

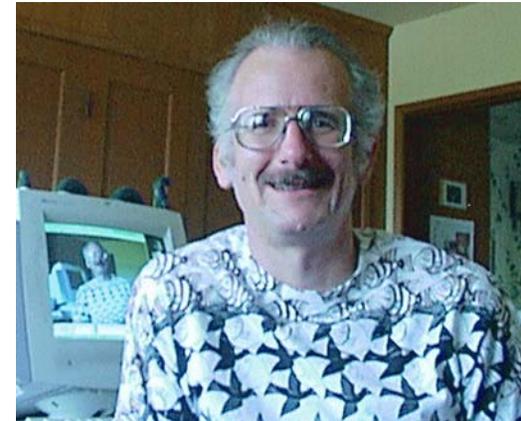


Infusing and Selecting V&V Activities



Martin S. Feather
Jet Propulsion Laboratory
California Institute of Technology

Martin.S.Feather@Jpl.Nasa.Gov
<http://eis.jpl.nasa.gov/~mfeather>



This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

Funded by NASA's:

- Code Q FDPP program
- Code Q/IV&V ARRT task
- Code R ECS program, and
- JPL CSMISS SET



The Challenge

The amount of flight software being flown and the complexity of demands on that software are increasing dramatically, so it is becoming increasingly more important to...
"...Do the right things right the 1st time..."

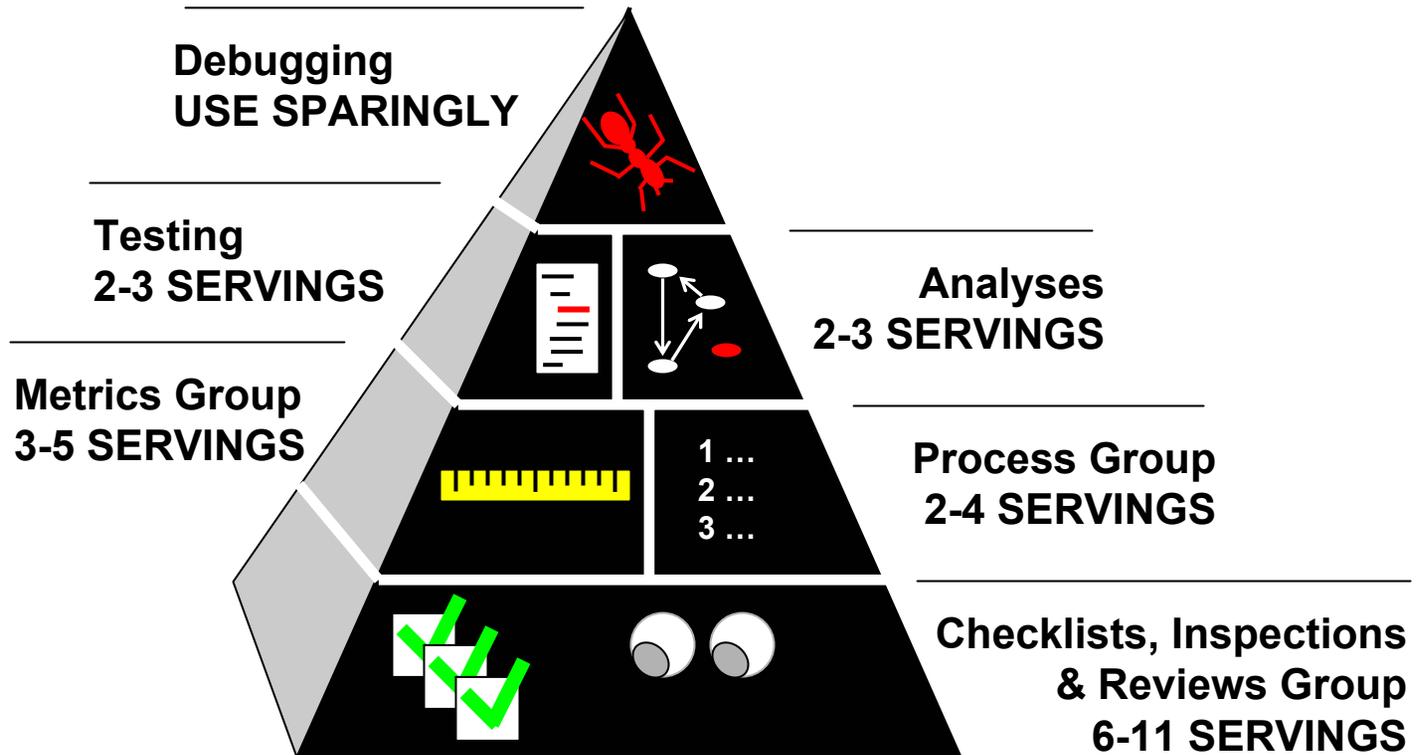
Easy to say, but

- How do we determine what are **the 'right' set of assurance activities** for a specific project?
- What are the **benefits of applying a set of assurance activities?**
- What are the **residual risks even after applying a selected set of assurance activities?**
- Are there **unnecessary redundancies in assurance activities** with respect to individual risks?
- Is there a way to **optimize** selection of the set of assurance activities?

Note: David Kuhn, 1st speaker of this combined session asked:
"When and where do these [formal] methods make sense?"



Hypothetical V&V Pyramid



Not complete - just to convey the idea!



V&V selection and infusion

Objective: improve development process and product

V&V Selection: risks include

- *Development process* risks
 - over budget,
 - behind schedule, ...
- *Product (in flight/use)* risks
 - catastrophic failure,
 - diminished length of survival,
 - degraded science return, ...

Reduce risk through

- Training, inspections, code walkthroughs, formal methods, defensive programming, unit tests, stress tests, ...

Objective: improve infusion of V&V techniques

V&V Infusion: risks include

- *Technical* risks
 - won't scale,
 - false alarms, ...
- *Acceptance* risks
 - can't predict budget for
 - resistance to yet another tool/language
 - skepticism, ...

Reduce risk through

- Courseware, pilot studies, further research, teaming, automation, abstraction, ...

(see paper for example)

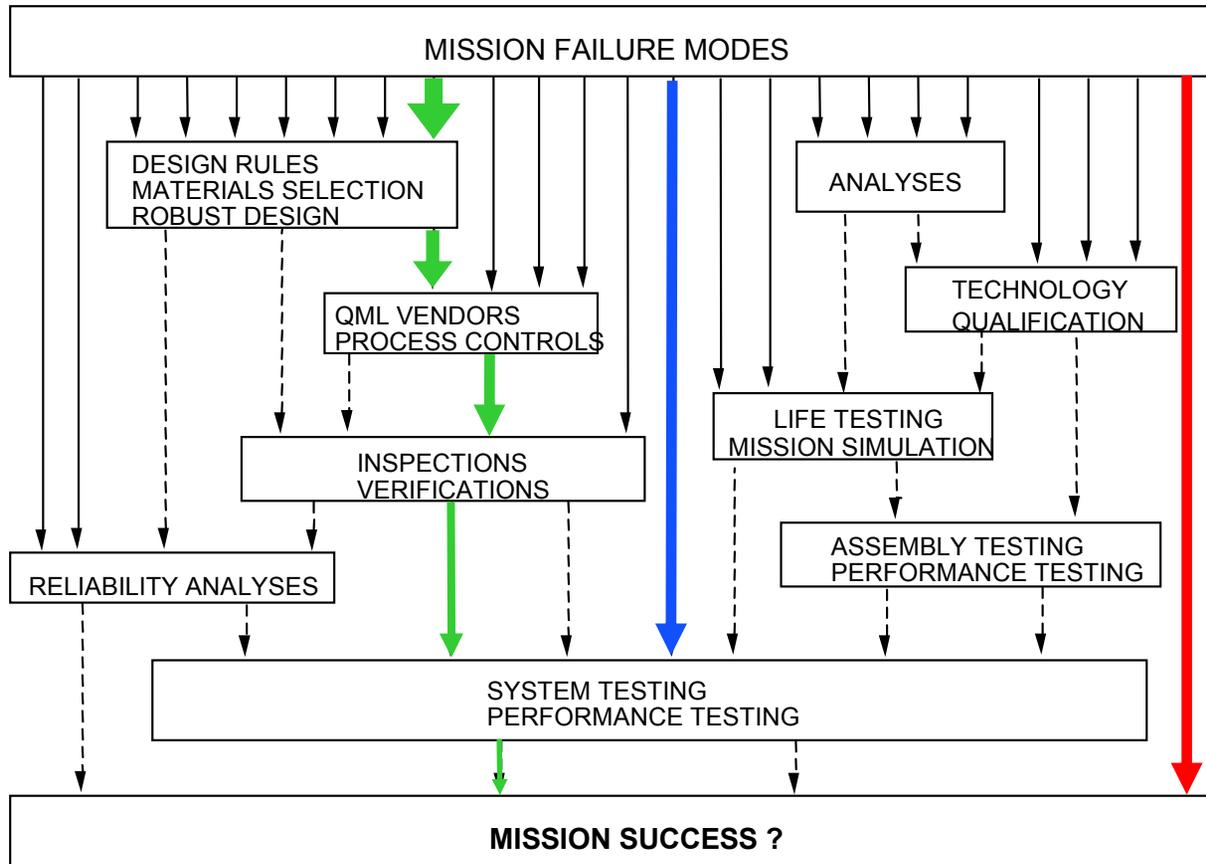


Steve Cornford's Inspiration

assurance activities "filter out" risk



overfiltered risk *singly filtered risk* *unfiltered risk*





DDP: A quantitative model of risk and means to reduce it



Risks, should they occur, cause loss of objectives.

Risks derive their severity from how much they adversely impact objectives, and how important those objectives are.

Assurance activities, if they are applied, reduce risks by:

- **Preventing** them from arising in the first place.
- **Detecting** them (tests and analyses) prior to use in flight (so that there is the opportunity to repair them).
- **Alleviating** their impacts should they occur.

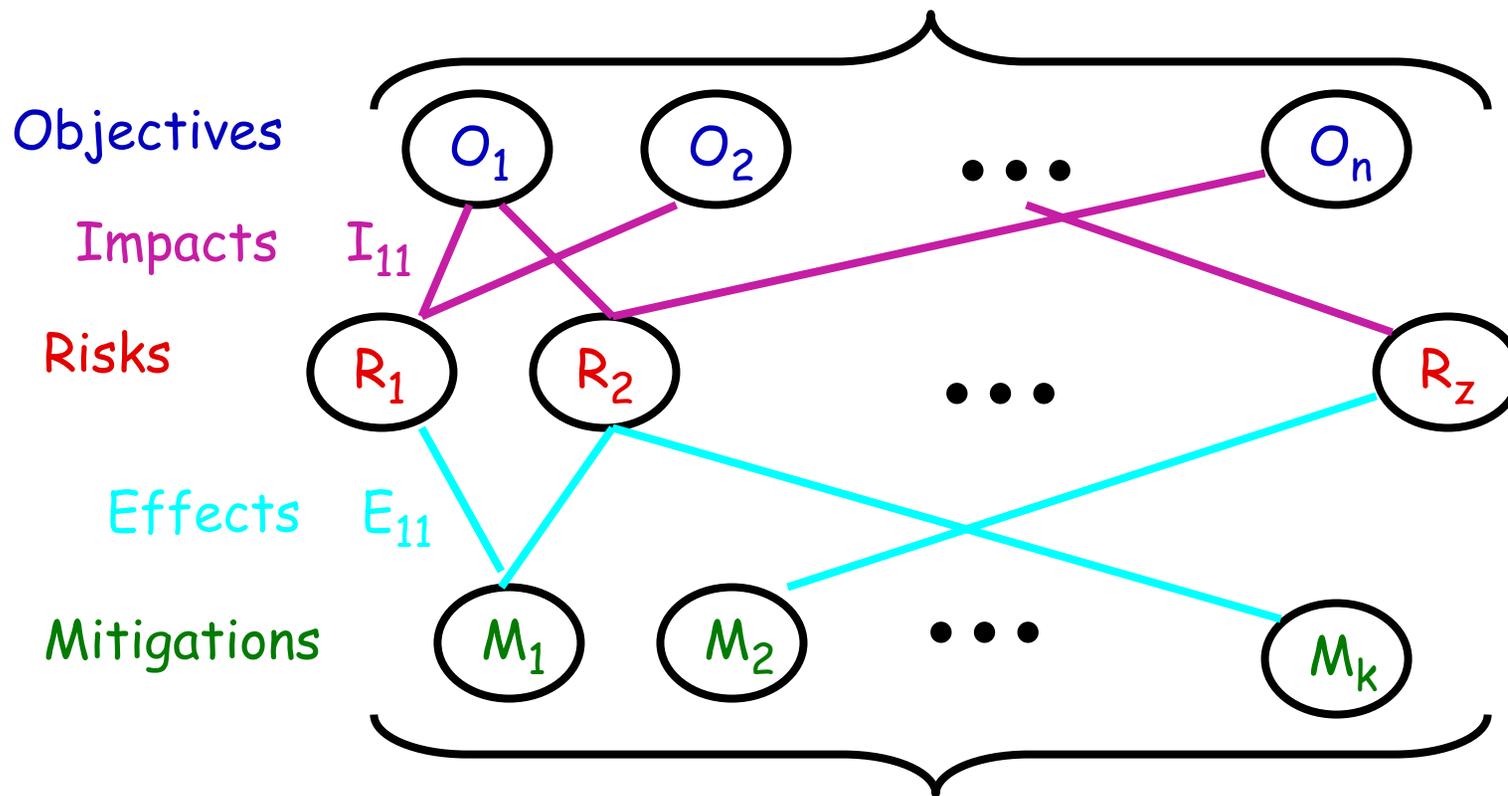
But, assurance activities have costs.

"Risk as a Resource" - Dr. Michael Greenfield
<http://www.hq.nasa.gov/office/codeq/risk/>



DDP Risk Model - the Topologist's View

Benefit = Σ attainment of Objectives



Cost = Σ cost of Mitigations & Repairs

Shallow but broad "influence diagram" (a.k.a. Bayesian)



DDP's Risk Model - Overview



Objectives (what you want)

Risks (what can get in the way of objectives)

Mitigations (what can mitigate Risk - decrease likelihood/severity)

Impact (how much Objective loss is caused by a Risk)

Effectiveness (how much a Mitigation reduces a Risk)

Note: **Objectives**, **Risks** and **Mitigations** inclusive of all relevant concerns

In the past we have also referred to these as:

"Requirements", **"Failure Modes"** and **"PACTs"** -

Preventative measures (e.g. design rules, training), **Analyses** (e.g., software fault tree analyses (SFTAs)), process **Controls** (e.g. coding standards), **Tests** (e.g. unit tests, system tests, stress tests)



DDP Risk Model - Details

Objectives - have **weights** (their relative importance)

Risks - have **a-priori likelihoods** (how likely they are to happen if not inhibited by Mitigations), usually left at the default of 1 (certain!)

Mitigations - have **costs** (\$, schedule, high fidelity test beds, memory, CPU, ...)

Impact (Objctv x Risk) - if Risk occurs, proportion of the Objective lost.
Combine *additively* (n.b., objectives can be more than 100% killed!).

Effectiveness (Mtgn x Risk) - if this Mitigation applied, *proportion* of Risk reduction. Combine as serial filters: $E1 \& E2 = (1 - (1-E1)*(1-E2))$

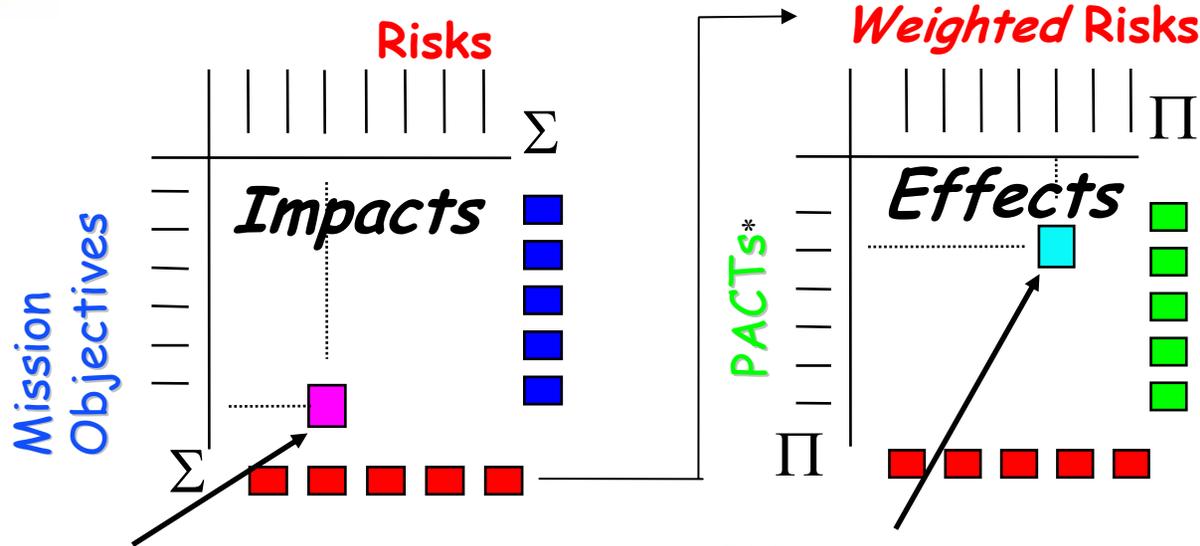
e.g., a 0.8 effectiveness Mitigation catches 80% of incoming Risk ,
a 0.3 effectiveness Mitigation catches 30% of incoming Risk ;
together have 86% effectiveness: 100% -> 20% -> 14%

$$(1 - (1 - 0.8)*(1 - 0.3)) = (1 - 0.2*0.7) = (1 - 0.14) = 0.86$$

Purpose of DDP is to judiciously decide which Mitigations to apply, to balance **cost** (of their application) and **risk** (loss of objectives of not applying them).



DDP Risk Model - Statistician's View



Impact of a given Risk on a particular Objective

Effectiveness of a given Mitigation to detect, prevent or alleviate a particular Risk

Sum the rows: how much each objective is "at risk".
 Sum the columns: how much each Risk causes loss of Objectives.
 Transfer columns to 2nd matrix.

Sum the rows: how much each Mitigation reduces Risks; "solo" or delta".
 Sum the columns: how much each Risk detracts from Objectives (1) when Mitigations off, (2) when Mitigations on.

DDP's quantitative treatment allows Risk to be the interim concept that connects benefit (Objectives attainment) with cost (performing Mitigations).



DDP in Practice

Applied early in lifecycle, when lack detailed and/or well understood designs

- _ Maximal influence is when have minimal information
- _ Handle programmatic risk as well as technical risk

Must scale to large problems

- _ Spacecraft domain involves a multitude of challenges, many experts involved
- _ Pushing the envelope deployment of new technology, mixes old and new challenges

Typical numbers

- _ Objectives, Risks, Mitigations: 30-200 of each
- _ non-zero Impacts and Effects: approx. 1000 of each
- _ 10-20 **experts** involved in 3 half-day sessions

Objectives

- _ Optimize selection of Mitigations
- _ Push back on Objectives (trade for cost savings)
- _ Understand purpose of Mitigations (which Risks they reduce)



DDP Results

Initial reluctance / skepticism of value of process

Anecdotal evidence of success

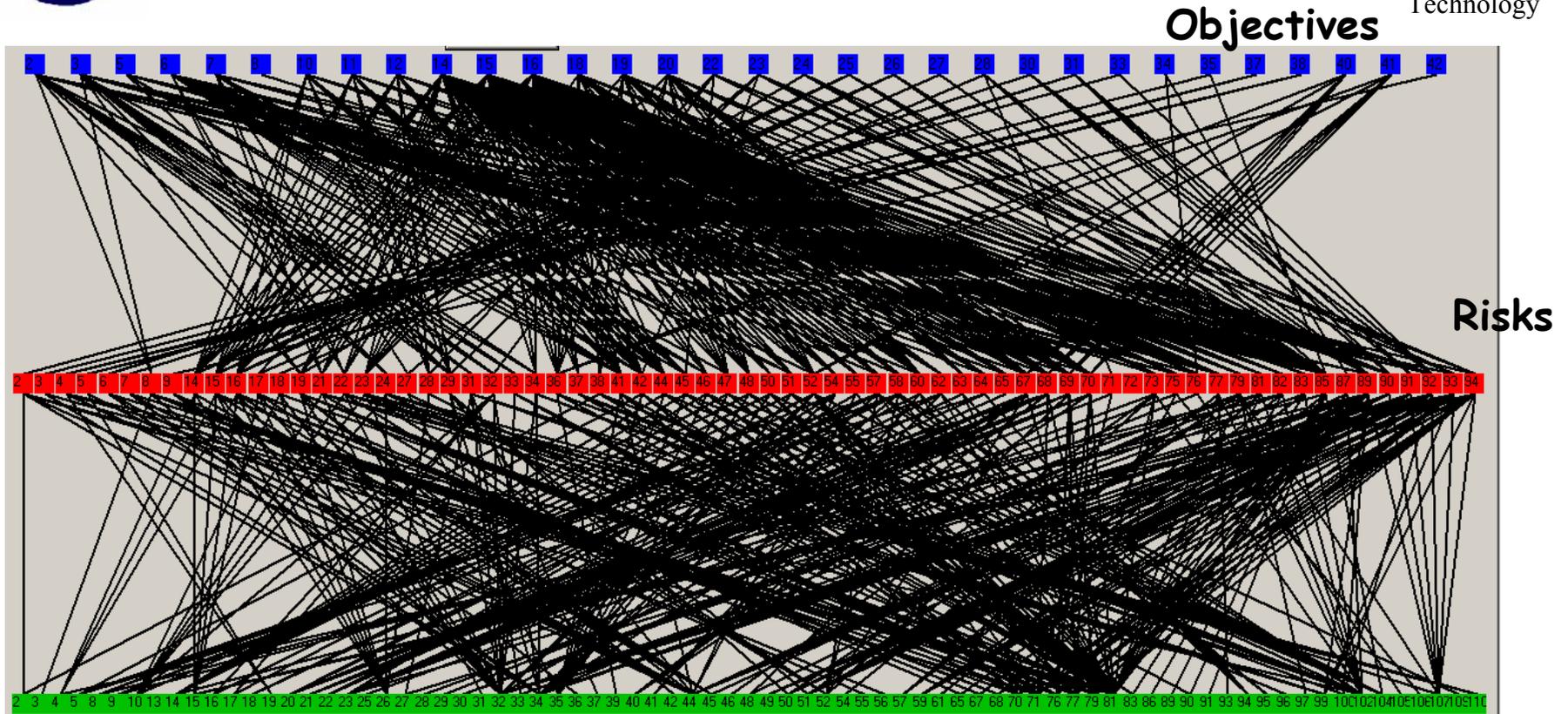
- _ Final consensus on high value of process
- _ Homed in on genuine problems
- _ Identified superior solutions in resource challenged problems
- _ Provided defensible solutions

Recurring drawbacks of approach

- _ Combination rules require explanation
- _ Effort it takes to input the data
- _ Skepticism of validity of results, based as they are on simplistic model and multitude of estimates
- _ Data/Estimates particularly weak for software



Raw topological presentation of a DDP risk model

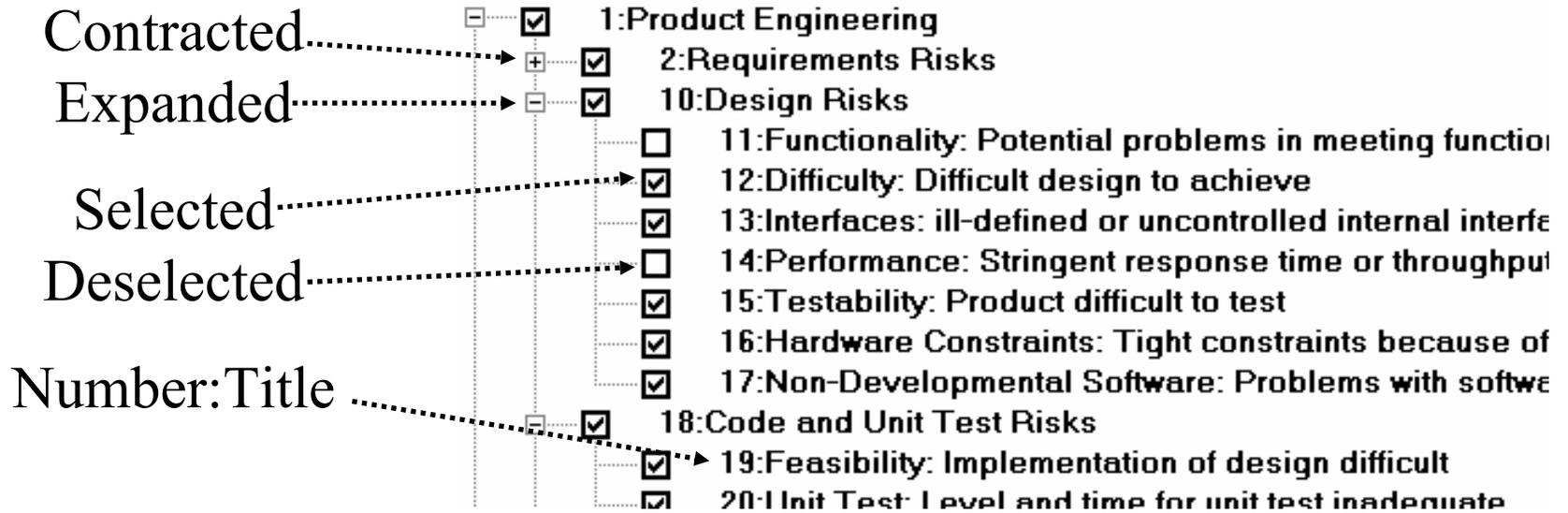


DDP process and custom tool enables models of this scale to be built and used effectively without ever seeing the underlying topology



DDP Trees

Objectives / Risks / Mitigations



Autonumbering: *linear* 1,2,... or *tree* 1, 1.1, 1.2, 1.2.1, ...

Taxonomies are good for reminders, navigation & abstraction (DDP computes aggregate values)



DDP Matrices

Effects (Mitigation x Risk)

		FMs	[-]Product Engineering					
		FMs	[-]Requirements Risks					
		FMs	Stabilit	Compli	Clarity	Validity	Feasib	Pre
PACTs	PACTs	FoM/R	0.5	0.5	0.5	0.5	0.5	0.5
	Authori	7.95	0.1	0.1	0.1	0.1	0.1	0.3
	Identify	2.3						
	Mainta	0						
	Softwa	2.65						
	Implem	1.85	0.9	0.3	0.9	0.9	0.3	0.3
	Manag	0.15						
	Docum	1.65	0.3	0.9	0.9	0.1	0.3	0.3
	Peer	2.8	0.9	0.9	0.9	0.9	0.9	0.9

numbers
supplied by
experts and/or
based on
accumulated
metrics

proportion of
Risk reduced
by Mitigation

Impacts (Objective x Risk) are similar:
proportion of Objective loss if Risk occurs

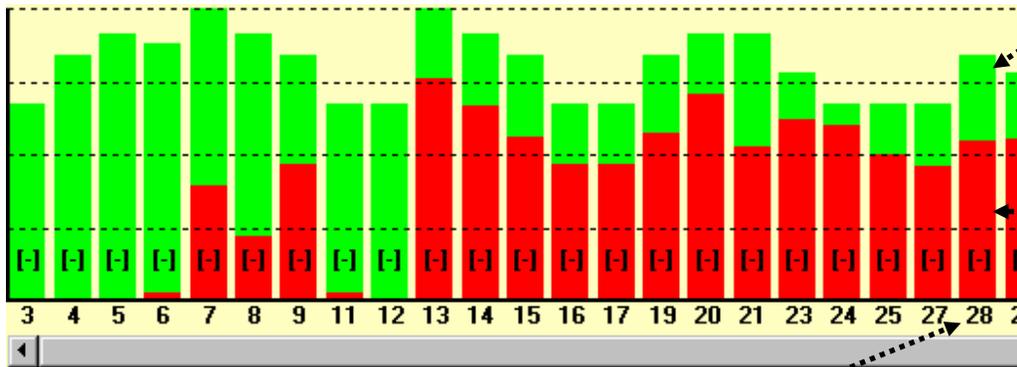


DDP Visualizations: Bar Charts



Risks bar chart

Unsorted – order matches leaf elements in Risk tree

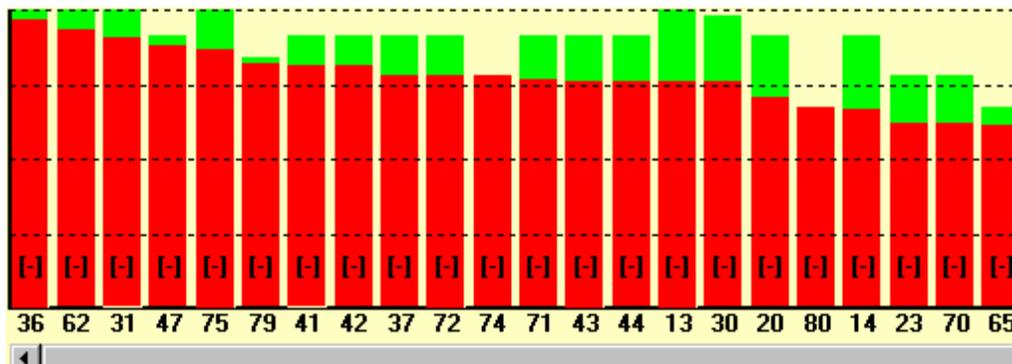


Item number in tree

Green: of this Risk's total Impact on Objectives, that *saved* by Mitigations

Red: of this Risk's total Impact on Objectives, that *remaining* despite Mitigations

Sorted – in decreasing order of remaining Risk



Objectives bar chart similar – how much each is impacted

Mitigations bar chart similar – how much impact each is saving

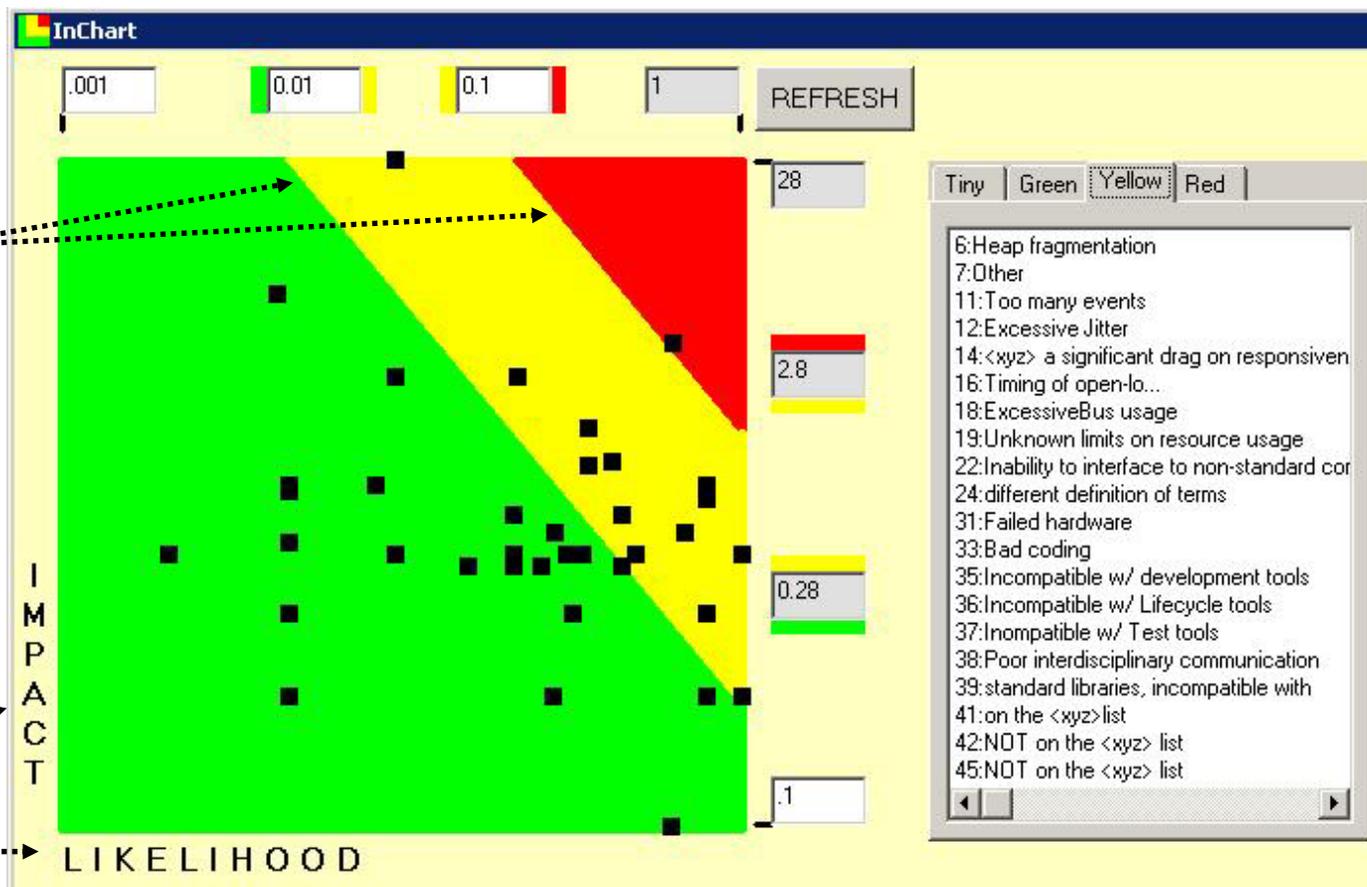


Risk Magnitude = Likelihood x Impact (Severity)

User defines risk levels demarking red/yellow/green/(tiny) risk regions

Log/Log
scale:
diagonal
boundaries
= risk
contour
lines

Conventional
measure of
risk
as impact
(severity) x
likelihood.





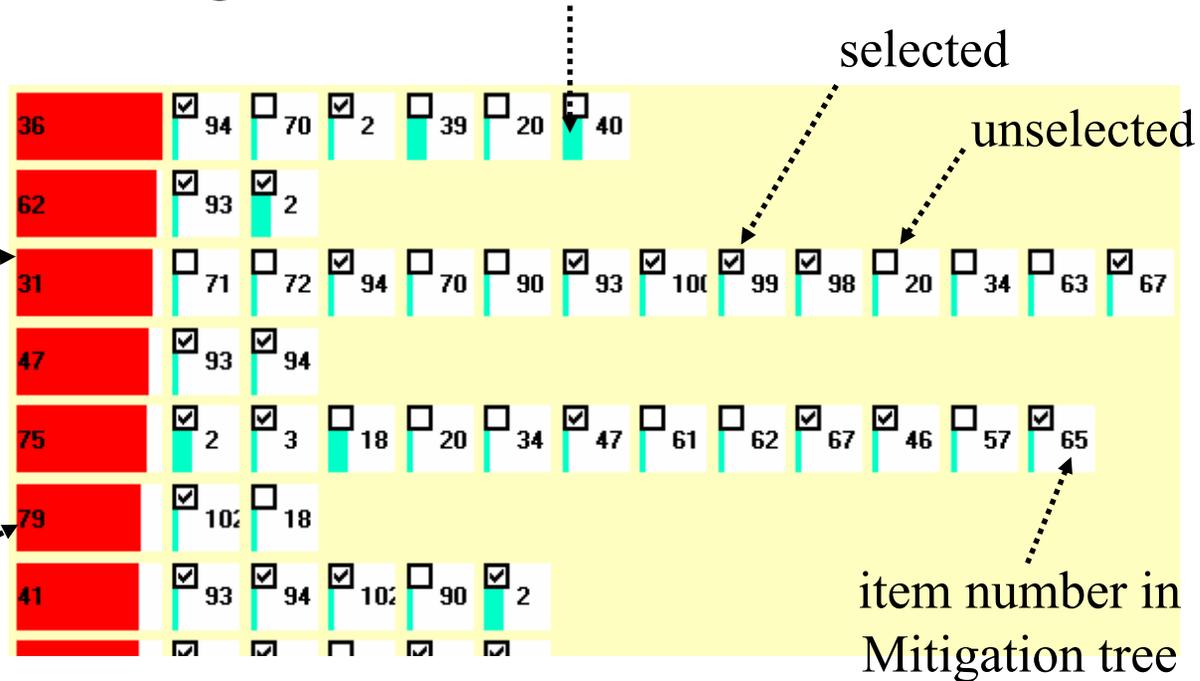
DDP Visualizations: Stem-and-Leaf(*) Charts

E.g., Risks
& their
Mitigations

Mitigations – turquoise width \cong effect

Risks – red
width \cong log
outstanding
 Σ impact

item number
in Risk tree



(*) Tufté attributes these to John W. Tukey, “Some Graphical and Semigraphic Displays”
Their usage was introduced into RBP by D. Howard, extended further by us in DDP.

Compact visualization of DDP's sparse matrices



Cost/Benefit Refinements



Mitigations grouped into phases (e.g., requirements, design, coding, ...)

- _ Match spending with budget profile
- _ Implies risk reduction by phase: compute risk reduction profile

Mitigation subtypes

- _ *preventions*: decrease likelihood of problem arising (e.g., training; coding conventions)
- _ *alleviations*: decrease severity of problem if it occurs (e.g., defensive programming)
- _ *detections*: imply need to repair problems so detected (e.g., testing; analysis)

Cost of repair separated from cost of detection

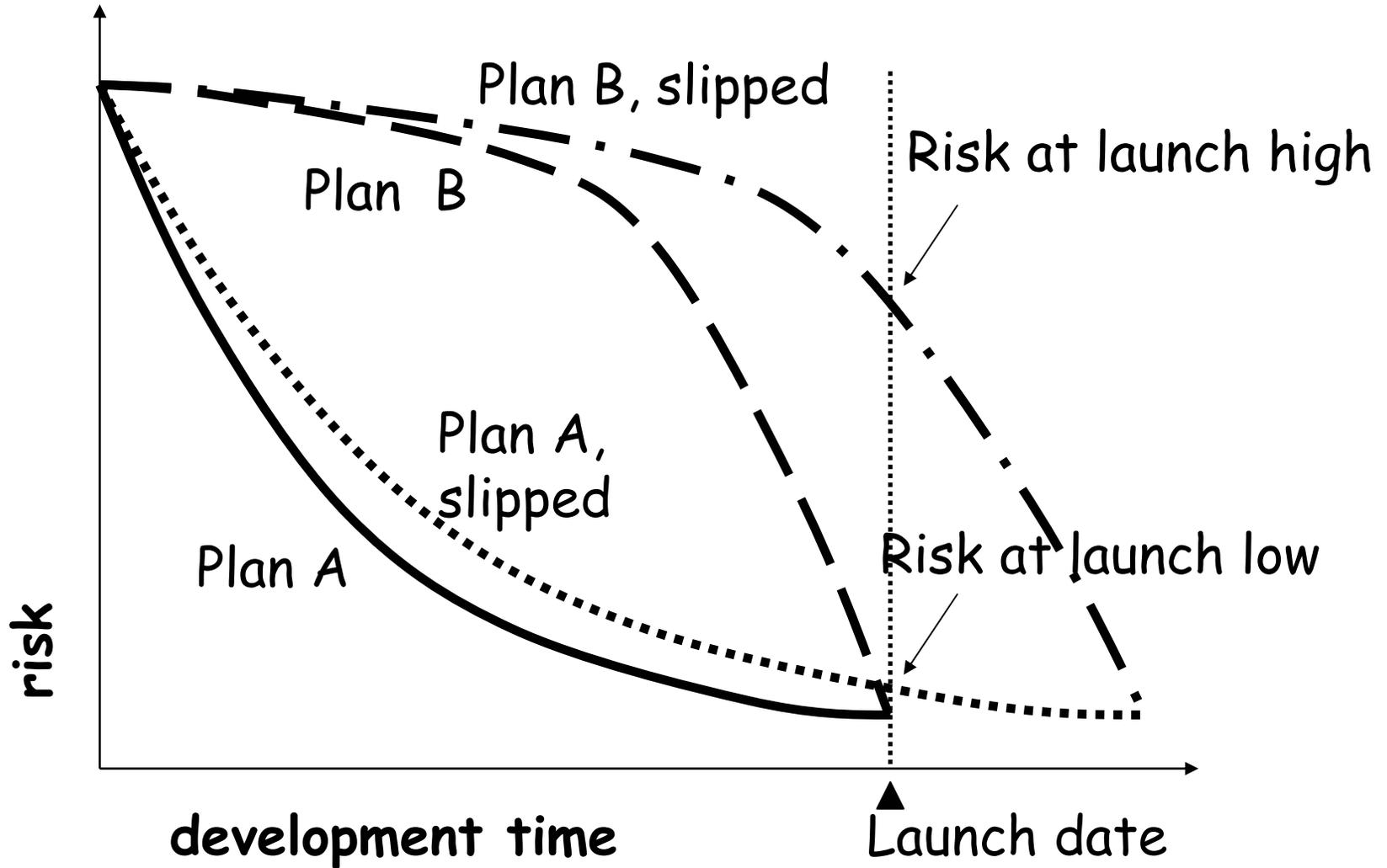
- _ repair costs typically escalate greatly over time
- _ reveals net savings of up-front effort

Mitigation induced & aggravated failures

- _ software bugfix introduces new bugs
- _ turning on/off array bound checking changes timing



Risk Reduction Profile





Optimization

Typical model had 99 Mitigations, i.e., 2^{99} (approx 10^{30}) possible solutions (choices of Mitigations to perform).

Discrete choices (perform/not perform), so few traditional optimization methods apply

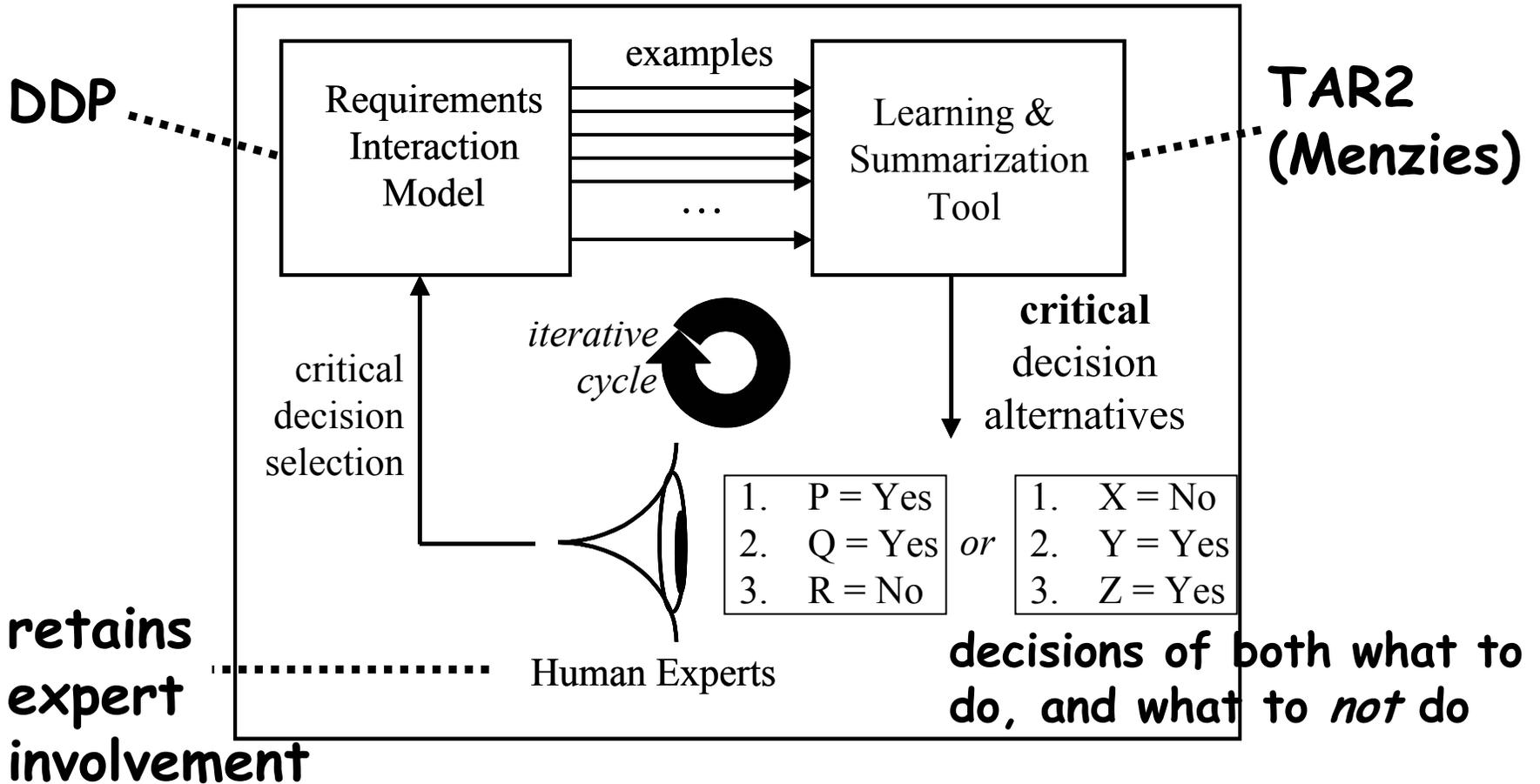
Bad enough with simple cost/benefit model - harder yet as model becomes more complex

Promising Solutions:

- **Genetic Algorithms** (a form of heuristic search):
promising results on simple DDP cost/benefit model
- **Machine Learning based approach of Menzies:**
pilot study results good
method also identifies *critical* decision points
- **Simulated annealing:** fast convergence, simple to use;
now packaged as part of DDP tool distribution



Optimization Using Menzies' (*) Machine Learning based approach

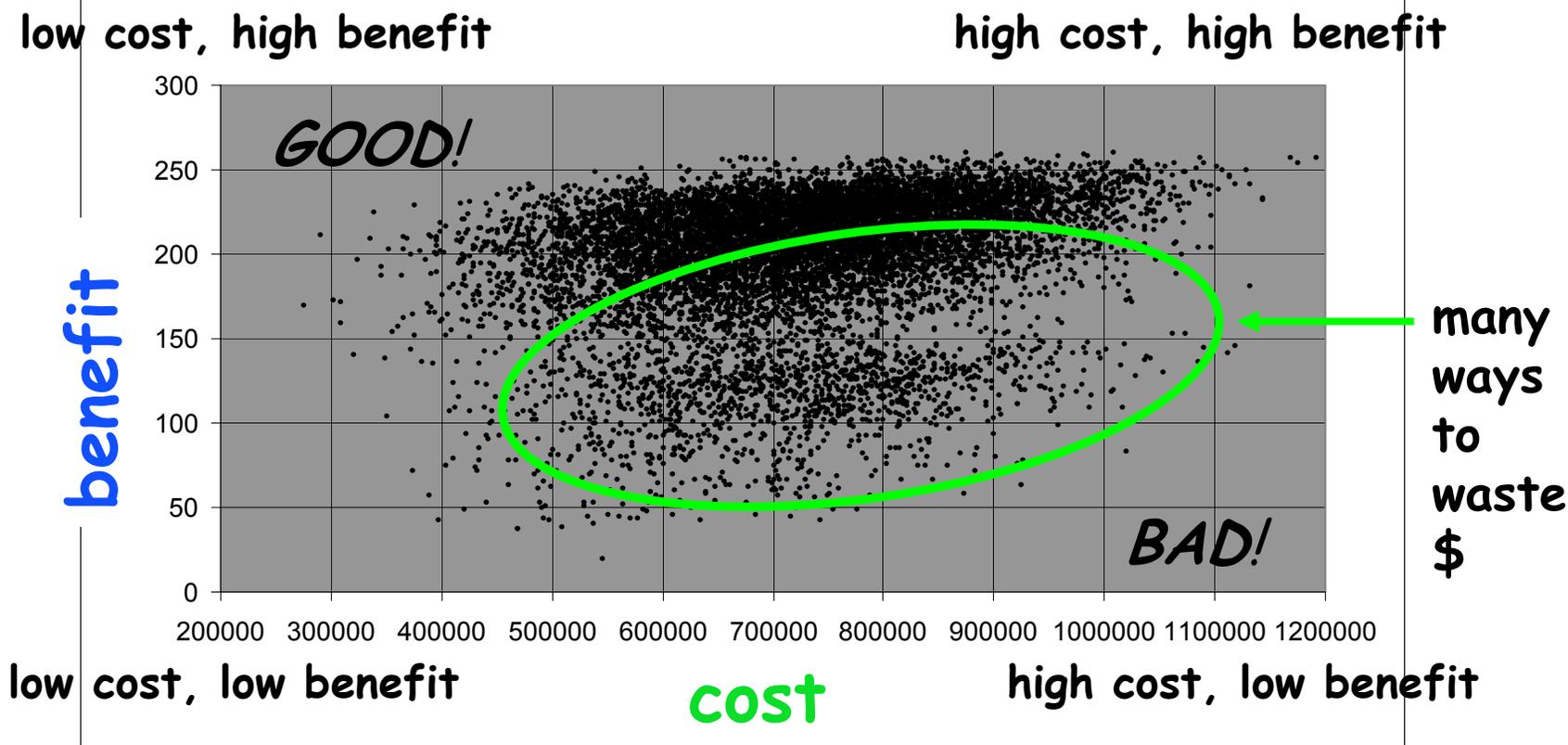


***<http://tim.menzies.com>**



Dataset before Optimization **JPL**

California
Institute of
Technology



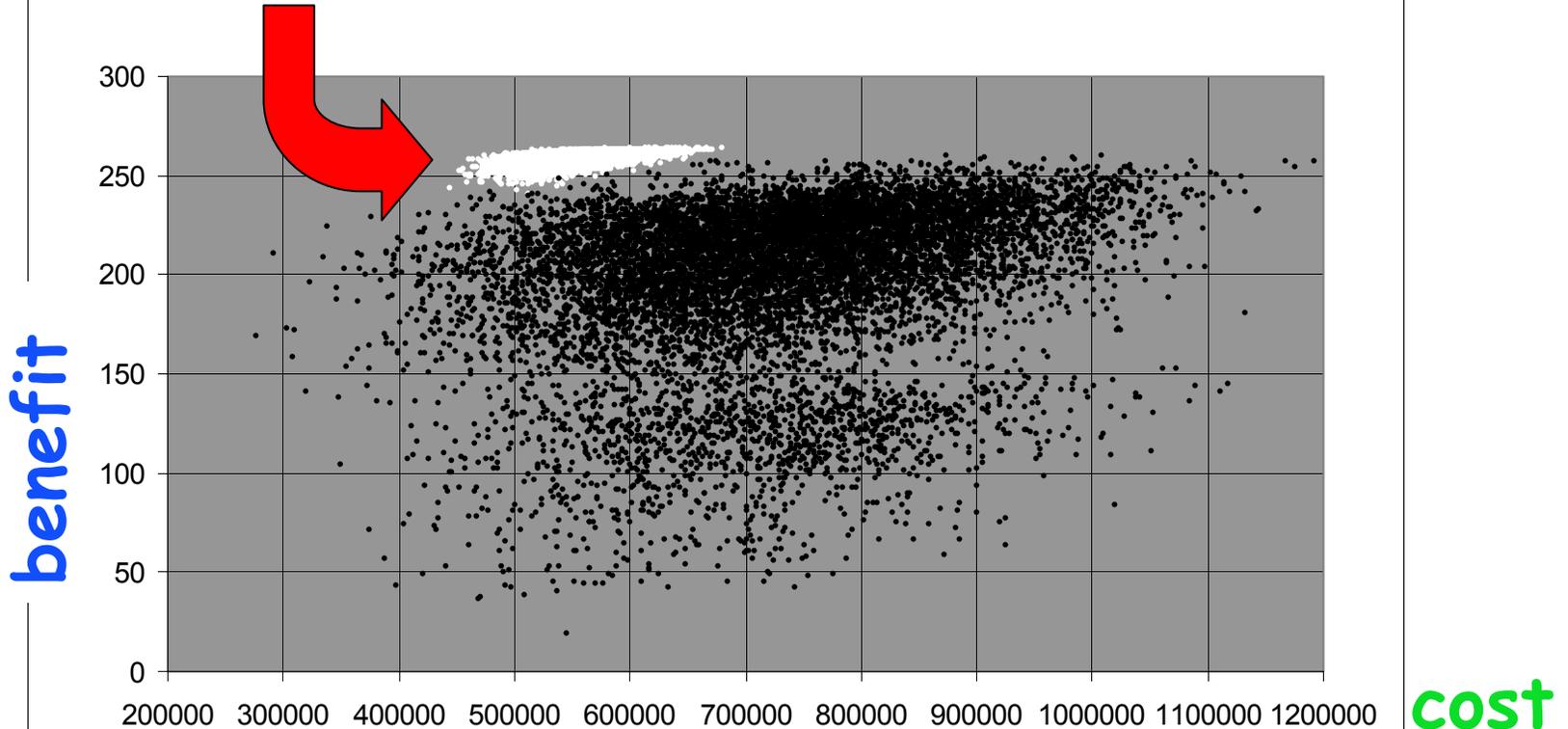
Each black point a randomly chosen selection of dataset's assurance activities. DDP used to calculate **cost** and **benefit** of each such selection.



Dataset *after* Optimization

California
Institute of
Technology

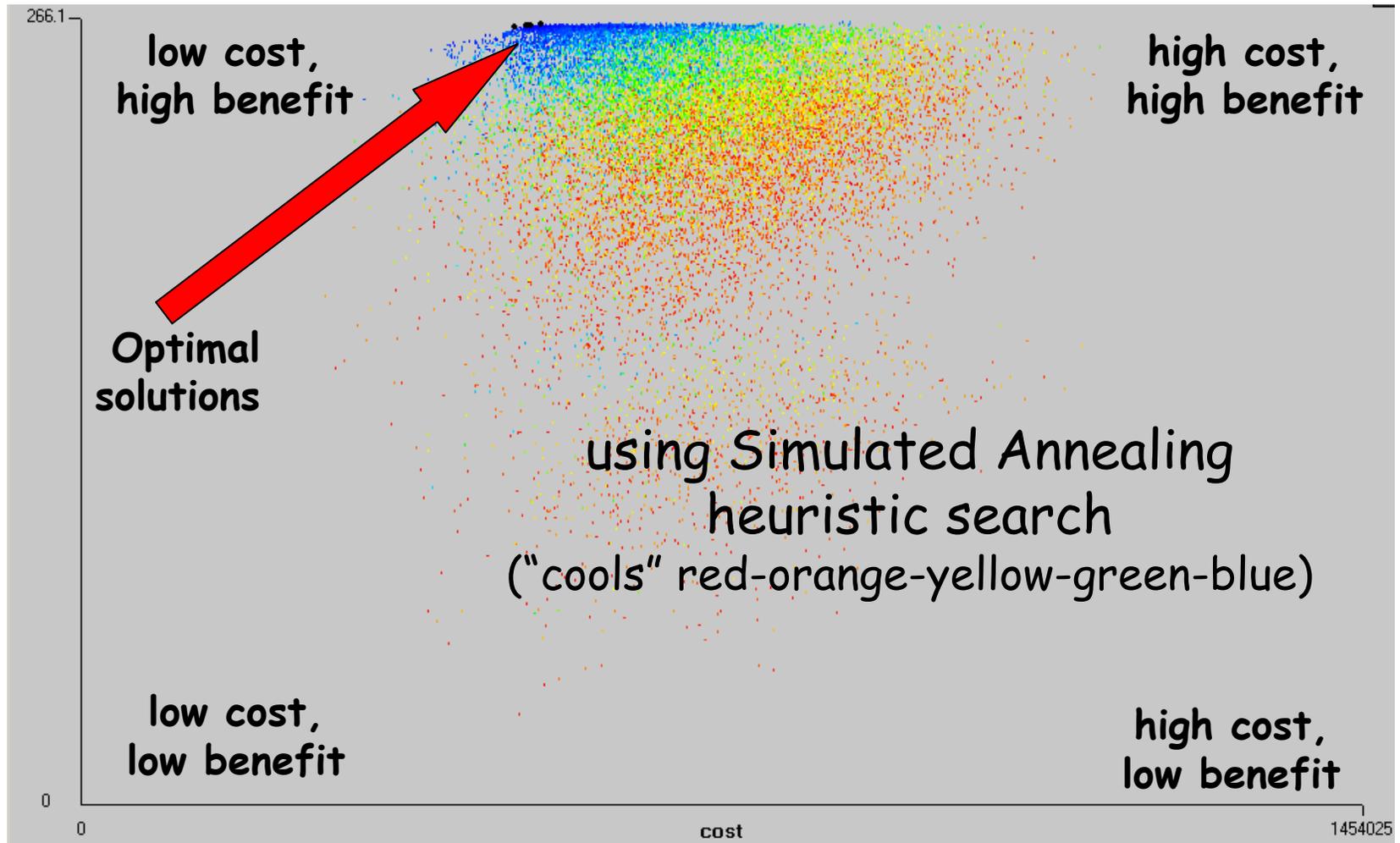
Each white point is an optimized selection of dataset's assurance activities (33 critical ones are as directed by TAR2, other 66 chosen at random).



Menzies' TAR2 identified 33 most critical decisions:
21 of them assurance activities to perform
12 of them assurance activities to *not* perform.



Simulated Annealing now part of DDP tool





DDP Sensitivity Analysis

- 1) Menzies' technique showed optimal solution robust
- 2) Vary effect values one by one, recompute requirements attainment, tabulate results:

Change	% Change	PACT	Failure Mode
-2.02	0.76	Select/make laser	Insufficient power
-0.834	0.314	CCD Qualification	CCD degradation
-0.6	0.226	System Study	Other technologies are better
-0.329	0.124	Hermetic packaging	non-Hermetic
-0.246	0.0926	Fibre qualification	Fibre degradation

- 3) Use results for relative decision making, *not* as absolute measures of reliability.
Having identified areas of critical concern, apply other techniques (e.g., probabilistic risk assessment).



Software Engineering Community **JPL**

Starting Points

California
Institute of
Technology

Risks: Software Risk Taxonomy (SEI)

Mitigations: two datasets:

1. CMM Key Practices (Infrastructure and Activities)
2. Software Quality Assurance activities from Ask Pete (NASA Glenn tool)

Effects: cross-linkings of the above

1. Expert's best estimates of *which* help
2. Experts' 1000+ best estimates of *how much* (quantified effectiveness) they help

Note: Objectives are PROJECT SPECIFIC

**Seeking experience-based data
(e.g., from CeBASE consortium)**



V&V Selection is an Assurance Optimization Problem

The selection of assurance activities such that:

For a **given set of resources**
(time, budget, personnel, test beds, CPU, memory, ...)
benefits are maximized

or

For a given **set of objectives**
(science return goals; on-time and in-budget
development; 99+% expectation of successful landing)
costs are minimized.



For Further Information on the DDP quantitative risk model and tool support



<http://ddptool.jpl.nasa.gov>

Steven L. Cornford@Jpl.Nasa.Gov

Martin.S.Feather@Jpl.Nasa.Gov