

< ESTIMATION />

A Science, Not an Art?

.....

5-13-2015

John Nollin

Director of Operations

Project Management,
Process, Estimation,
Tools

Chicago, IL.



“The typical software organization is not struggling to improve its estimates from +/-10% to +/-5% accuracy. The typical software organization is struggling to avoid estimates that are incorrect by 100% or more.” -- Steve McConnell

What is this presentation about?

- Fundamentals (overlooked)
- Big Risk factors
- Selling your estimate (Team)
- 20%-100% over budget

What is the presentation NOT about?

- Specific estimates (content type)
- Magical formulas
- Fine tuning error from 20% to 5%
- Selling your estimate (client)



Agenda

1 What is the Problem with Software Estimation?

2 Avoiding Risk

3 Estimation Techniques





What is the Problem with Software Estimation?

The greatest discrepancy between what the client thinks, an executive thinks, and a developer thinks is often the definition of ESTIMATE



Estimates, Targets, and Commitments

1 Estimates

Prediction

- A **Prediction** of how long a project will take or how much it will cost

2 Targets

Statement/Desire

- A **Statement** of a desirable business objective

3 Commitments

Promise

- A **Promise** to deliver defined functionality at a specific level of quality and by a certain date



Estimates, Targets, and Commitments

1

Estimate

2

Target

3

Commitment

**Cut From
Phase**



Your best developer is not necessarily your best estimator





The Solution Architect

About your unicorn:

- Knows the most about Drupal
- Experience in
 - Ecommerce
 - Migration
 - Integration
 - Site building & Theming
- Understands business value
- Can magically debug anything

Things that may be missing:



How fast can a DEVELOPER do the work?



How much time will be spent training, instructing, reviewing, and doing oversight?



Avoiding unfounded optimism



What is the definition of a “GOOD” estimate?



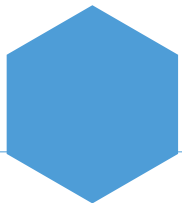
90% confidence



What is the surface temperature of the Sun? (Give a range in Fahrenheit)



Make your range wide enough that you feel you are 90% confident that your range includes the answer



10,000 Degrees F



After all, the process is called Estimation, not Exactimation.



- Phillip Armour



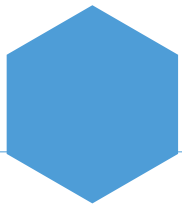
90% confidence



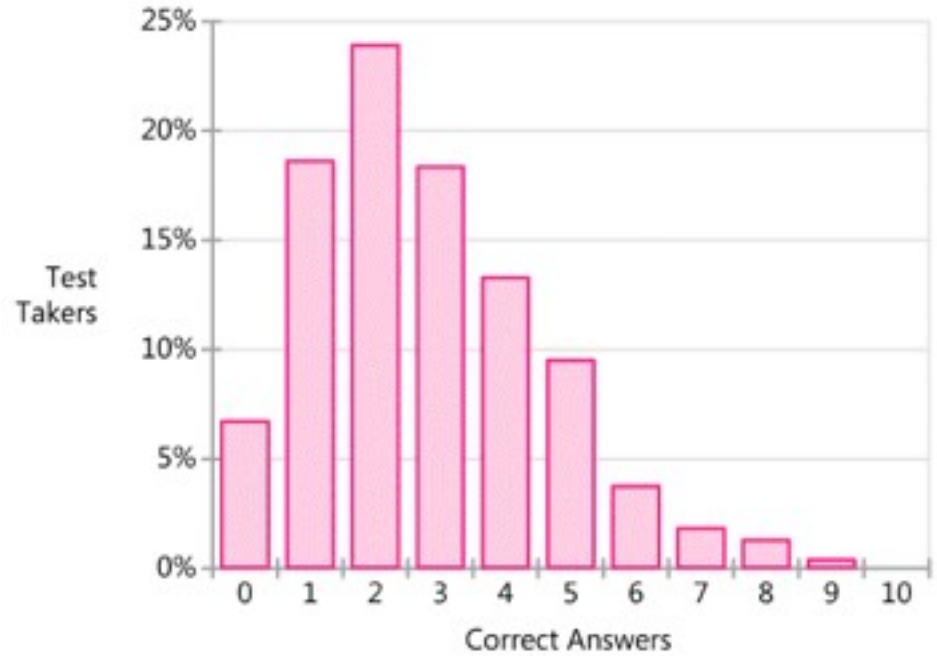
How confident is 90% confident?



The average number of correct answers was 2.8. Only 2% of quiz takers scored 8 or more correctly.



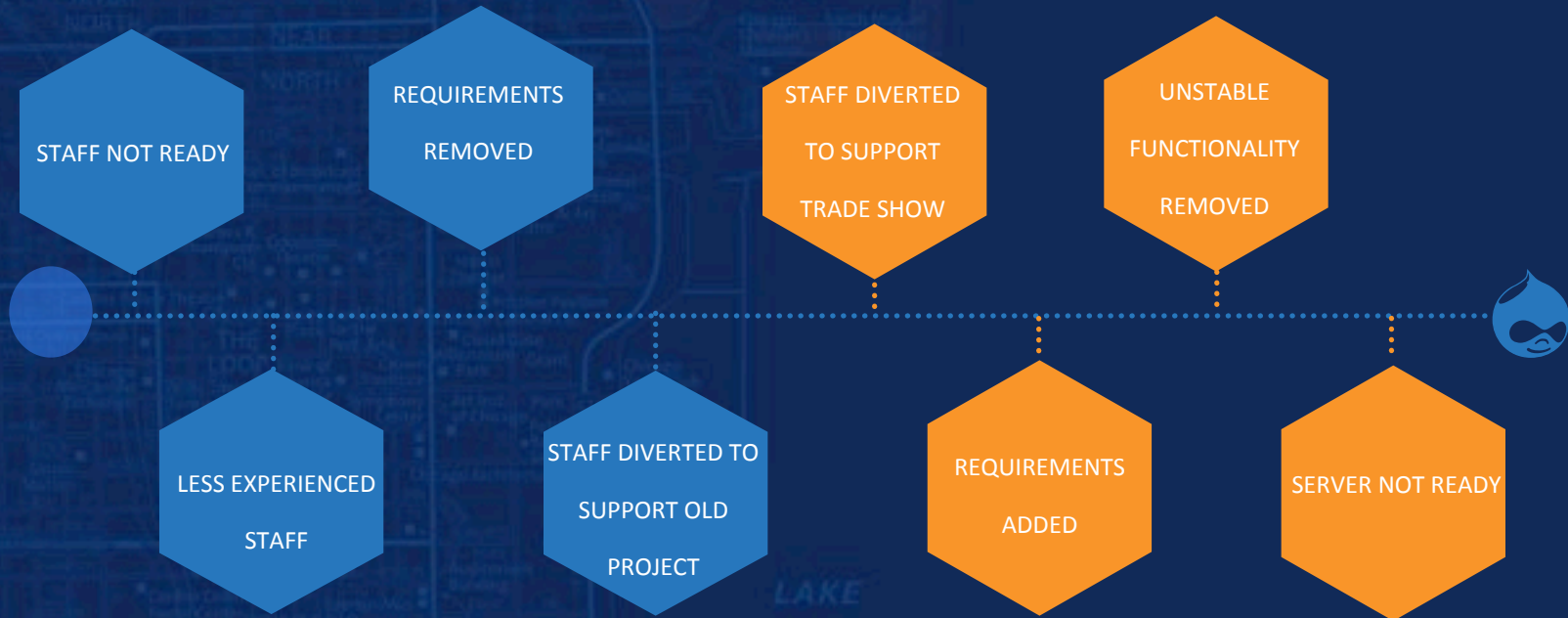
Most peoples intuitive sense of 90% confidence is in fact closer to 30% confidence.



Software is not developed in a vacuum



EXTERNAL FACTORS ARE OFTEN UNACCOUNTED FOR



Events that happen during the project nearly always invalidate the assumptions that were used to estimate the project in the first place.

1

**Functional
Assumptions**

2

**Staffing
Assumptions**

3

**Priority
Assumptions**

4

**Input
Assumptions**

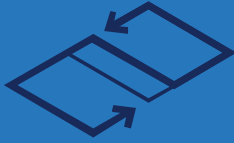


Software estimation is not just about adding up the sum of the parts

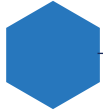


Project

Dynamics



The activities of a late or compressed project create destructive “late-project” dynamics that make the project even WORSE than nominal.



More
Status
Meetings

Frequent Re-
estimation

Apologizing to
key customers

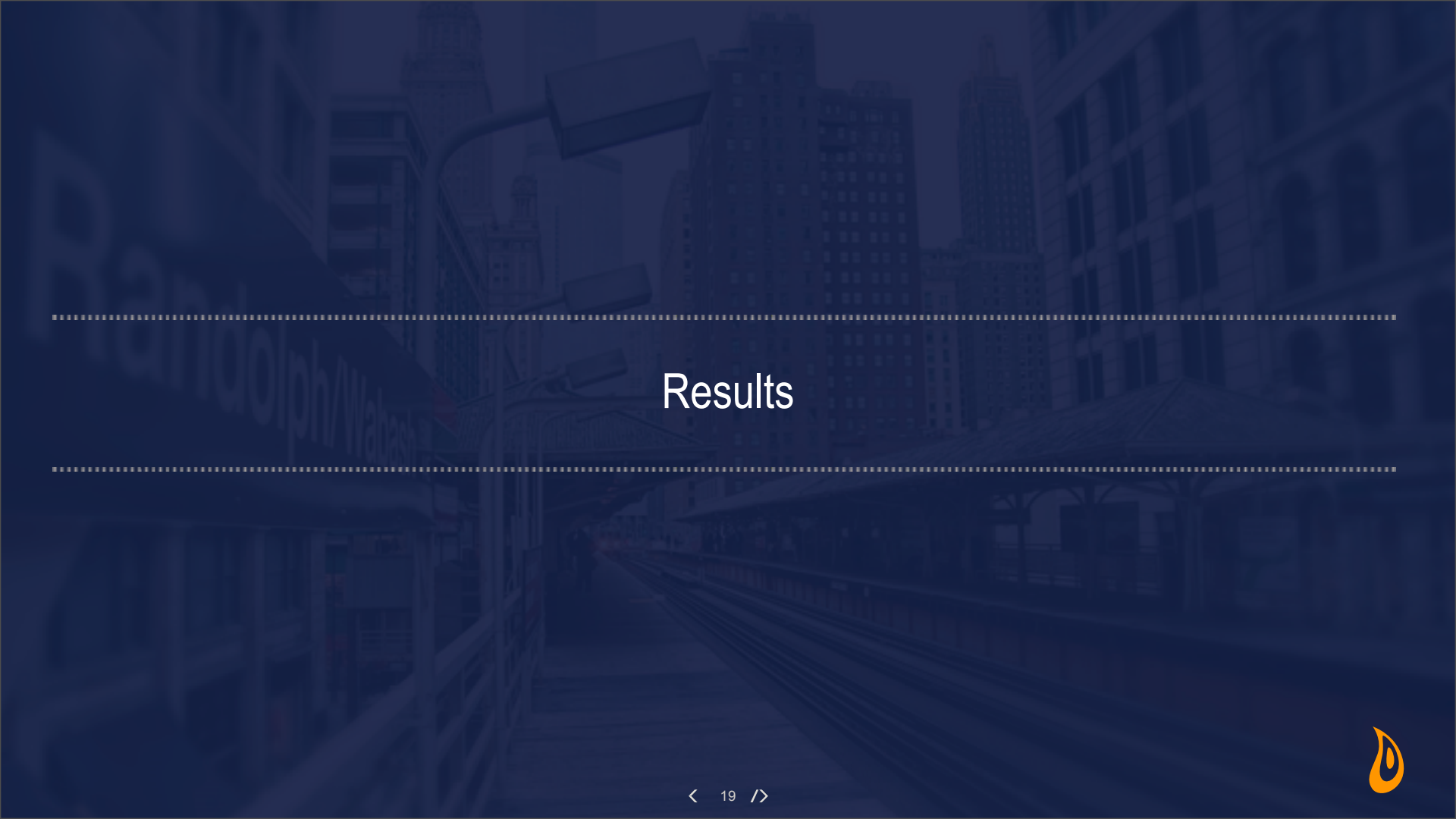
Preparing
interim
releases

More
discussion on
priority &
requirements

Fixing
problems that
arise from
quick and
dirty
workarounds

Going back to
the contract



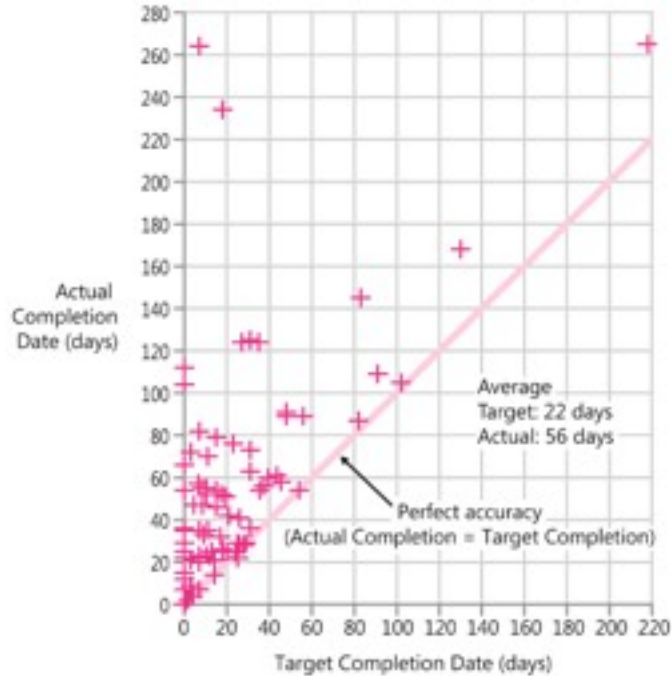


Results

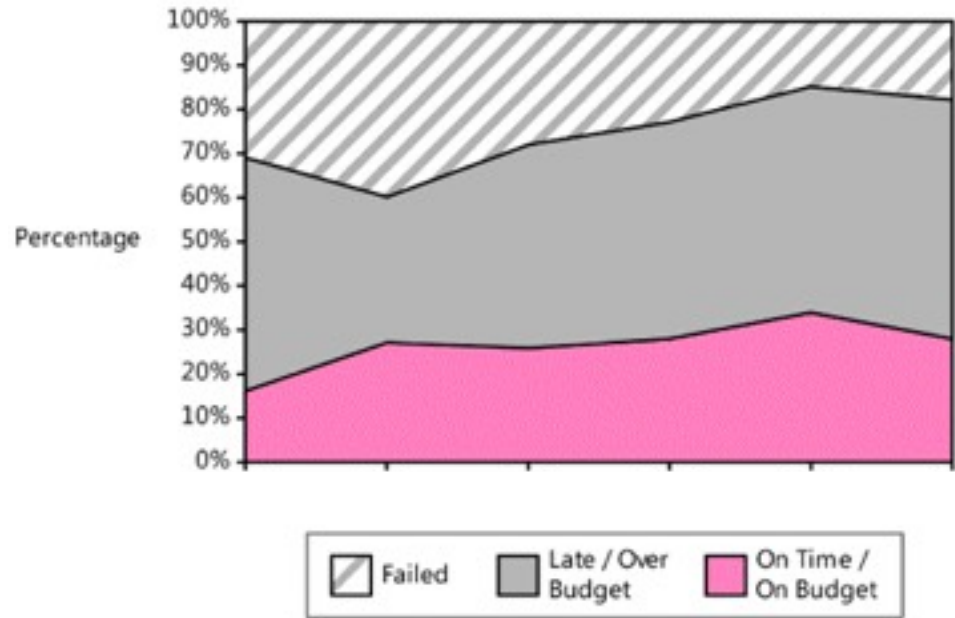


Results

Timelines



Budgets



Results

Costs



“Software does not have a neutral estimation problem. The industry data shows clearly that the software industry has an underestimation problem.”

-- Steve McConnell



When given the option of a shorter average schedule with higher variability or a longer average schedule with lower variability, 8 out of 10 executives will choose the second option

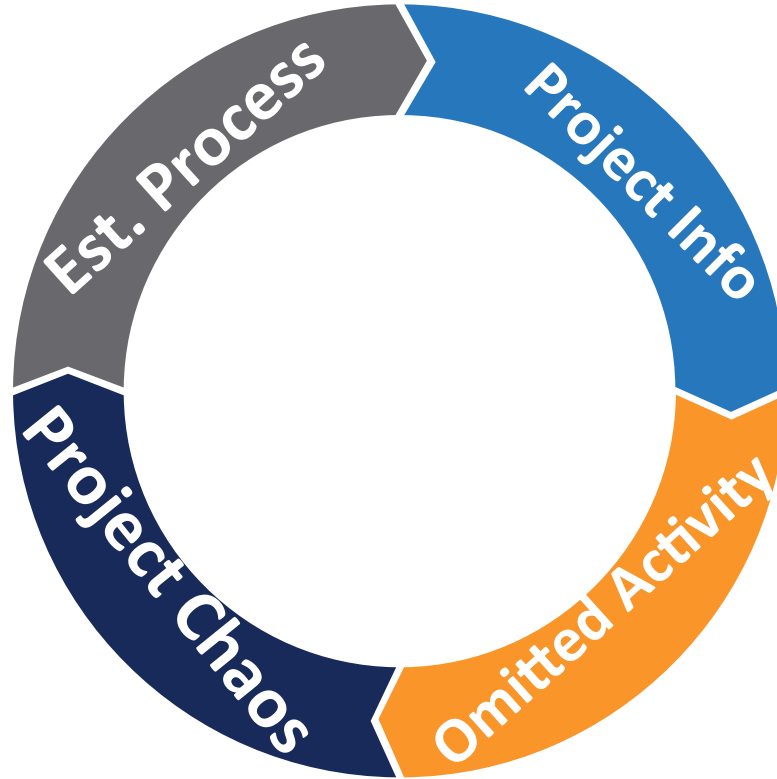


Avoiding Risk

Major Areas of Risk

- 90% confidence
- Best Estimator
- Terminology
- External Factors

- Lack of end-user involvement or production owner
- Timeline
- Poor wires/design



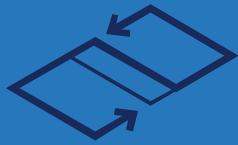
- What kind of content type/field/thing
- Scope w/o specs
- Specific definitions

- Setup Costs
- Management Costs
- Other (non-dev) Costs

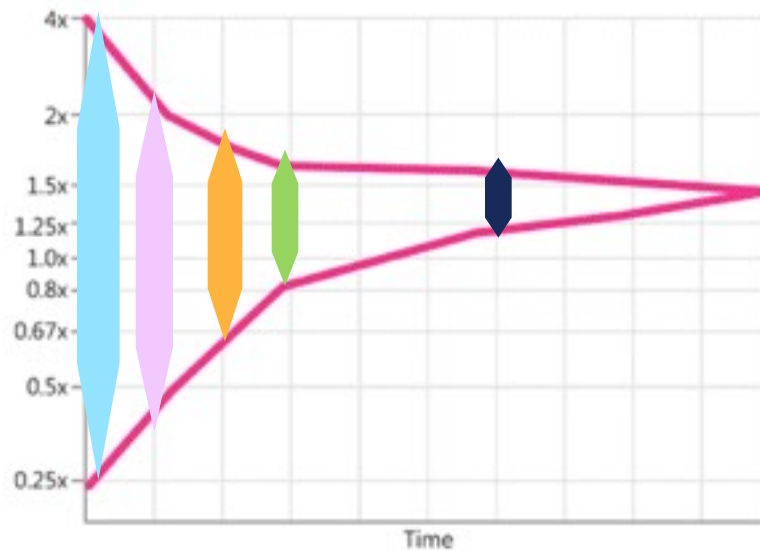


Consider your project's current position within the cone of uncertainty, this will help you gauge the level of accuracy in your estimate.





The Cone of Uncertainty



Initial Product Concept

"I want a website"



Product Definition

"I know the scope of services"



Requirements Complete

"I know the specs of the project"



User Interface Design Complete

"I have wireframes"



Detailed Design Complete

"I have full documented specs and design files"



Leverage the power of well defined products to move your way through the cone of uncertainty



Products

Drupal Concept Education as an entry into the product Discovery Process

Training Workshops



Evaluation of the inner workings of a current Drupal site, for upgrade and migration planning

Technical and UI/UX planning for a large scale development projects



Know when you are in a chaos project and how to reset



Chaos Projects

How does BAD go to TERRIBLE?

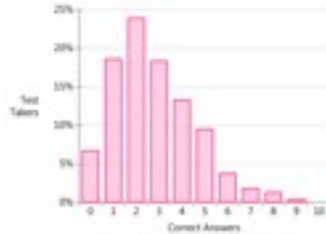
External
Factors



Poor Project
Dynamics



Re-estimation
or Planning



Pure Chaos



“Chaos projects introduce variability for each following step. The answer is not to continue to re-estimate but instead address each issue through project control.”



NEVER, let your estimator forget (or your sales team remove) these important activities



Omitted Activities

Setup



- Setup, Installation, Downloads, Builds
- Ramp-up, Kickoff, Onboarding, Accounting

Maintenance



- Updates, Upgrades, Deployments, Performance
- Configuration, Automation, Reviews

Non-software



- Vacation, Holiday, Sick, Drupalcon
- Outages, Troubleshooting, Workstations

Management



- Stand ups, Contract review, Training, QA review
- Engagement Management, Staffing, Governance



Don't overcomplicate your "science". Use few factors and simple formulas over complex.

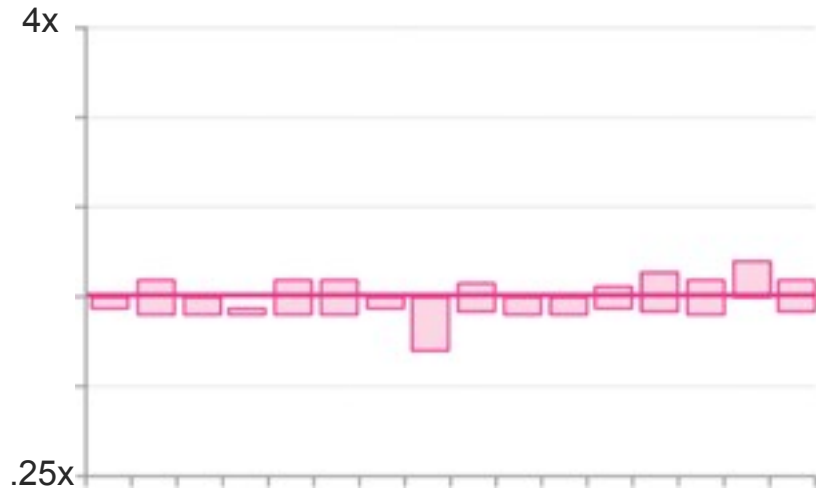


Overcomplicating Estimates

Many Factorial Adjustments



Few Factorial Adjustments



Estimation Techniques



Estimating “Things”



Lets assume that you are all expert estimators and that all the process and risk areas we have spoken today you know PERFECTLY how to avoid.

How many people are in this room right now?



APPROACHES TO ESTIMATING “THINGS”



The Hobbyist

*“I’m guessing
75 people”*

Judgement



The Factor Guy

*“15 rows 5
per row”*

Count



The Percentage Guy

*“150 person
capacity and its
half full”*

Compute



The Data Guy

*“The last 3
sessions have
had XX”*

Calibration



Count, Compute, Judge



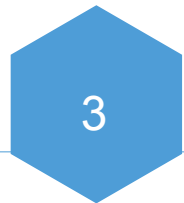
COUNT (& Calibrate)

Find something to count that has a direct correlation to what you are building



COMPUTE

Multiply by a statistically meaningful (proven) average



JUDGE

Only make assumptions as a final option (document)



Count if at all possible. *Compute* when you can't count. Use *Judgement* alone only as a last resort.



Common Items in these areas

1 Count/Calibrate

Site Specs

Content types

Taxonomies

Menus

Views

Wireframes

Designs

Migration

“features”

2 Compute

Non-build items

QA

Deployment

Project Management

Training

Governance

Engagement Management

3 Judge

Adjustment Factors

Risk Multipliers

Contingency

Unknown



Count & Calibrate

1) Determine an item that's highly correlated with the size of the software

2) COUNT how many you have

3) CALIBRATE your count with data from

- Industry Data
- Historical Data
- Project Data

How many hours for feature "X"?

Factor	Factor Count	Calibration	Gross Estimate
# Content Types	8	4	32



How many hours for Project Management?

Compute

- 1) Determine a defined value to base your computation
- 2) Determine a multiplier that is a statistically meaningful average
- 3) COMPUTE subtotal for your line item

Development Subtotal

1200 Hours

Subtotal	PM %	PM SubTotal
1200	20%	240



Judgement

- 1) JUDGE the rating of a specific multiplier (factor)
- 2) Apply multipliers based on rating of your estimate
- 3) Determine factors influence on the total project
- 4) COMPUTE

Factor	Ratings						Influence
	Very Low	Low	Nominal	High	Very High	Extra High	
Site Building Definition	2.0	1.25	1.0	.95	.85	.75	1.1

Site Build	PM %	Gross Estimate	Rating	Influence	Net Estimate
1200	240	1,440	Low = 1.25	1.1	1,980



Judgement

Cocomo II

- 17 effort multipliers
- 5 scaling factors

These are often applied to project subtotals

You can create your own multipliers based on your historical data

Factor	Ratings						Influence
	Very Low	Low	Nominal	High	Very High	Extra High	
Applications (Business Area) Experience	1.22	1.10	1.00	0.88	0.81		1.51
Database Size		0.90	1.00	1.14	1.28		1.42
Developed for Reuse		0.95	1.00	1.07	1.15	1.24	1.31
Extent of Documentation Required	0.81	0.91	1.00	1.11	1.23		1.52
Language and Tools Experience	1.20	1.09	1.00	0.91	0.84		1.43
Multisite Development	1.22	1.09	1.00	0.93	0.86	0.78	1.56
Personnel Continuity (turnover)	1.29	1.12	1.00	0.90	0.81		1.59
Platform Experience	1.19	1.09	1.00	0.91	0.85		1.40
Platform Volatility		0.87	1.00	1.15	1.30		1.49
Product Complexity	0.73	0.87	1.00	1.17	1.34	1.74	2.38

Site Building Definition 2.0 1.25 1.0 .95 .85 .75 1.1



Overcoming Judgement

Magnitude of Relative Error

$$\text{MRE} = (\text{Actual Result} - \text{Estimated}) / (\text{Actual})$$

Expected Case

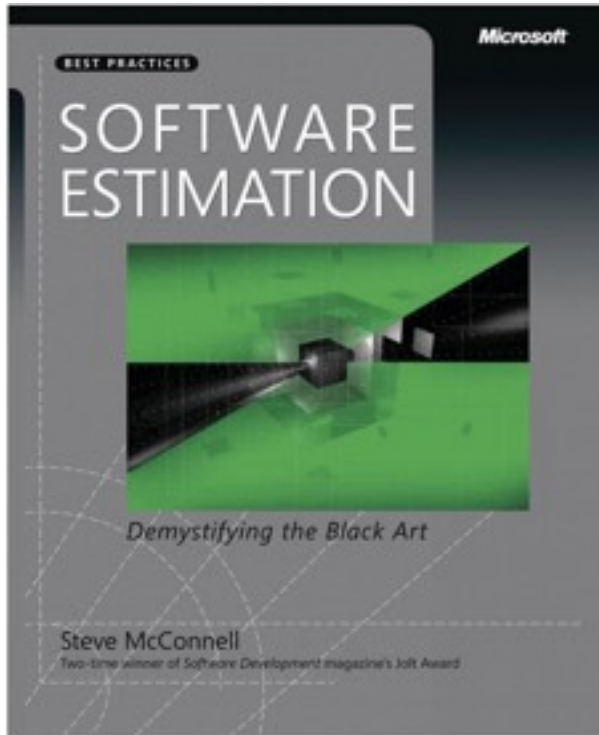
$$\text{EC} = (\text{Best Case} + (3 \times \text{Most Likely Case}) + (2 \times \text{Worst Case})) / 6$$



Judgement is the most hazardous kind of estimation due to *subjectivity* and unfounded *optimism*.



To Learn More...



Software Estimation: *Demystifying the Black Art*

Steve McConnell



Thank You

< CHICAGO, ILLINOIS />

1802 W. Berteau Ave. / Suite 209 / Chicago IL, 60613
773-525-8255

D.O / Twitter

Jsnoles49

Email:

John@PrometSource.com

John.Nollin@gmail.com

