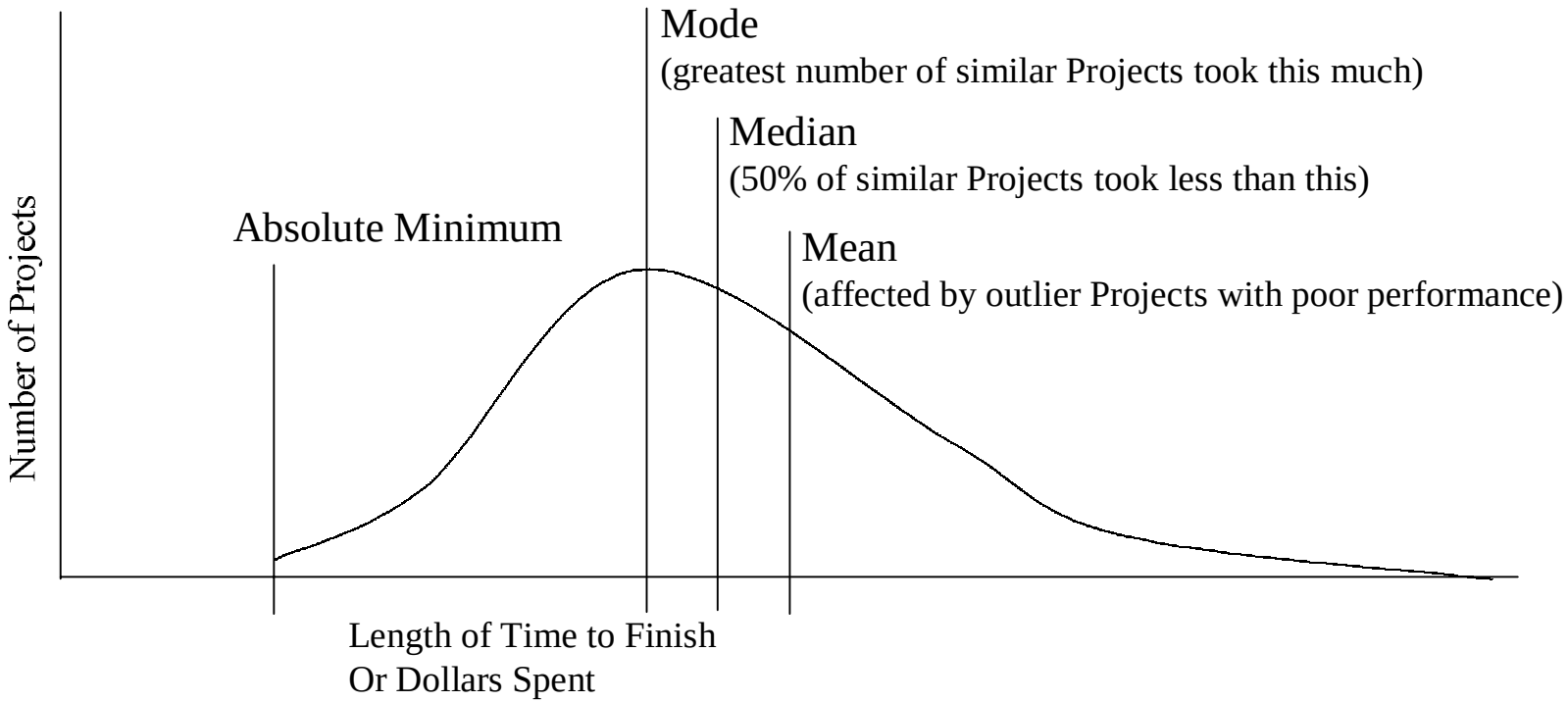
In general, the fewer the number of participants and the more concrete the project processes, the less complex the project.

The formal project processes are a template for the project forecast but project reality is the interaction between the participants using the project processes and the organizations' instruments and apparatus.

A project plan is a projection of the real event, the project reality, onto a set of documents that capture a projection of the real thing. The project plan is an abstraction of the project reality and like any abstraction does not capture the full project nor does it capture the full project reality projection onto human perception.

Human perception includes non-linguistic senses.

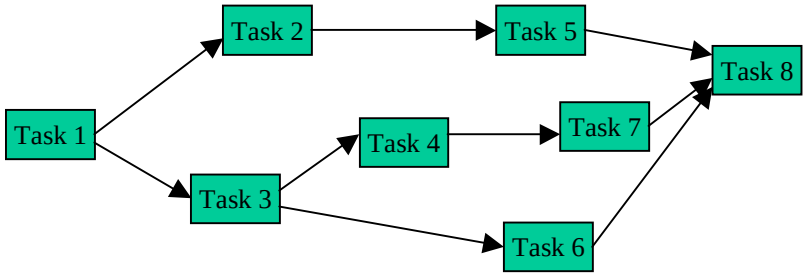
Project Schedule and Budget



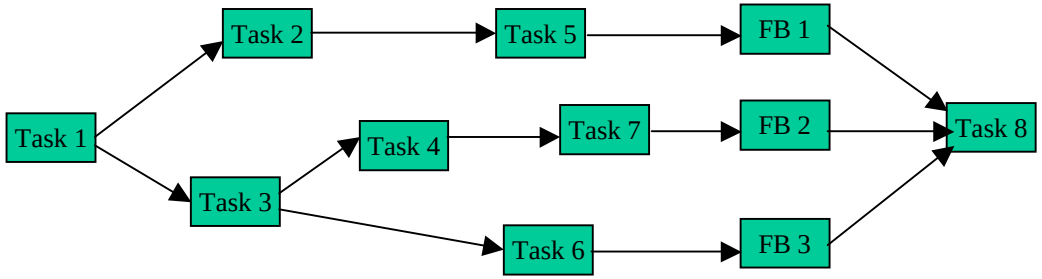
The amount of time for similar projects to complete or the amount of money spent on the project will vary. In a population of similar projects, there is a distribution of completion times and dollars spent which is a Beta distribution.

The question is how to minimize the time and dollars.

Task Level Buffers vs Project Level Buffers



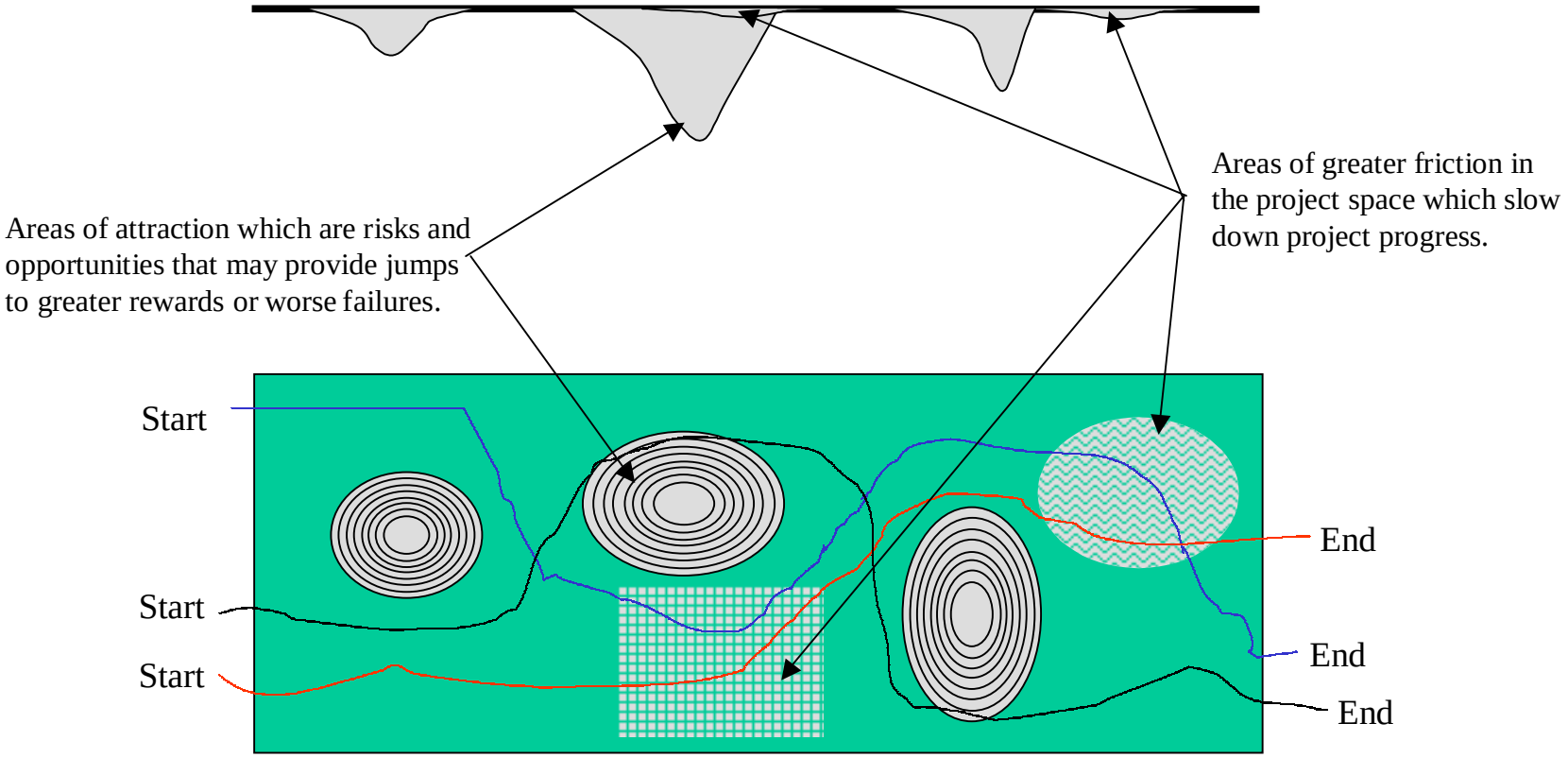
Each task in the project has an internal, unacknowledged buffer for its scheduled time and money. Depending on the confidence and personality of the estimator, the estimate may be within a 50% median or an 80% median or higher with a buffer commensurate with the amount of perceived risk for the task to complete within the scheduled time/budget.



The project has a set of acknowledged buffers for its scheduled time and money which are inserted at locations within the project task flow. The individual tasks are then provided with realistic estimates using a 50% median with the knowledge that the actual values may be greater than the forecasted or planned values. Using 50% median does not imply the project has only a 50% probability of completing in the budgeted time/money.

Project Traversing Project Space Terrain

A project traverses a “terrain” or project space. This project space contains multiple paths which a specific instance of the project may traverse depending on such variables and constraints as type and skill level of resources, quality targets of product, quality of intermediate products and deliverables, and budget.



Surprise Generating Mechanisms

- A major source of surprise is the human inability to trace through the logical consequences of our assumptions.
  - Inadequate mental model of the system
  - Untrue assumptions of the system's behavior - axioms

From: Complexification
John L. Casti

Mechanism	Surprise Effect
Logical tangles	Paradoxical conclusions
Catastrophes	Discontinuity from smoothness (extrapolation)
Chaos	Deterministic randomness (feedback)
Uncomputability	Output transcends rules
Irreducibility	Behavior cannot be decomposed into parts
Emergence	Self-organized patterns

From: The Collapse of Chaos
Jack Cohen, Ian Stewart

The four things of knowing

Things you know that are true
Things you know that aren't true
Things you know that you don't know
Things you don't know that you don't know

The three types of model mistakes

Errors made within the mental model; easy to spot

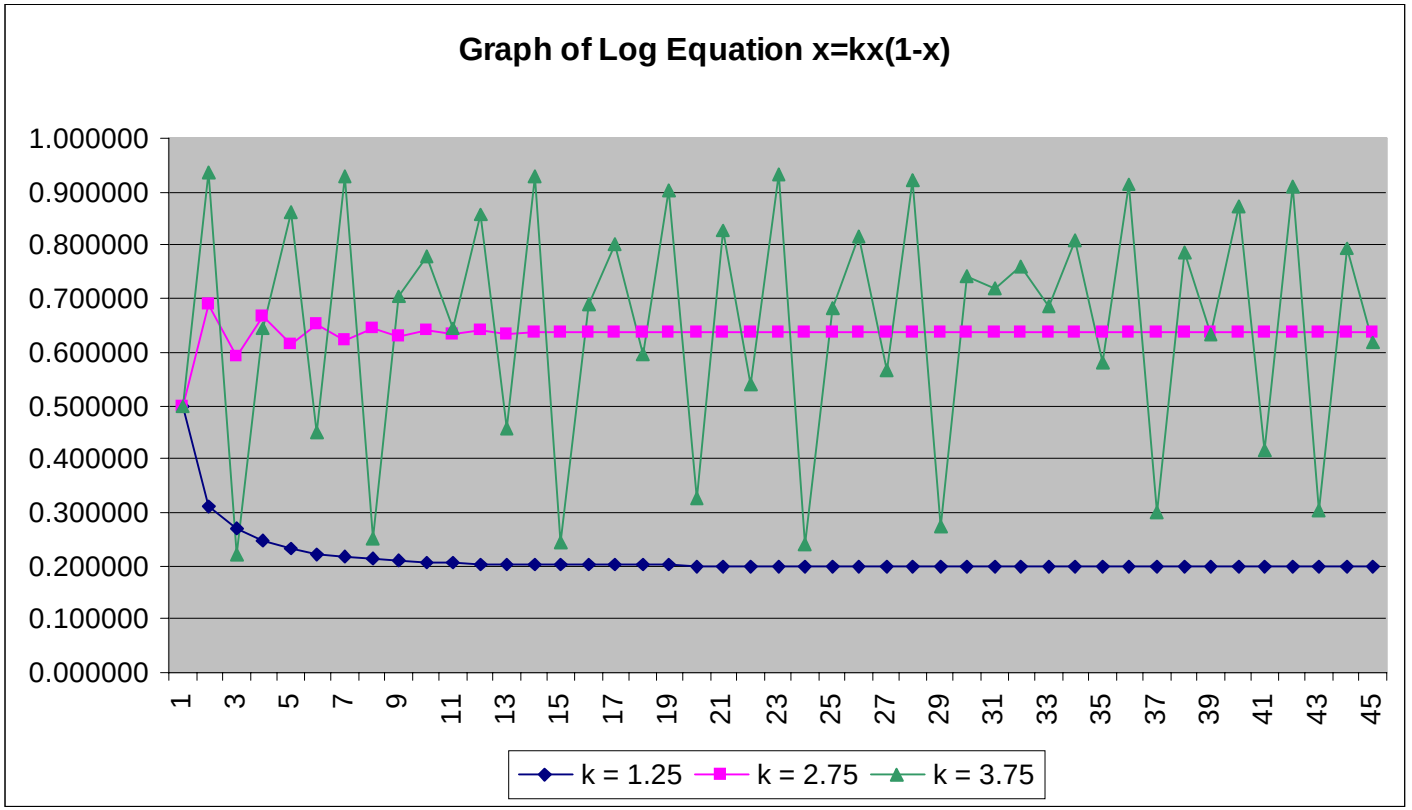
Errors made in the explicit assumptions that lie behind the model; axioms

Errors in the implicit assumptions that suggested the model; worldview, bias, and prejudice

Dependency on Initial Conditions

The log equation is $x_{i+1} = kx_i (1 - x_i)$ which means the next value for x is calculated by k times x times (1 - x).

Values of k from 1 to 3 converge to a single value. Values of k greater than 3 show a periodic change phenomena.

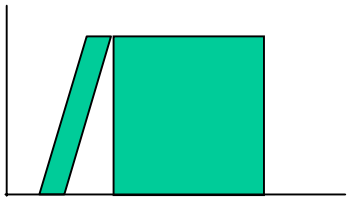


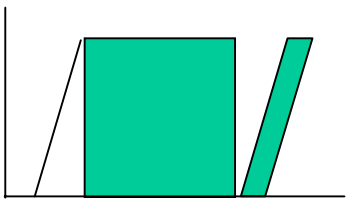
From the book Chaos Theory in the Social Sciences. L. Douglas Kiel and Euel Elliot (editors).

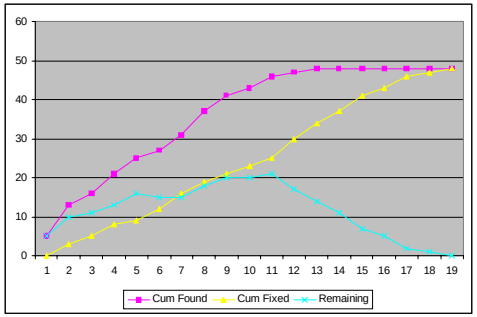
Design Factory - Queues

- Queue responsiveness is affected by uncertainty in task arrival times and uncertainty in task durations
- Factors affecting queue throughput and responsiveness
  - Capacity of the actor(s) which are queued on
  - Managing demand for the actor(s) (queue length)
  - Reducing process variability (task arrival time and task duration)
  - Using queue control systems (measuring queues, intervention)
- Process queue length indicates the health of our process
- Batch transfers cause queues, intermingle large and small batch sizes cautiously

From: Managing the Design Factory
Donald Reinertsen







Design Factory - Information

From: Managing the Design Factory
Donald Reinertsen

Design Factory - Feedback

From: Managing the Design Factory
Donald Reinertsen

Trash Can Theory

5 Political Process Principles

- Inclusion
  - People with any influence or any interest in the problem under consideration must be included; include stakeholders
- Comprehension
  - Remove zero sum game by discussion of the underlying issues bringing assumptions to the foreground; understanding one another
- Deliberation
  - Reviewing openly and honestly the positive and negative faces of the issues to arrive at a consensus of what needs to be done
- Cooperation
  - Working together for mutual benefit, aligning through conflict resolution; resisting mutual disadvantage through competition
- Realism
  - No illusions about the ease of solving the problems, meeting the challenges, and resolving the issues that confront us

From: Practical Politics
Michael K. Briand

8 Ways Smart People are Stupid

- David Perkins lists eight deadly sins of the stupid smart person, (from Why Smart People Can Be Stupid):
  - impulsiveness (doing something rash)
  - neglect (ignoring something important)
  - procrastination (actively avoiding something important)
  - vacillation (dithering)
  - backsliding (capitulating to habit)
  - indulgence (allowing oneself to fall into excess)
  - overdoing (like indulgence, but with positive things)
  - walking the edge (tempting fate)

Data Collection

- Status reports
  - Defect count statistics (Open, Failed, Reply, Solved, Aging)
  - Test case counts (Pass, Fail, Waiting, Executed,)
  - Function point statistics (points submitted, forecasted points)
  - Testing coverage statistics
  - Inspection reports (defects found, re-reviews required)
  - Mapping defect statistics and inspection reports on to system components
- Data collection issues
  - Incomplete reports
  - Errors in the reports
  - Old data or data not to date
  - Unnecessary or misleading data
  - Confusion of data with inferences from data (structural mobile)
  - Inconsistent reports from multiple sources with overlapping areas

Project Status Information

- Data needs to be turned into information
  - Trend analysis based on historical data
  - Trend analysis based on documented patterns
  - Past performance does not reflect future performance
- Information needs to flow to decision makers
  - Actual quality versus quality targets
  - Actual schedule versus schedule targets
  - Actual budget versus budget targets
  - Actual resources/skills versus resource/skill targets
  - Actual deliverables versus deliverable targets

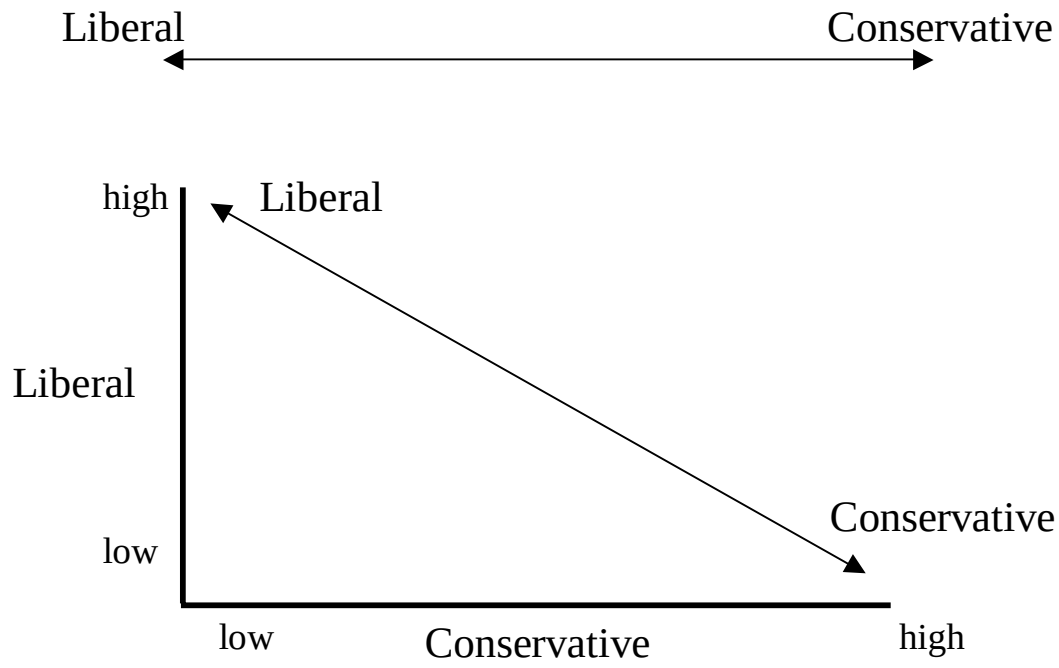
Project Status Information Used

- Necessary information
  - What is project status versus plan
    - Schedule and budget
    - How good is the plan thus far and what changes are needed
  - What are upcoming events, tasks
  - What interventions are required and by who, timeframes
  - What investigations are required and by who
  - Who is doing what, when, with what
  - Feedback for decision makers
    - Decision consequences
    - Adjustments to courses of action

Information

- How good is the product
  - Quality targets of available functionality
  - Customer needs met with available functionality
- External Service level requirements
  - From vendors internal or external to the organization
  - Measurements as to service level quality
  - Documentation as to interfaces and products transferred over the interfaces
- What interventions are required
  - Additional resources (staff, equipment)
  - Scope changes (add or remove functionality)
  - Change quality targets

One Dimensional vs Two Dimensional

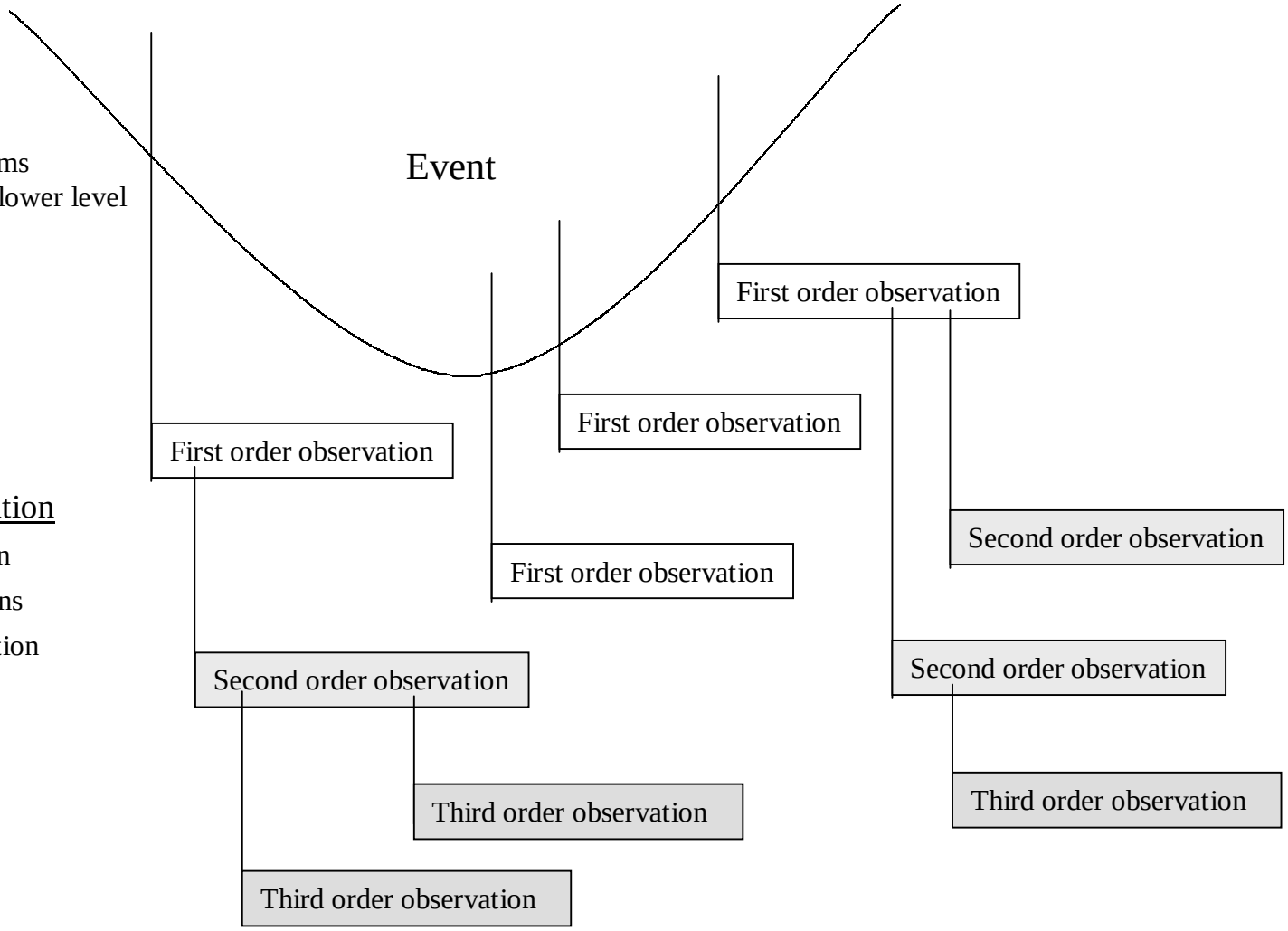


One dimensional categorization polarizes two extremes. People are basically asked to pick a place on a one dimensional continium or line between the two extremes. Using two valued logic, people are asked to pick between the two alternatives when they may not believe either alternative is “good”.

Two dimensional categorization allows people to mix the two extremes using different levels and mixtures of the two categories.

Structural Mobile

Level 1: Perceived data items
Level n: Abstraction from lower level



Observation Filters

Selective Attention

Selective Distortion

Selective Memory

FOG Factor for Observation

Facts - verifiable observation

Opinions - Level n deductions

Guesses - Level n extrapolation

From: Science and Sanity
Alfred Korzybski

See also General Semantics

Ladder of Abstraction

asset — 7. When Bessie the cow is referred to as "asset" only those characteristics she has in common with other assets such as farm assets, manufacturing assets, city assets, etc are referred to.

farm assets — 6. When Bessie the cow is referred to as "farm assets" only those characteristics she has in common with other salable items on the farm such as livestock, farm machinery, farm buildings, etc are referred to.

livestock — 5. When Bessie the cow is referred to as "livestock" only those characteristics she has in common with pigs, chickens, goats, etc are referred to.

cow — 4. The word cow stands for the characteristics we have abstracted as common to all cows in general.

Bessie the cow — 3. The name or symbol attached to the perceived cow standing in the field

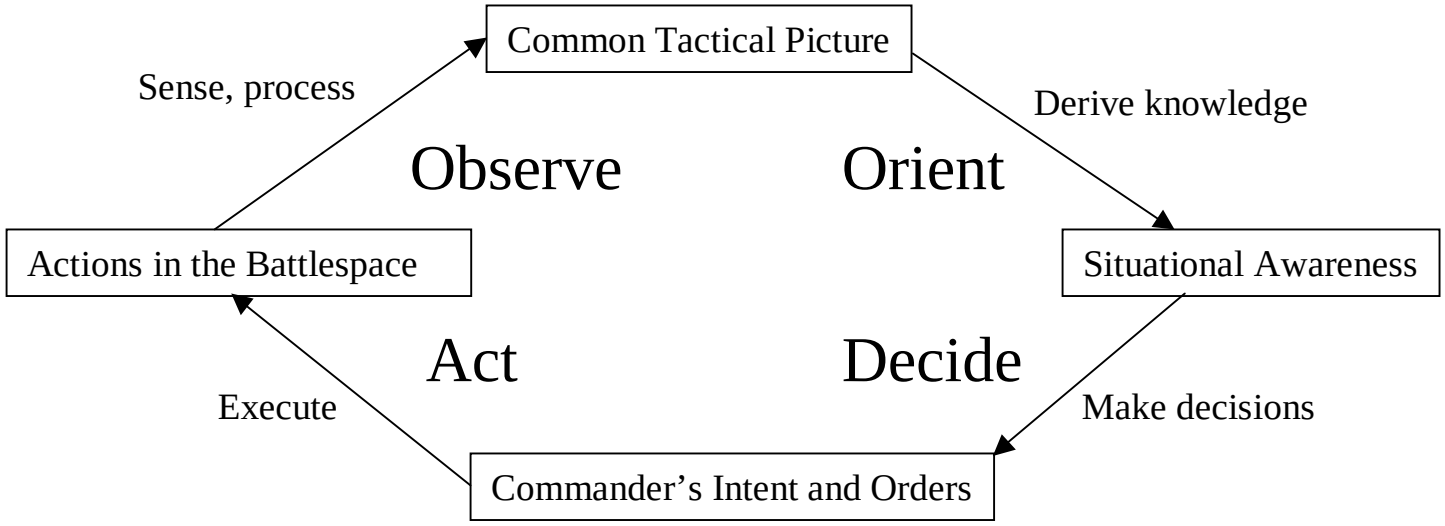
2. The perceived cow standing in the field

1. The physical cow standing in the field

From: Language in Thought and Action

See also General Semantics

OODA Cycle



From: Unpublished work
Colonel John Boyd, USAF

Project Plan Variables

- Project task schedule interdependencies
  - Probability variables or fuzzy variables
- Project information flows
  - Bias in project information due to sender and receiver preconceptions
- Project supply requirements
  - Goods and services
- Project task deliverables
  - Coupling between tasks and task deliverables
  - Cohesion of tasks and activities
  - Interfaces between tasks and task groups
  - Defect counts in task activities and task deliverables
  - Process improvements eliminating opportunities for defects

Project Management Tools

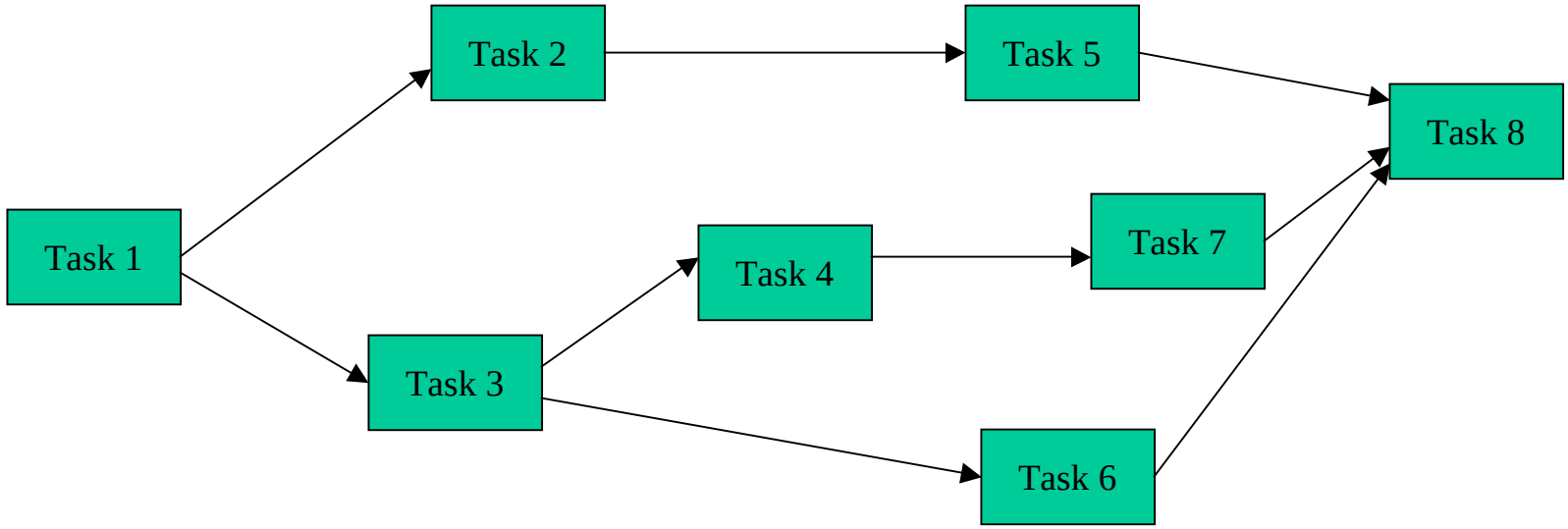
- Tools to show
  - Activities, durations, dependencies
  - Deliverables and intermediate work items
  - Interfaces between work items
  - Interfaces between work groups
  - Development process and work flow
  - Budgets
  - Current status of deliverables
    - Compare against schedule
    - Compare against budget
    - Compare against quality targets

Project Management Tools

- Tools to create/generate
  - Project plan documents
    - PERT charts
    - GANTT charts
    - Work item and project deliverable maps
    - Budgets with past and forecasted expenditures
  - Development and test effort maps
  - Vendor test results documents
  - Defect statistics documents
  - Status reports

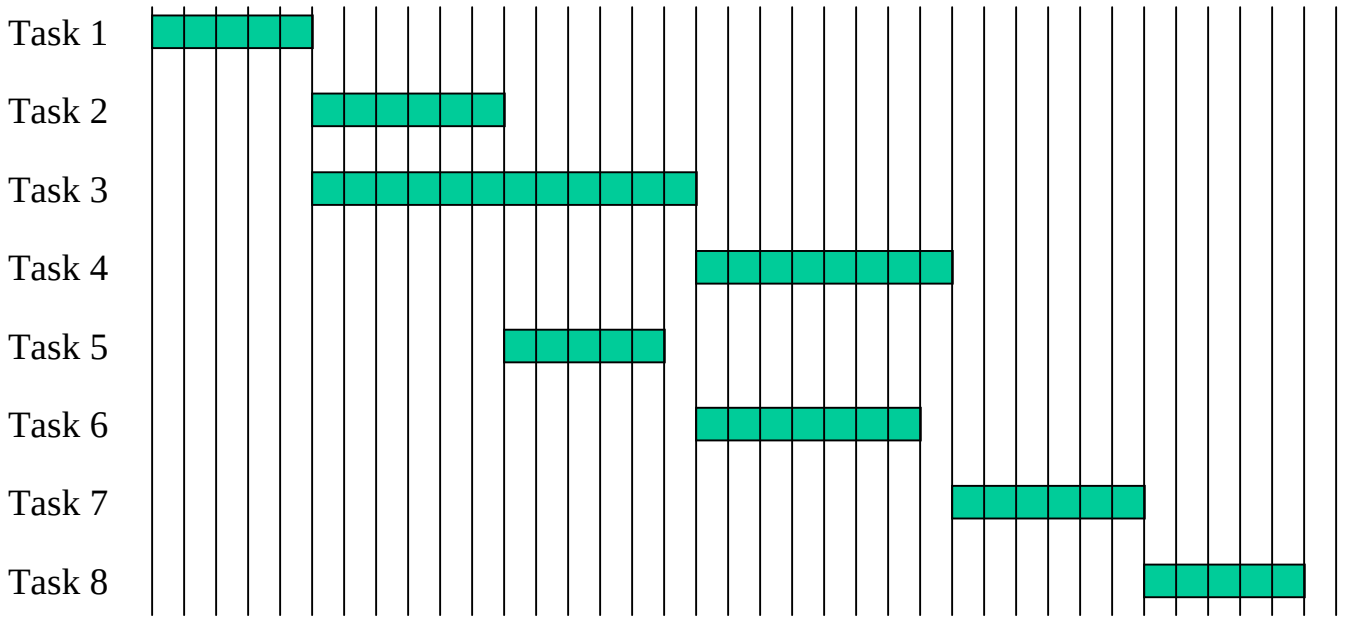
PERT Chart

The PERT chart provides a picture of dependencies of effort for the project.



GANTT Chart

The GANTT chart provides a picture of timelines and concurrency of effort for the project.



Subproject Status Map

Item	Schedule	$ Budget	Heads	Equip	Skills
Subproject #1					
Subproject #2					
Subproject #3					

Managers and team leaders manage their internal problems but must report problems which may or will impact other groups meaning problems which may or will impact outputs (deliverables) of the subproject required by other subprojects or the project itself. New side effects or modified side effects must also be reported so that other subprojects can evaluate whether the changed conditions impact them.

Everything manageable	No impacts to other groups.
Intervention probable	Problems which may impact other groups.
Intervention required	Problems which will impact other groups.

Development and test effort map

Allocation of system functionality to components.

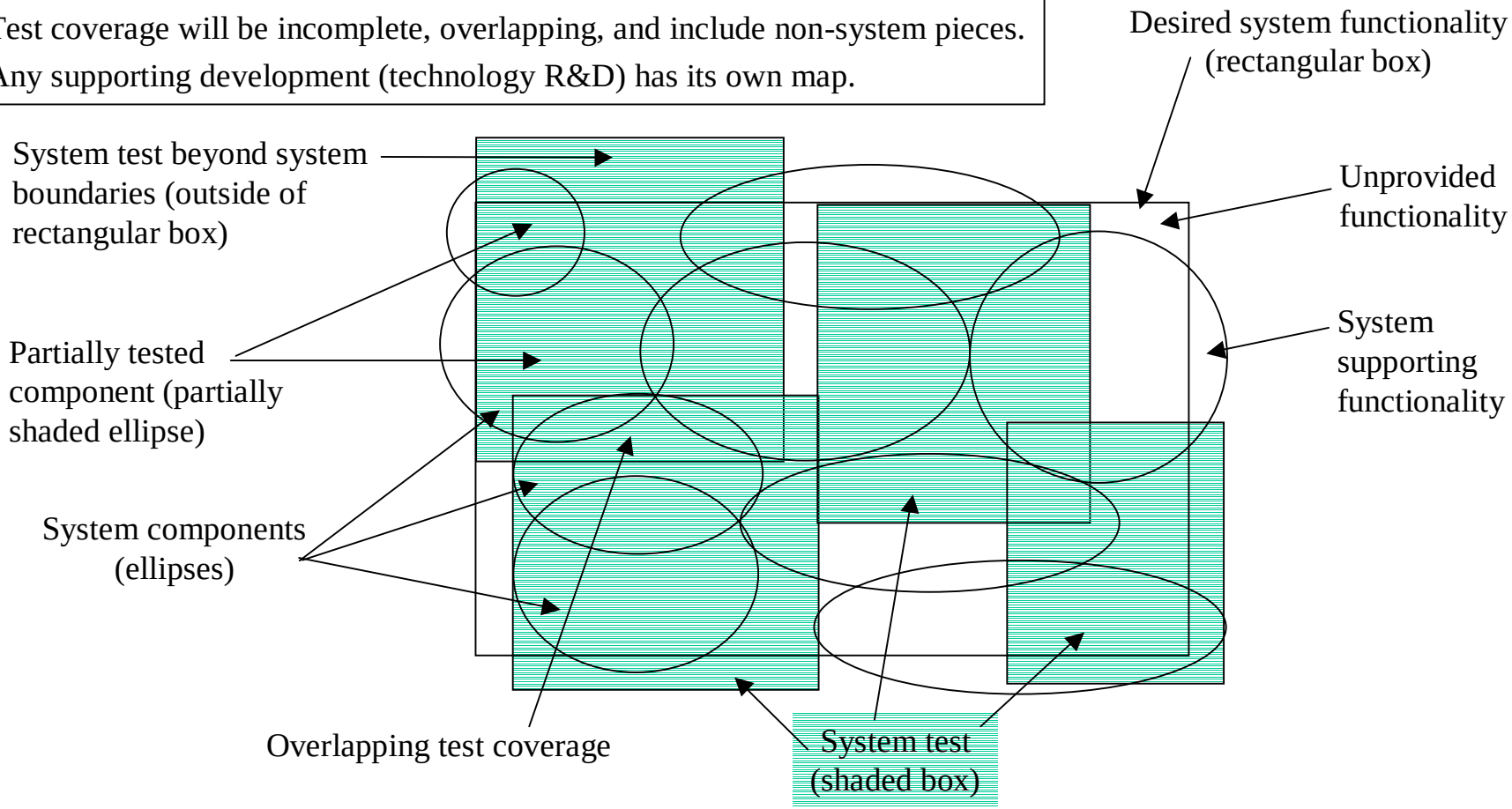
Estimates as to component development requirements.

The desired system functionality may not be achievable.

The actual system functionality is union of system components.

Test coverage will be incomplete, overlapping, and include non-system pieces.

Any supporting development (technology R&D) has its own map.



Software Defect Removal Step Efficiencies

Types of Defects
Conceptual
Design
Implementation
Transformational

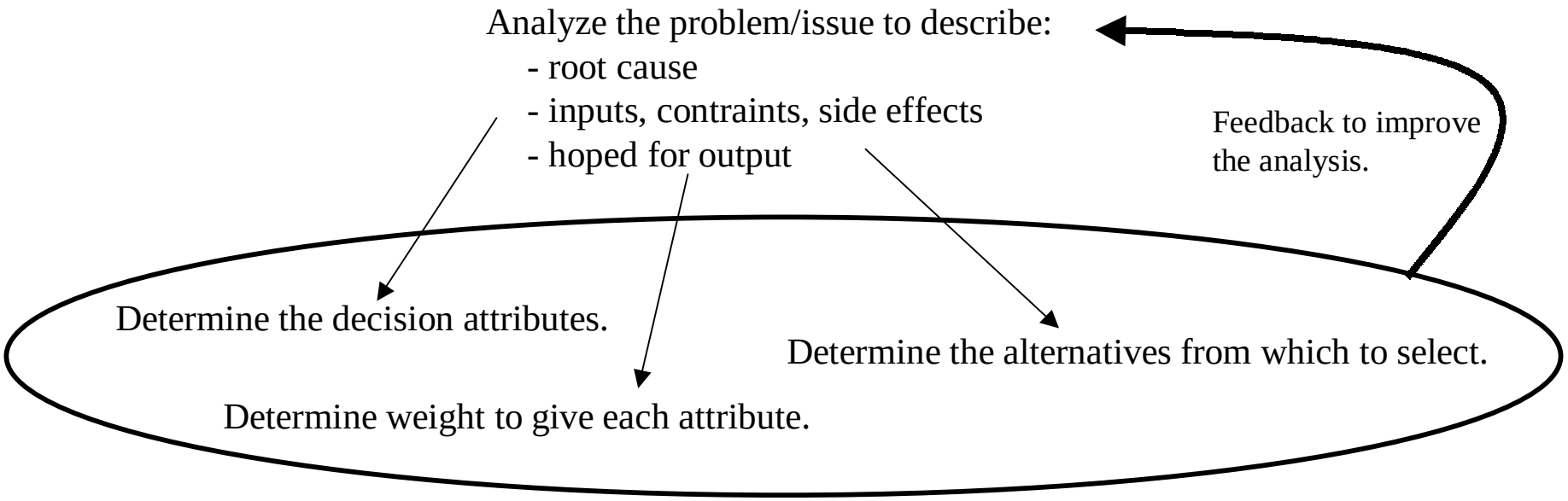
Note: testing has defect opportunities similar to software design and implementation defect opportunities. Development of a test plan is similar to development of a software design since the test plan can be seen as a type of computer program that will be executed by a person.

During test plan execution, there are opportunities for missing defects as well as finding false defects. Whenever possible, the software should be instrumented to assist the tester with the identification of error conditions.

Source: Capers Jones, 1986
Programming Productivity

Removal Step	Lowest Efficiency	Modal Efficiency	Highest Efficiency
Personal checking of design or documents	15%	35%	70%
Informal group design reviews	30%	40%	60%
Formal design inspections	35%	55%	75%
Formal code inspections	30%	60%	70%
Modeling or prototyping	35%	65%	80%
Personal desk checking of code	20%	40%	60%
Unit testing (single modules)	10%	25%	50%
Function testing (related modules)	20%	35%	55%
Integration testing (complete system)	25%	45%	60%
Field testing (live data)	35%	50%	65%
Cumulative efficiency of complete series	93%	99%	99%

Weighted Scoring Decision Model



Evaluate each alternative value for each attribute.

Score each alternative using attribute weights and scores
- assigned value times weight for each attribute
- total attribute score is sume of attribute scores
- pick alternative with highest total score

Weighted Scoring Decision Model

Determine the attributes that are involved in a good decision. The attributes should map to specific issues, concerns, risks, assumptions, goals, and objectives for the decision. Write out the basis for the attribute so that as you score a decision alternative on each of the attributes, you will judge all alternatives on a reasonably equitable basis.

Determine the weight of each of the attributes as a part of the decision making mix. The weight signifies the importance of the attribute towards making a good decision. You should have some reason or reasons for the weight which is basically a comparitive weight as to how much the attribute should affect the total decision when compared to the impact of other attributes. The weight would normally be on a 6 point scale with 1 being low and 6 being high. Using 6 point scale, you can not be intellectually lazy and pick an average weight.

Assign a value to how well the decision alternative provides for, instantiates, or is imbued with each of the attributes in the decision model. Try to score using an absolute measure of how well the alternative imbues or instantiates the attribute. Use a 6 point scale with 1 low and 6 high. Using a 6 point scale means you must select high or low amounts of instantiation of the attribute but can not pick average.

Score each of the decision alternatives by doing the following for each: multiple the weight of each alternative by the value assigned to the alternative during the alternative evaluation and then sum the attribute scores to arrive at a decision alternative score. Pick the alternative with the highest score.

Relationship Diagram

Relationship between set of variables $x_1, x_2, \ldots, x_n$ and set of variables $y_1, y_2, \ldots, y_m$
Relationship function generates a table describing the relationship between the two variables x_i, y_j.

Example: Participant y_j is R-related to issue x_i if and only if participant y_j is neutral or favorable toward issue x_i

This relationship diagram shows that Saudi Arabia more nearly agrees with Israel on each of the issues than any other Arab state. It also indicates that the Palestinians and Israel are close on most issues. Finally it shows that there are particular issues on which all the Arab states agree yet Israel disagrees.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
y_1	0	1	1	1	1	1	0	0	1	1
y_2	1	1	1	0	1	1	1	1	1	0
y_3	1	1	0	0	1	1	1	1	1	1
y_4	1	1	0	0	1	1	1	1	1	0
y_5	1	1	0	0	1	1	1	1	0	0
y_6	1	1	1	0	1	1	1	1	1	1

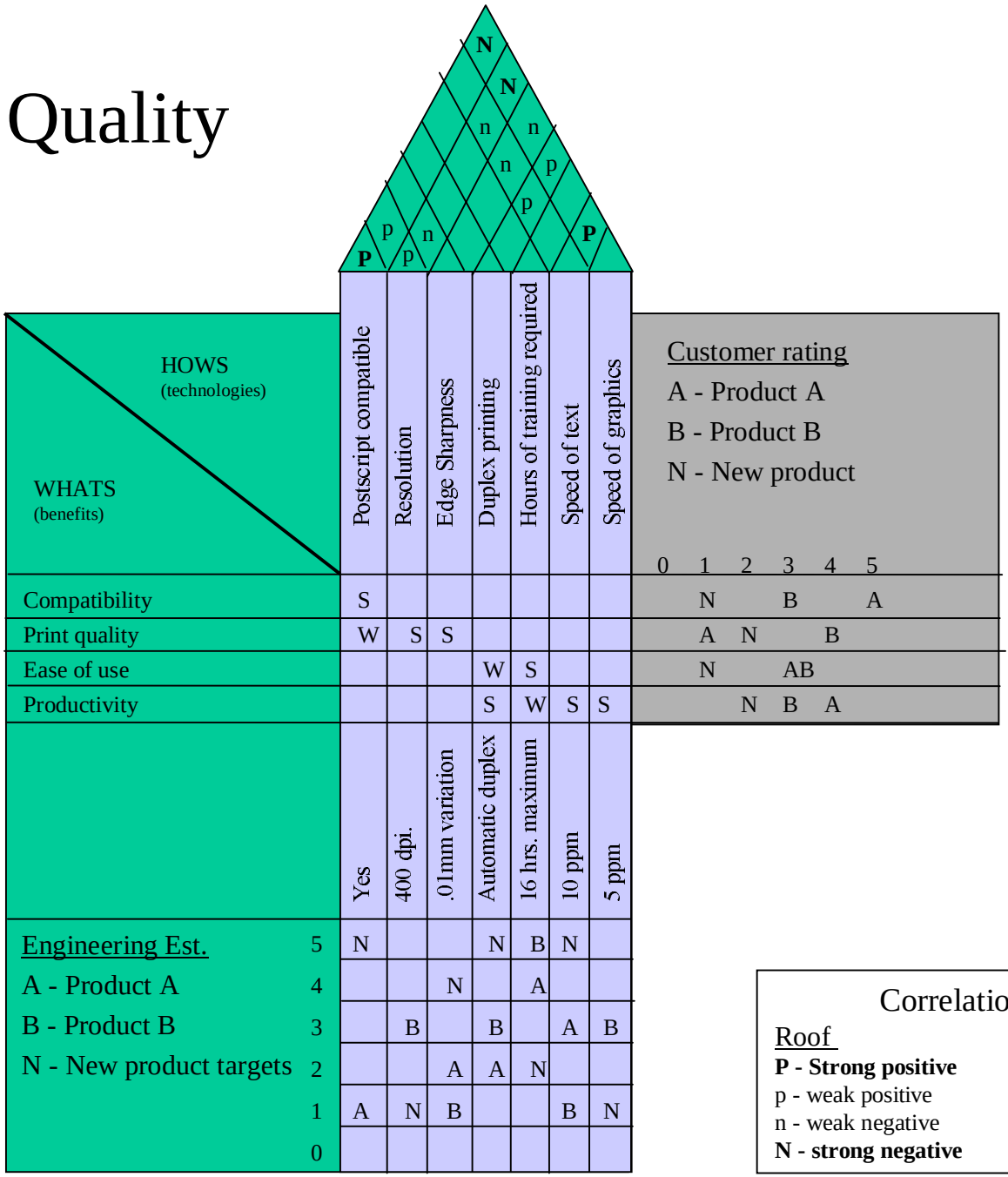
The matrix body contains true or false indications which show the Relationship between the two variables. The matrix body could also use fuzzy indicators (values between 0 and 1 inclusive with values higher than .5 indicating greater degrees of relationship).

From: Complexification
John Casti

y_1	Israel
y_2	Egypt
y_3	Palestinians
y_4	Jordan
y_5	Syria and Iraq
y_6	Saudi Arabia

x_1	autonomous Palestinian state on the West Bank and Gaza
x_2	return of the West Bank and Gaza to Arab rule
x_3	Israeli military outposts along the Jordan River
x_4	Israel retains East Jerusalem
x_5	free access to all religious centers
x_6	return of Sinai to Egypt
x_7	dismantle Israeli Sinai settlements
x_8	return of Golan Heights to Syria
x_9	Israeli military outposts on the Golan Heights
x_{10}	Arab countries grant citizenship to Palestinians who choose to remain within their borders.

House of Quality



Customer Attributes Identified through market research: focus groups, interviews, etc. Usually contains benefits (needs met) but sometimes features are so mandatory they are added as well.

Engineering Estimates part of table basically shows what is possible with more concrete engineering specifications.

Correlations

Roof	Matrix
P - Strong positive	**S - Strong**
p - weak positive	M - Medium
n - weak negative	W - weak
N - strong negative	

Function Point Analysis

- Purpose: forecast development effort
  - Amount of design
  - Amount of code
  - Amount of testing
  - Number of test cases
- Methodology
  - Function points measure the amount of functionality the system needs to provide
  - Thinking in terms of functionality allows for allocation of functionality to components.