Parametric Cost Analysis for Evolving Technologies

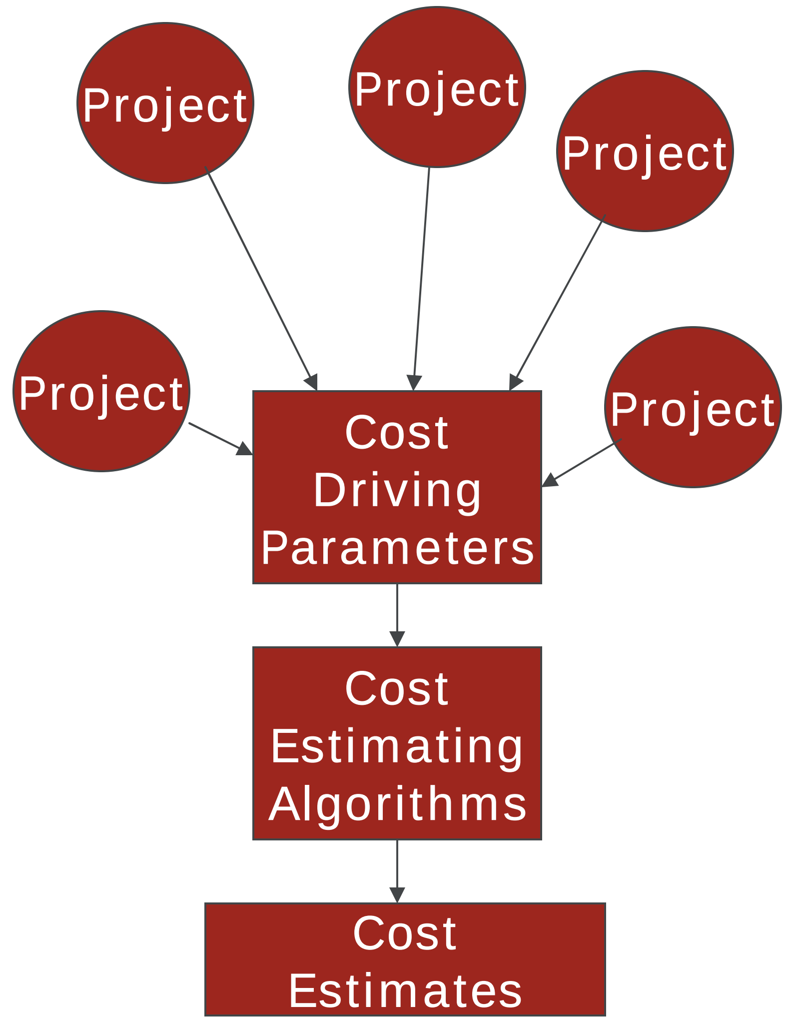
By Evin Stump

Senior Systems Engineer

Galorath Incorporated

The basic process of parametric cost analysis of technologies

- Identify projects involving the technology of interest and collect relevant cost and related data from them
- Identify the likely cost driving parameters
- Create cost estimating algorithms
- Use the algorithms to estimate future costs



Harsh realities

- For some technologies, there are few projects
- For competitive and other reasons, data from many of them may be unavailable

If you were asked to estimate a follow-on to the Hubble telescope. how would you go about it? What you would do is probably similar to what we show in this presentation.

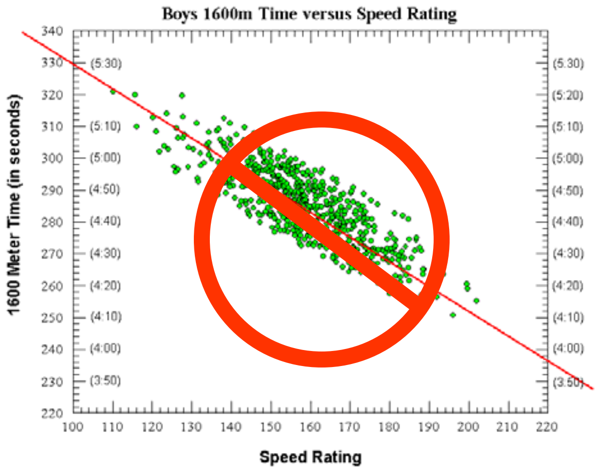
More harsh realities

SEER® by GALORATH

- Some technologies of high interest evolve rapidly
- Current costs may differ significantly from costs as recently as a year ago…why?
  - Changes in customer needs
  - Changes in aggregate demand
  - Increased competition
  - Innovations in tools and practices
  - Breakthroughs in the technology
  - New cost drivers; changes in relative importance of existing cost drivers.

Realities of parametric methods

- Typical parametric methods assume a relatively homogenous sample
- They also assume a relatively large sample
  - Absolute minimum, one data point per cost driver
  - Acceptable minimum about three data points per cost driver (some say more!)
- Accurate parametric models for advanced technologies seldom need fewer than three cost drivers (i.e., at least 9 data points)
- Some need 10 to 30 or so (30 to 90 data points!)



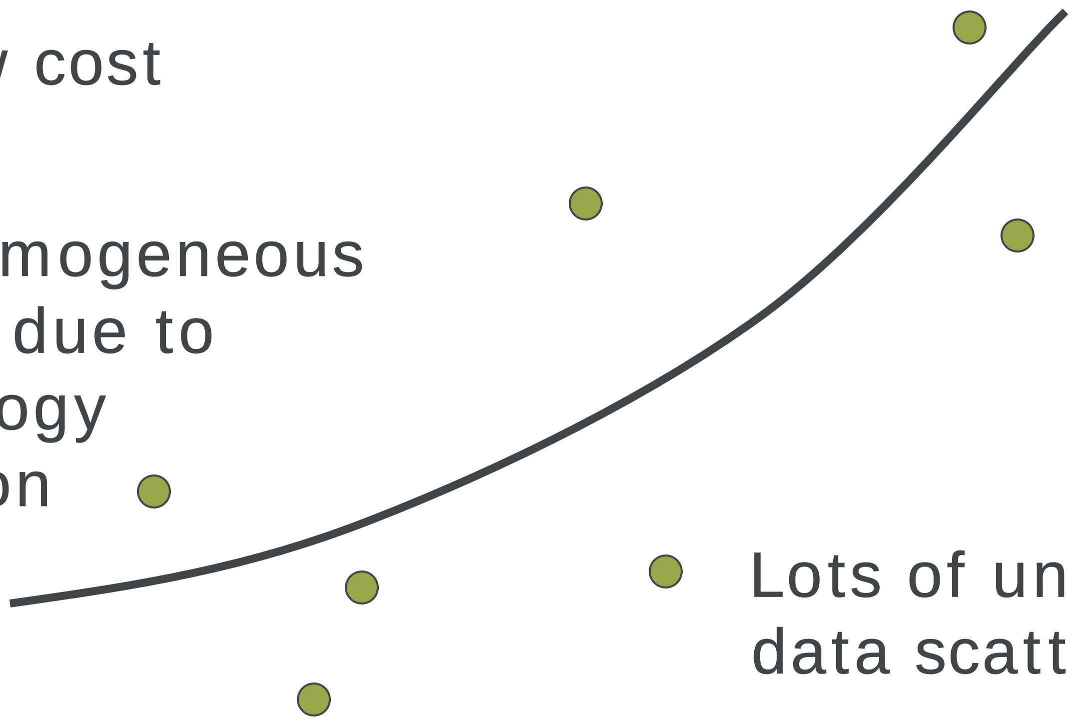
There may not be that many in the whole world!

Failures of parametric assumptions

Too few data points

Too few cost drivers

Non-homogeneous sample due to technology evolution



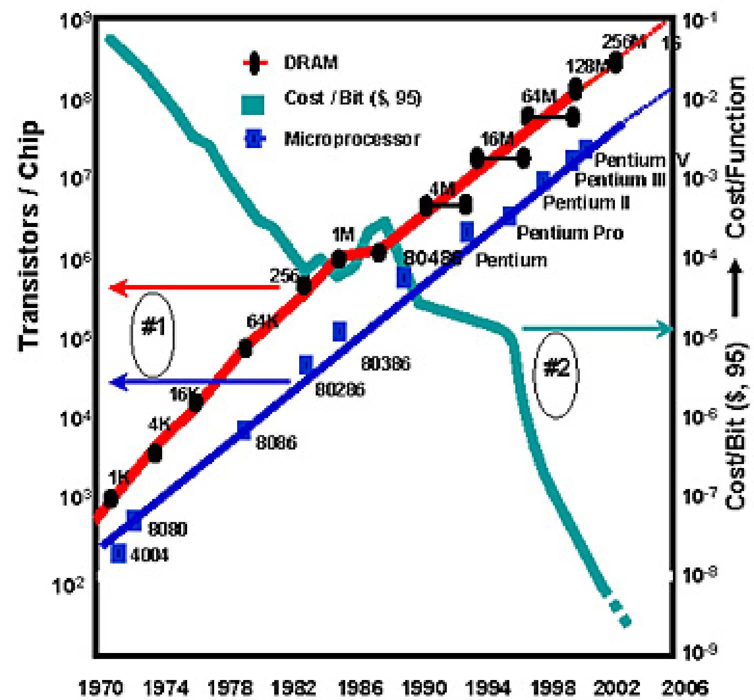
Lots of unexplained data scatter; low R^2

Estimating uncertainty

Less than useful cost model

Galorath's solution: evolutionary, focused data* parametric models

- If technology can evolve, why can't a cost model?
- It can, and has
- Prime exemplar: Our Spyglass plug-in to SEER-H
- Parts of our SEER-IC model are also suited to our evolutionary approach
- In this disclosure, we focus on Spyglass and its development



* More on this later!

What Is Spyglass?

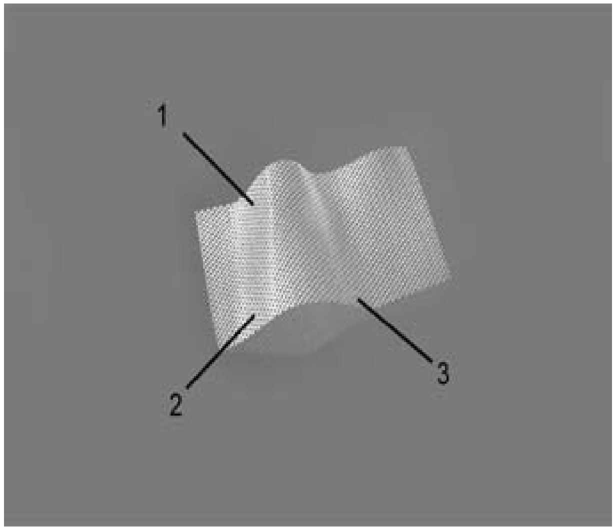
- Spyglass is a SEER-H plug-in
- It helps SEER-H make more accurate development and production estimates of space-based electro-optical systems (EOS)
- Currently it is being upgraded to estimate manned and unmanned aircraft and missile EOS applications
- In its next release it will estimate over 50 different EOS technologies in five platforms

Issues to be dealt with

- Limited initial data
- Selection of cost drivers
- Model architecture
- Accommodating technology change
- Keeping track of older technologies

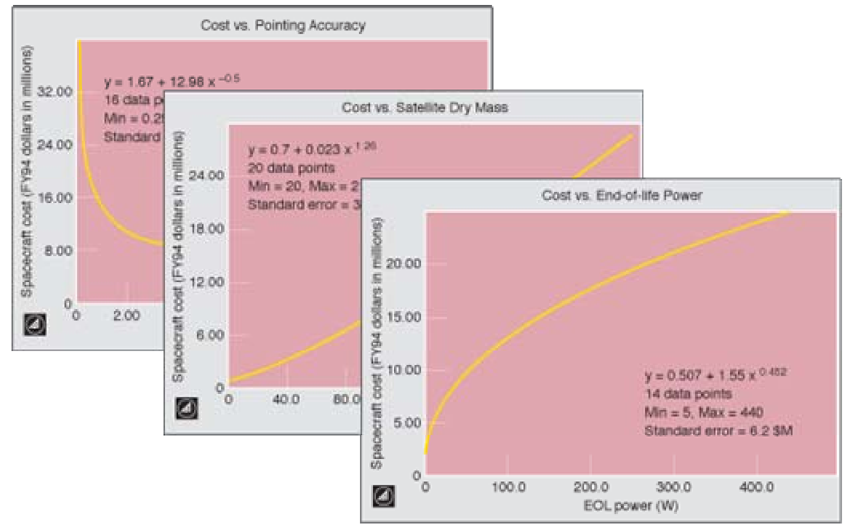
Limited initial data

- Our first (traditional) efforts to find data for Spyglass were not successful
- Our solution was to engage an EOS consulting company
- They had access to both current technology and sanitized cost data
- They also had high expertise in the "why" of costs
- For most technologies of current interest they could provide only one or two data points, but the data was of high quality, recent, and comprehensive



Selection of cost drivers

- SEER-H relies heavily on weight to get costs into the right ball park
- We decided to go down a new path with Spyglass
- We called our cost drivers Key Technical / Performance Parameters (KTPPs)



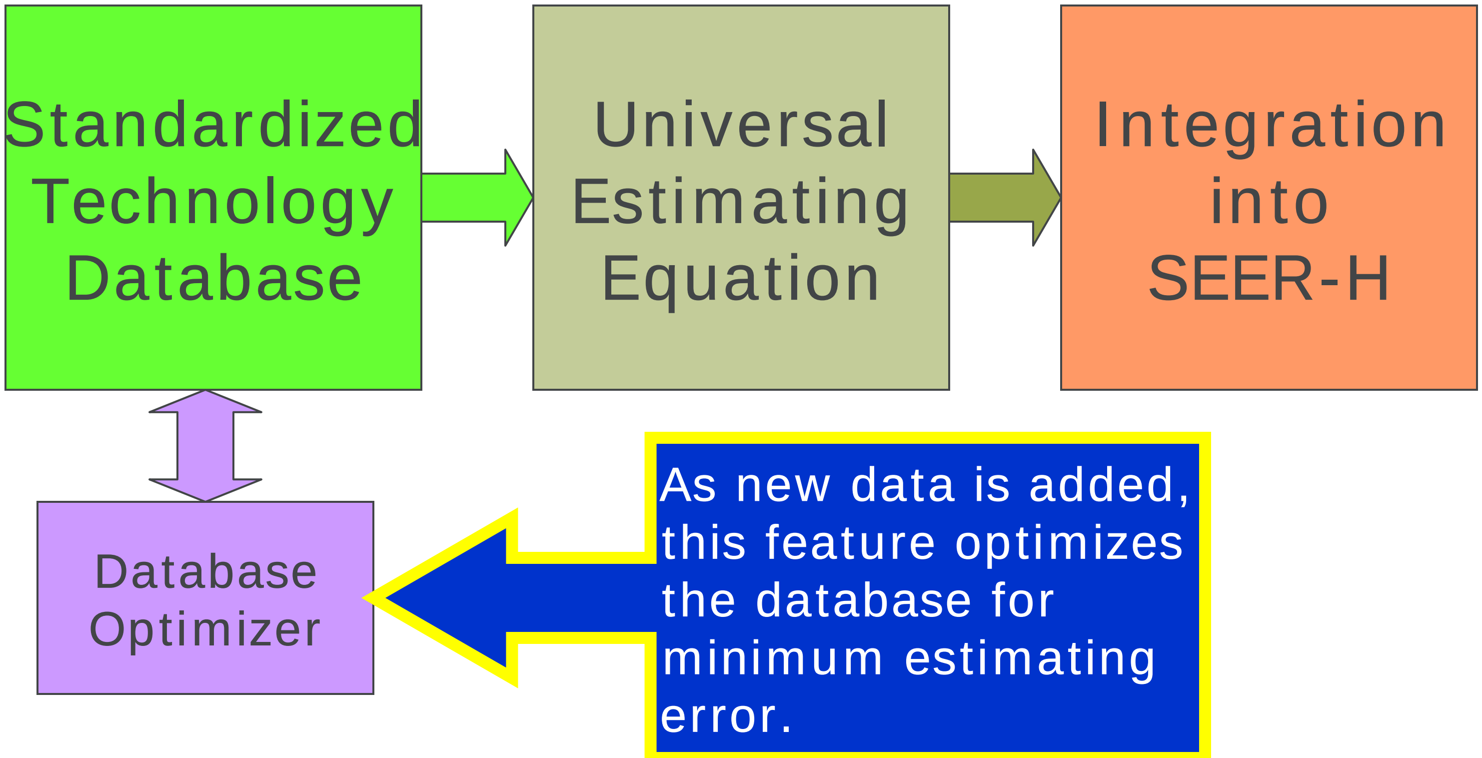
Examples:

Array size; pitch for detectors

Diameter; imaging elements for telescopes

Power consumed, delta temperature for coolers

Model architecture



This architecture simplifies Spyglass and eases model evolution and addition of new technologies.

Universal Estimating Equation

$$COI = E * S * \left[w_{PR} * f_{PR}() + \sum_{i=1}^{N} w_i * f_{PR}() * f_i() + \sum_{j=1}^{M} w_j * f_j() \right]$$

Cost of Interest

Prime exemplar raw cost

Normal-ization factor (to standard condition)

Influence function weights (AHP)

Influence functions

Initial weights are selected by experts using AHP. Influence functions are simple analytic functions fitted to plot points provided by experts. The prime exemplar raw cost is from the most representative, carefully studied project for the technology of interest. We call this focused data -- getting the most possible from one project.

Influence function defined

- A continuous influence function mathematically expresses, for a given technology, the average effect of a KTPP on cost across the KTPP's design range, relative to the prime exemplar
- A discrete influence function expresses, for a given technology, the cost effect of a discrete design option, relative to the prime exemplar

Creating influence functions

SEER by GALORATH

Technology KTPP

Best fit coefficients

Applicable cost (development material)

Best fit function

Area InGaAs					
Array Size		DMC	y=a*b^x*x^c		
a	b	c	d	Hoerl	
6.1402792000E-01	1.0000002000E+00	4.29524410E-02	NA		
Value	Influence	Xinfluence	Function	Inverse	Error %
2.62000E+05	1.50	1.1	1.1057944711E+00	1.536748	2.45
1.63840E+05	1.20	1.045454545	1.0626521959E+00	1.28651	7.21
6.55360E+04	1.00	1	1.0017516655E+00	1.007031	0.70
1.02400E+03	0.50	0.833333333	8.2713575339E-01	0.486184	-2.76
6.40000E+01	0.30	0.730769231	7.3413092136E-01	0.305706	1.90

% fit error check

Bottom of design range

Prime exemplar

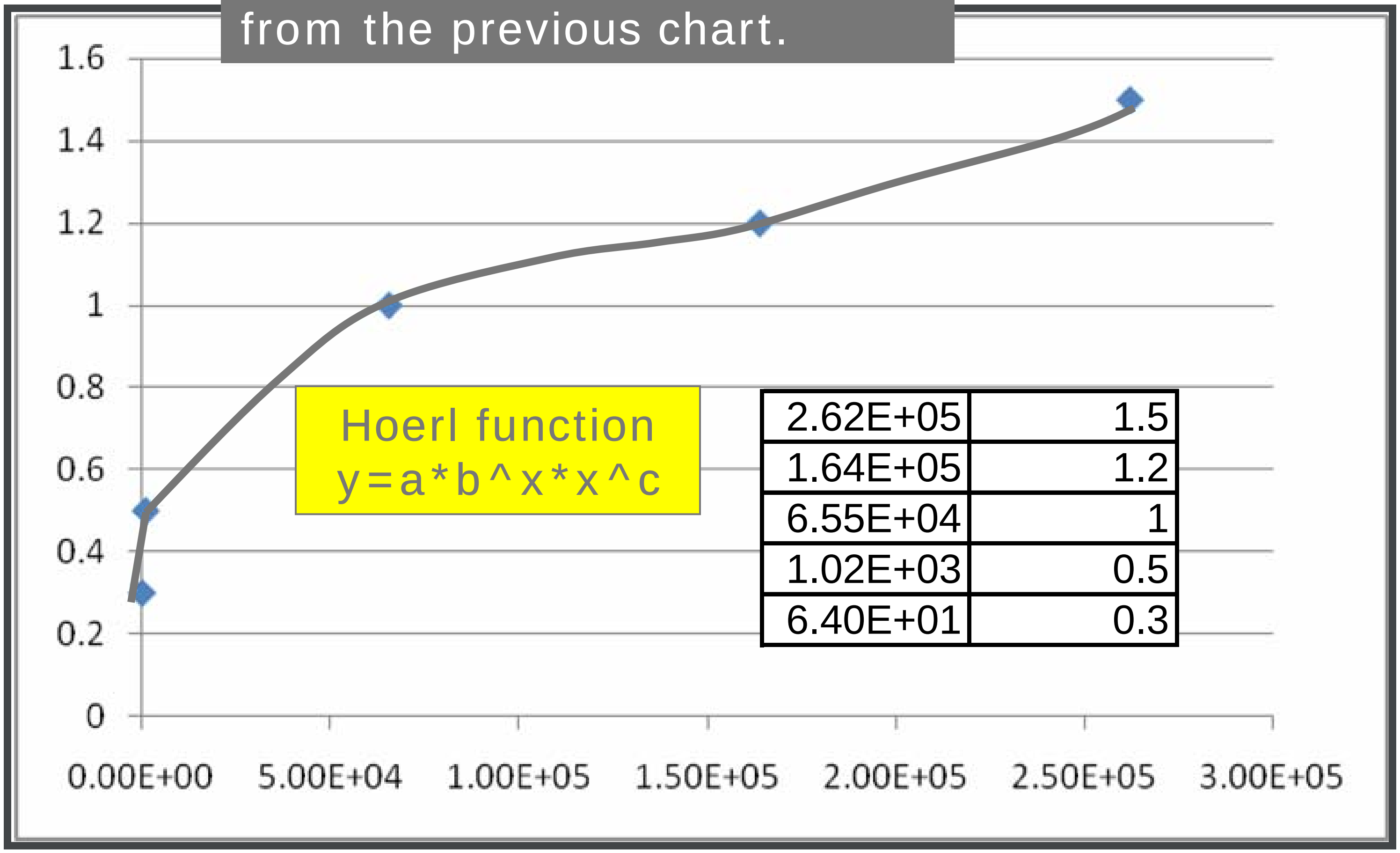
Top of design range

Influence values selected by experts

Modified Levenberg-Marquardt algorithm (selects best fit function & function coefficients)

Influence function example

This is the influence function from the previous chart.



2.62E+05	1.5
1.64E+05	1.2
6.55E+04	1
1.02E+03	0.5
6.40E+01	0.3

Role of expert judgment

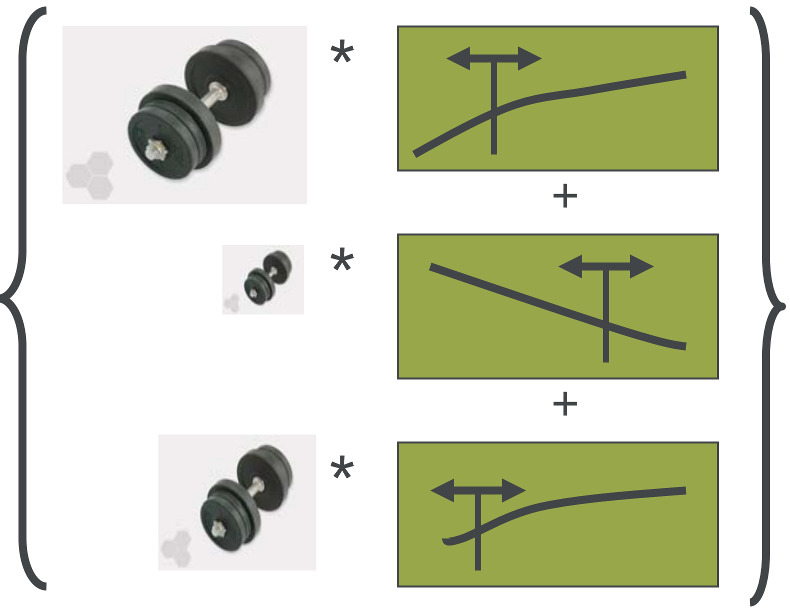
SEER by GALORATH

- Initial weights and influence functions are based on expert judgment
- We carefully control expert judgment to minimize lapses (AHP, multi-voting, modified Delphi, multiple credibility checks)
- The results have been excellent
- Still, project experience is the best teacher
- And, we want to capture evolutionary behavior of the technology

Using expert judgment

Database optimizer

COI = E*S*



This arrangement estimates the prime exemplar technology perfectly

But how much error will it have for another historical project? Probably not zero!

What set of weights and influence functions shapes will minimize the average error across both projects? Across any three projects? Any four projects?

The Database Optimizer finds that unique set

Accommodating technology change

- Some evolutionary changes are relatively small and the adaptive capability of the model can keep up with them
- Some are large and may require:
  - New KTPPs
  - New technology capability
- Either way, continuous monitoring of industry activity is a MUST!!

Keeping track of older technology

- Sometimes it is necessary to make changes to a technology so radical that a total replacement is needed
- But some users may still need to deal with the older version
- Spyglass maintains the older version "hidden" in its database

Summary

- Some technologies of high interest evolve rapidly
- Estimating them properly means that the cost model must keep up!
- Data is likely to be sparse at the leading edge of evolution
- Galorath's approach is the influence function method, making best use of expert judgment

Your comments are welcome.

See me later or contact me at estump@galorath.com.