Navigating the Minefield: Successful Software Estimation *before* Requirements are Complete

PRESENTED BY

CAROL A. DEKKERS

CAROLDEKKERS@GMAIL.COM

DANIEL B. FRENCH

DFRENCH@COBEC.COM

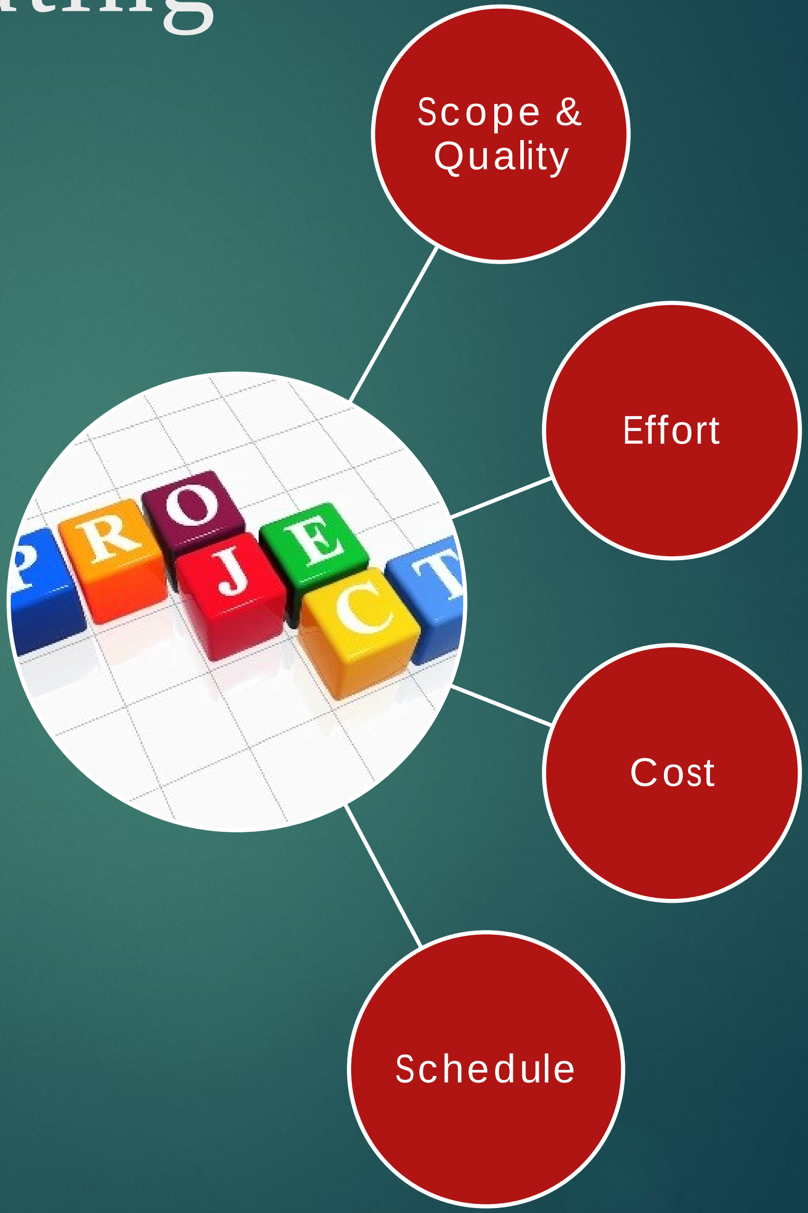
QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

Agenda

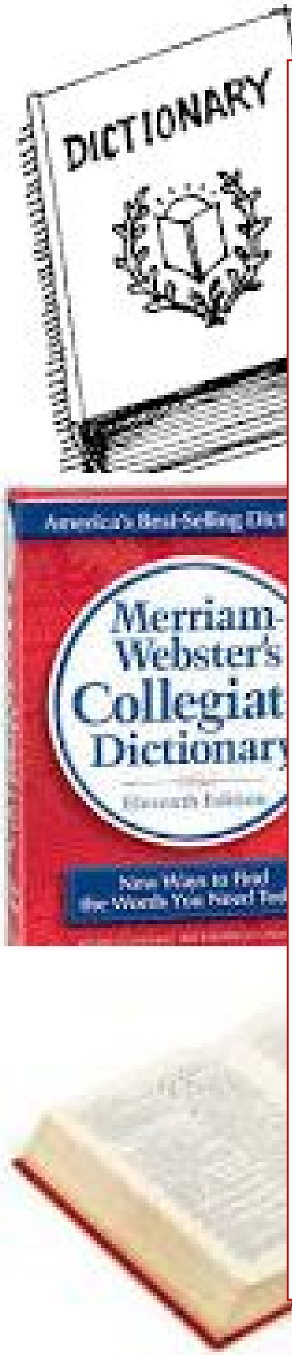
- Software Project Cost Estimating
- Project requirements - Demystified
- Estimates *before* Requirements
- DYB and gu(estimate)
- Q & A

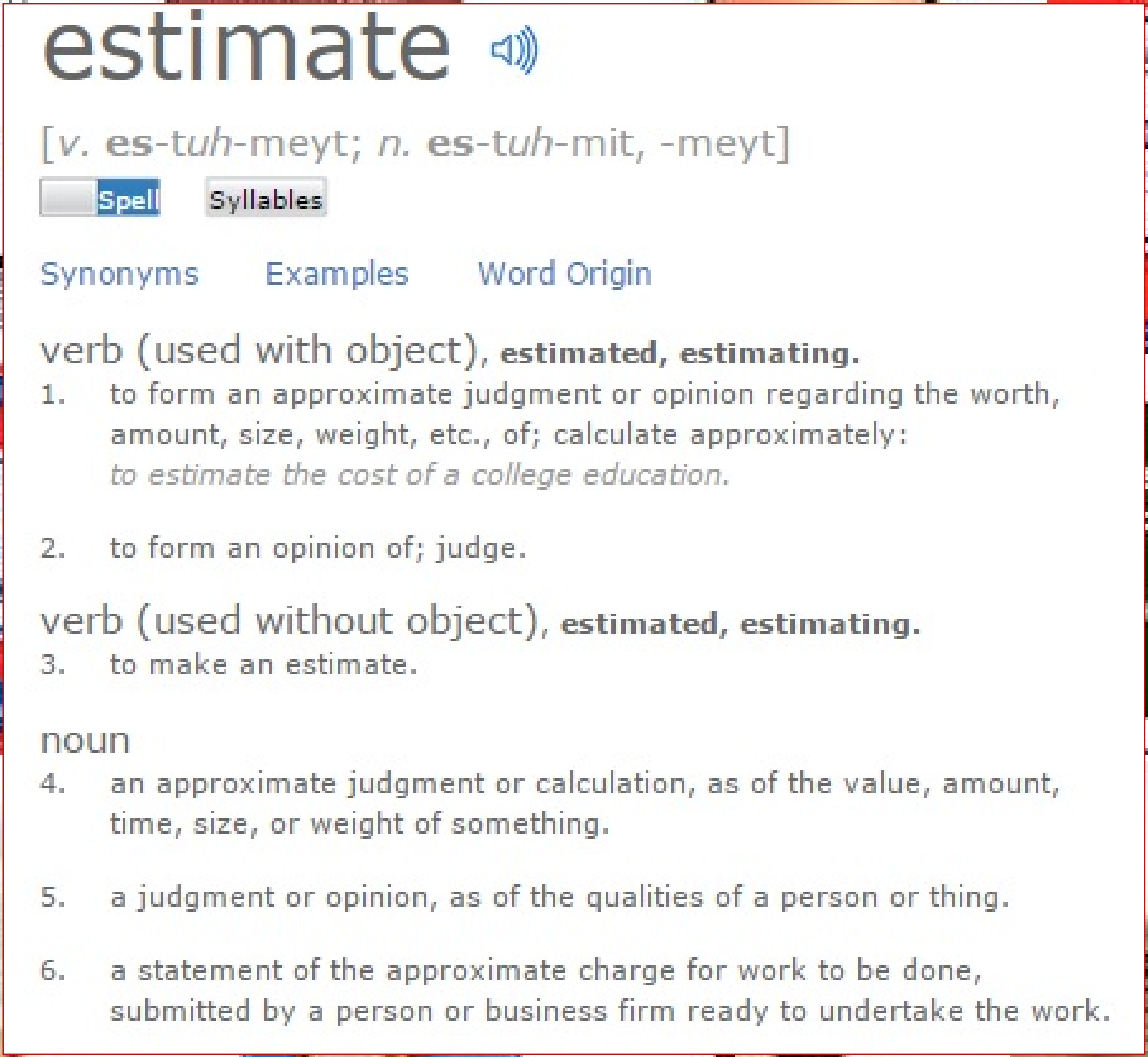
Project estimating



QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING


DICTIONARY
America's Best-Selling Dict
Merriam-Webster's Collegiat Dictionary
Eleventh Edition
New Ways to Find the Words You Need Fa



estimate 🔊

[*v.* **es**-t*uh*-meyt; *n.* **es**-t*uh*-mit, -meyt]

verb (used with object), **estimated, estimating.**

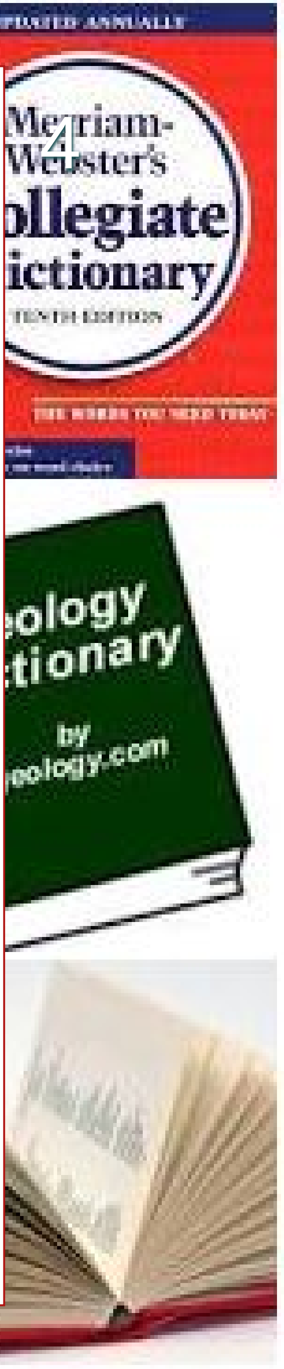
1. to form an approximate judgment or opinion regarding the worth, amount, size, weight, etc., of; calculate approximately:
   to estimate the cost of a college education.
2. to form an opinion of; judge.

verb (used without object), **estimated, estimating.**

3. to make an estimate.

noun

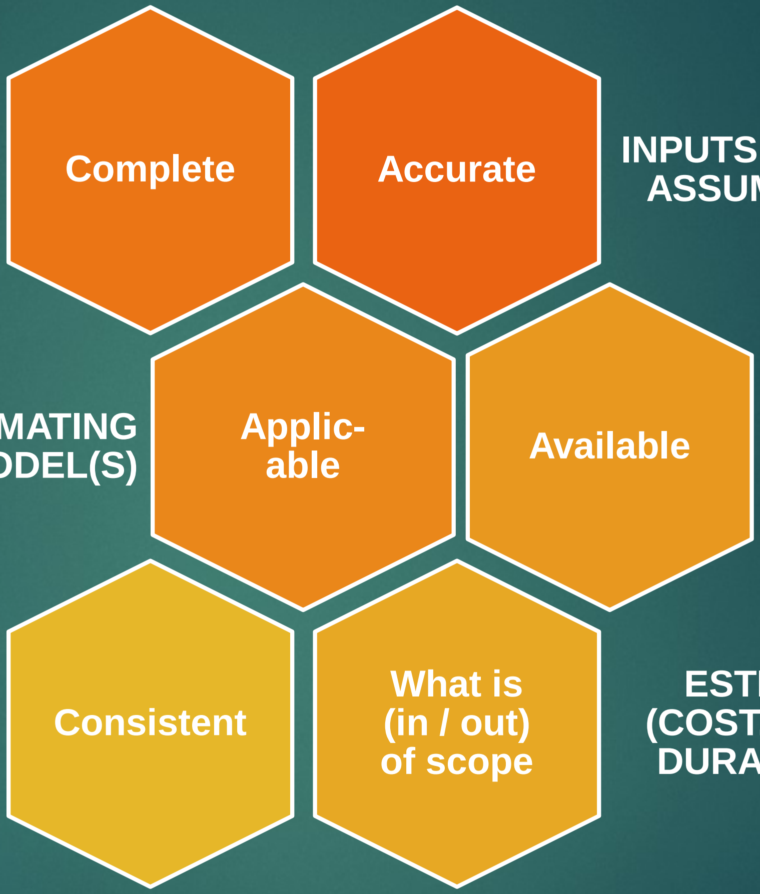
4. an approximate judgment or calculation, as of the value, amount, time, size, or weight of something.
5. a judgment or opinion, as of the qualities of a person or thing.
6. a statement of the approximate charge for work to be done, submitted by a person or business firm ready to undertake the work.



Project goals → Faster + Better + Cheaper…

delivered on-time & on-budget !

Typical estimating challenges



Complete

Accurate

INPUTS (KNOWN + ASSUMPTIONS)

ESTIMATING MODEL(S)

Applic-able

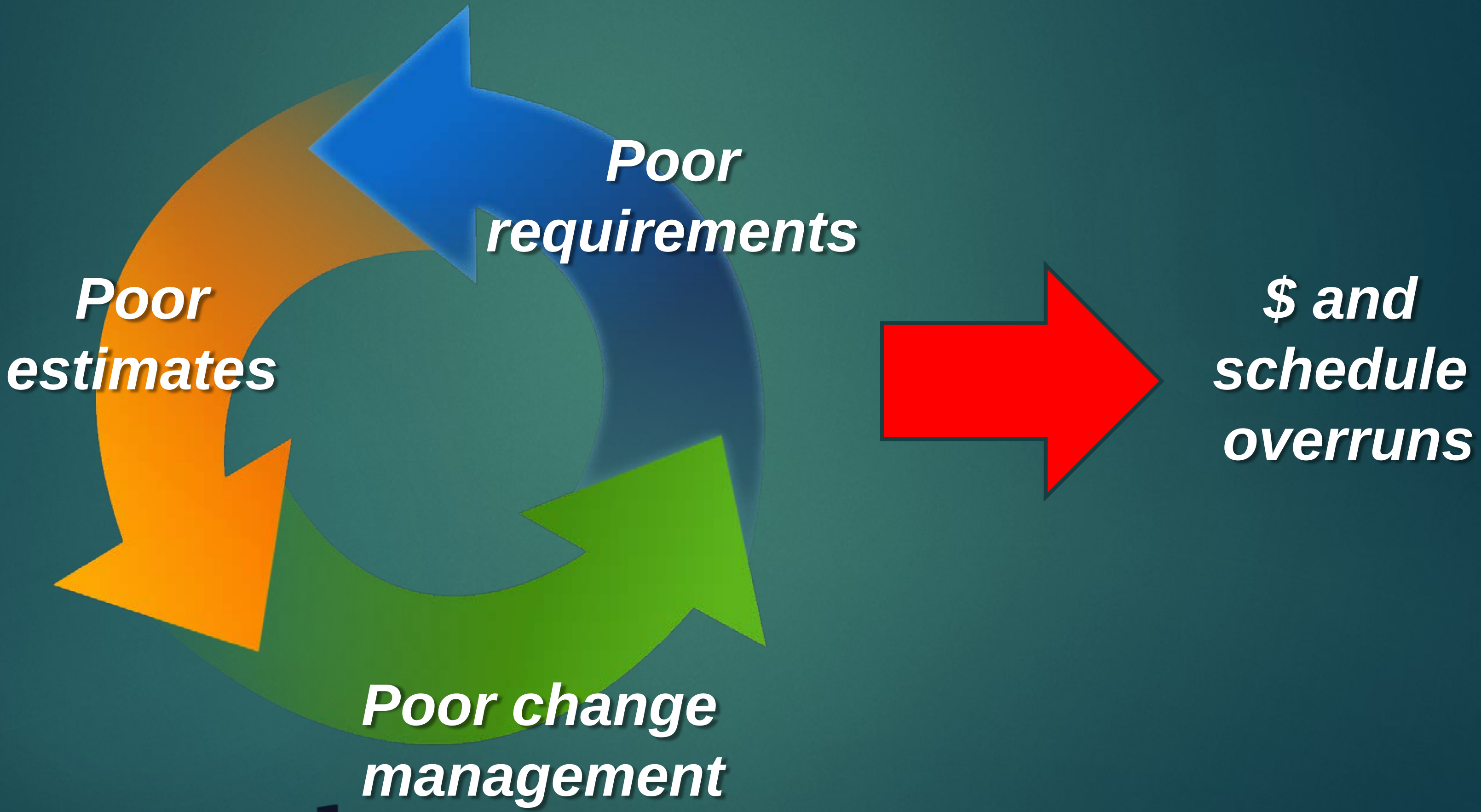
Available

Consistent

What is (in / out) of scope

ESTIMATES (COST, EFFORT, DURATION, …)

“Dog chasing its tail” Cycle



Poor requirements

Poor estimates

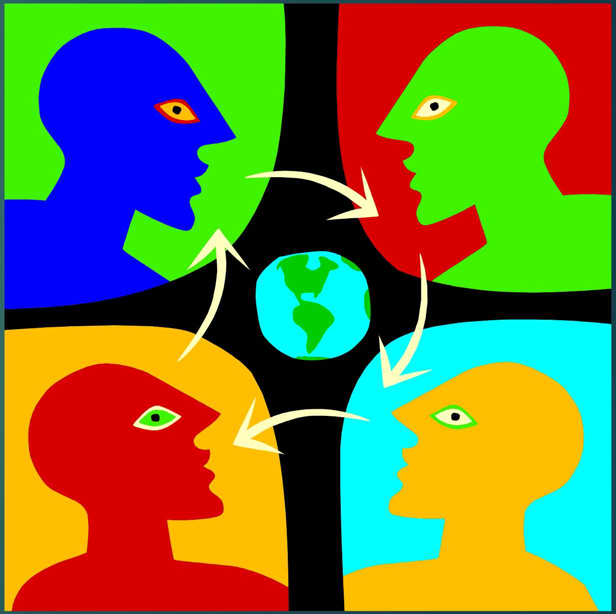
Poor change management

$ and schedule overruns

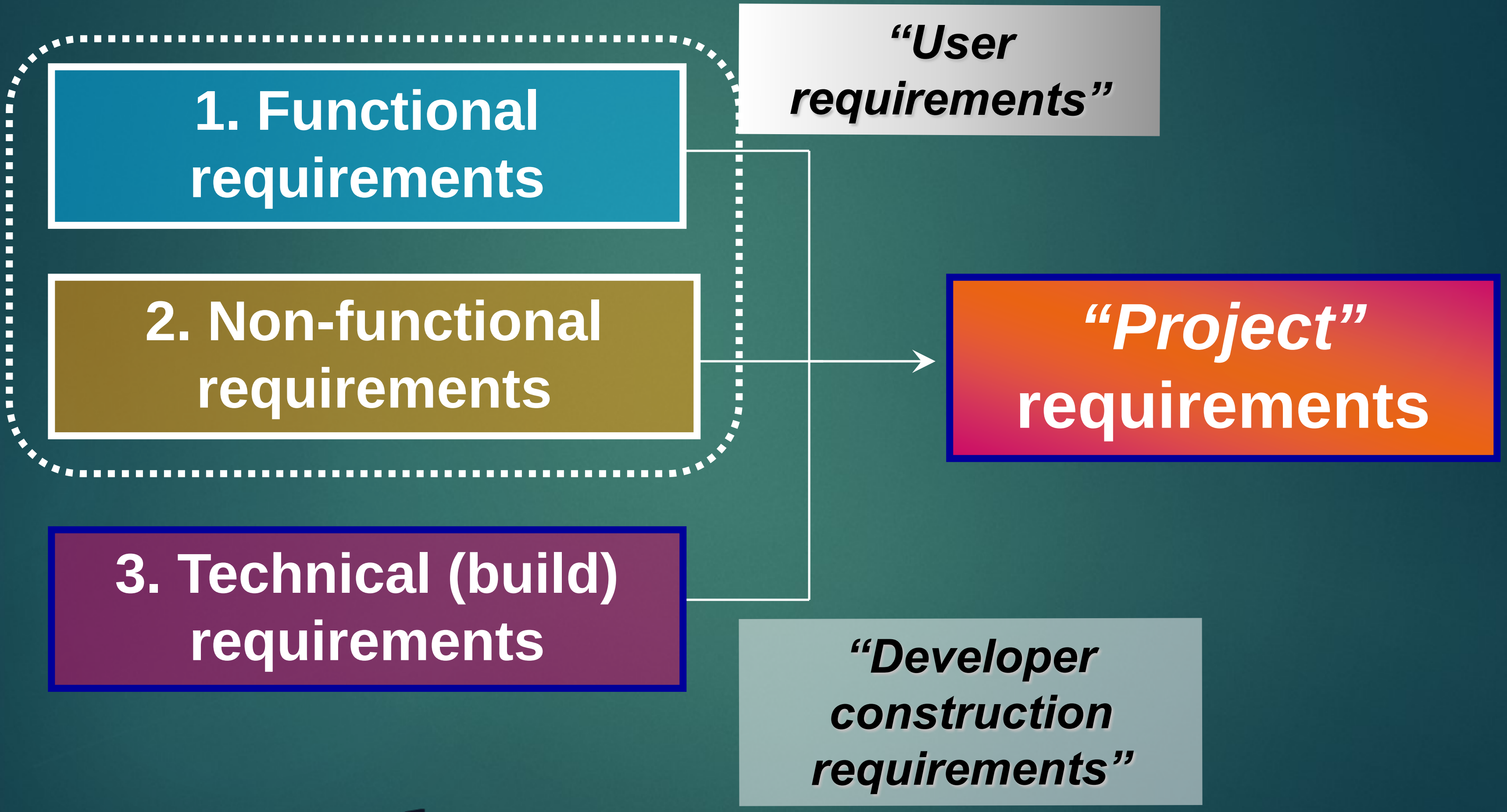
QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

Software requirements



Three distinct types of software requirements



1. Functional requirements

2. Non-functional requirements

“User requirements”

3. Technical (build) requirements

“Developer construction requirements”

“Project” requirements

How customers view requirements – non-transparent

Functional

Non-functional

Technical

Project requirements

How developers views requirements – blended

Functional

Non-functional

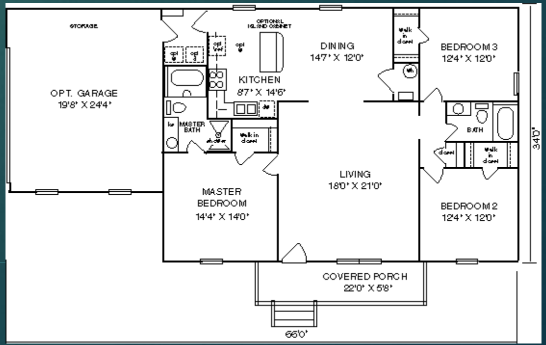
Technical

Project requirements

1. Functional requirements

WHAT software shall do
→ Documented with use cases or user stories

- Responsibility of customers/users to define
- Objective, documented
- Can be quantified (and estimated)
  → Functional size of software (function points: FP)
  → Key input to estimating
- “Floor plan for software”



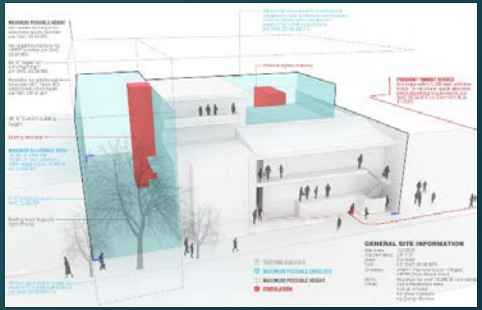
2. Non-functional requirements

<u>HOW</u> software shall perform: *Suitability, Accuracy, Interoperability, Compliance, Security, Reliability, Efficiency, Maintainability, Portability, Quality and "ilities"*

- Responsibility of customers/users to define
- Documented with supplementary specs or user stories
- Can be quantified:

  → SNAP points, VAF, COCOMO II effort multipliers
  → Can DOUBLE an estimate

- "International code for construction"



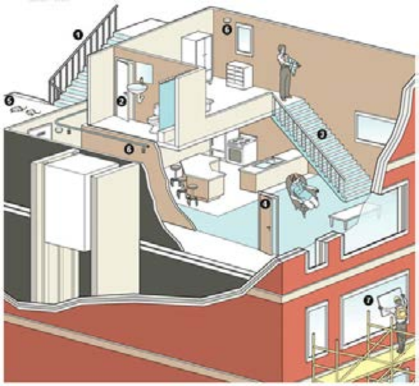
QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

3. Technical requirements

How team will develop or build the software

- May be influenced by Functional and Non-functional requirements plus environment
- Tools, methods, skills, programming language, etc.
- IT responsibility
- “Blueprints for software”
- NOT part of Use Cases, not part of FP, but….directly a part of estimate



QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

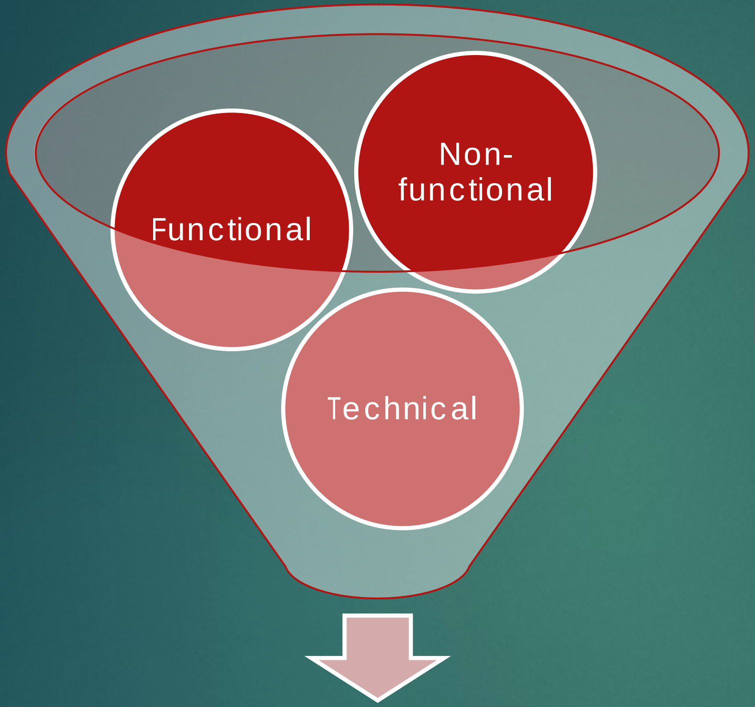
Segue: Agile User Stories

- Collection of high level software requirements:
  - Functional user stories (functions);
  - Non-functional (how, quality, usability, performance, etc.);
  - May include technical user stories;
  - May include other requirements (such as research spikes or bugs)
- May be inter-related and overlapping
- Details emerge during sprints
- Agile is part of technical requirement (How to build software)

QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

Estimating techniques

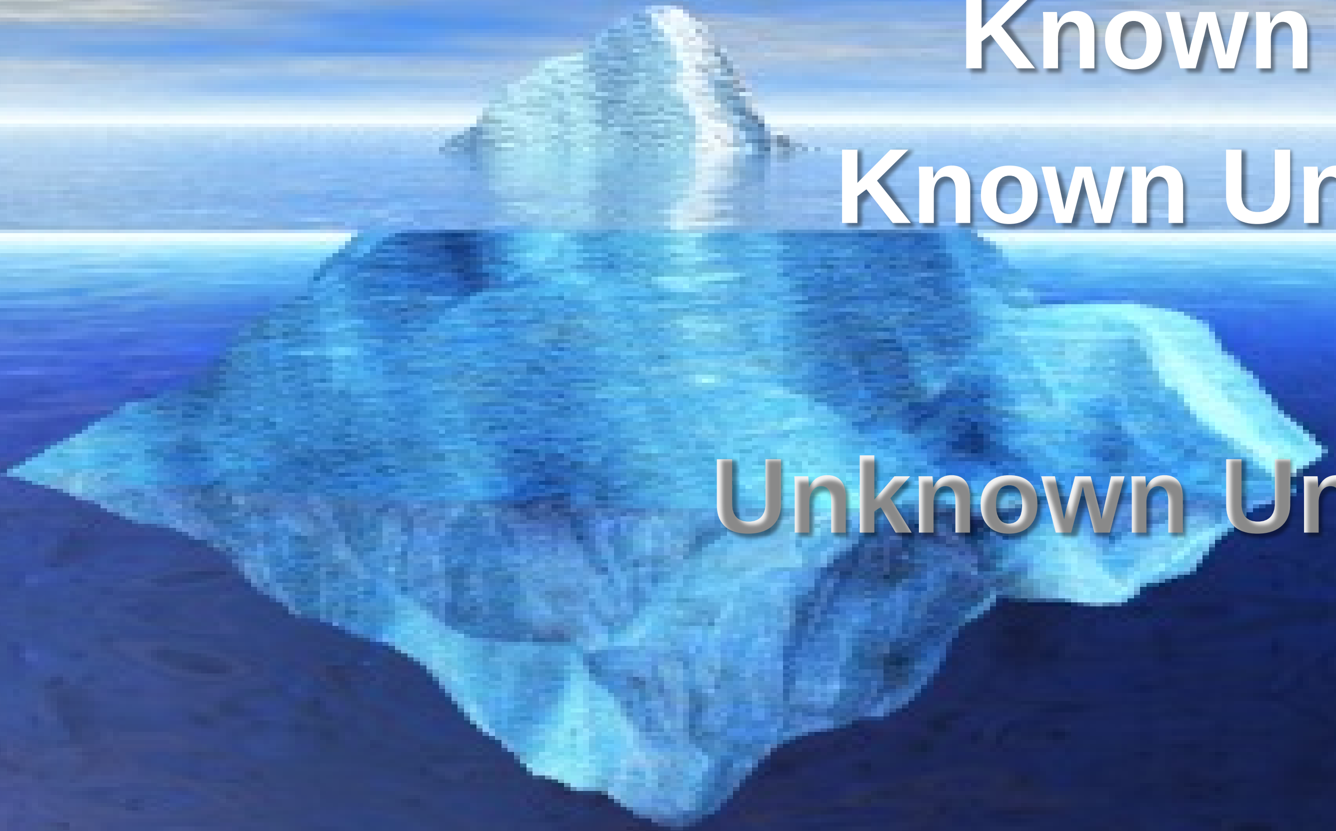


Estimates (cost, effort, schedule)

- Analogy
- Parametric
- Expert opinion
- Extrapolation from actuals
- Engineering build-up

- Non-trivial
- History is repeatable

Project estimating



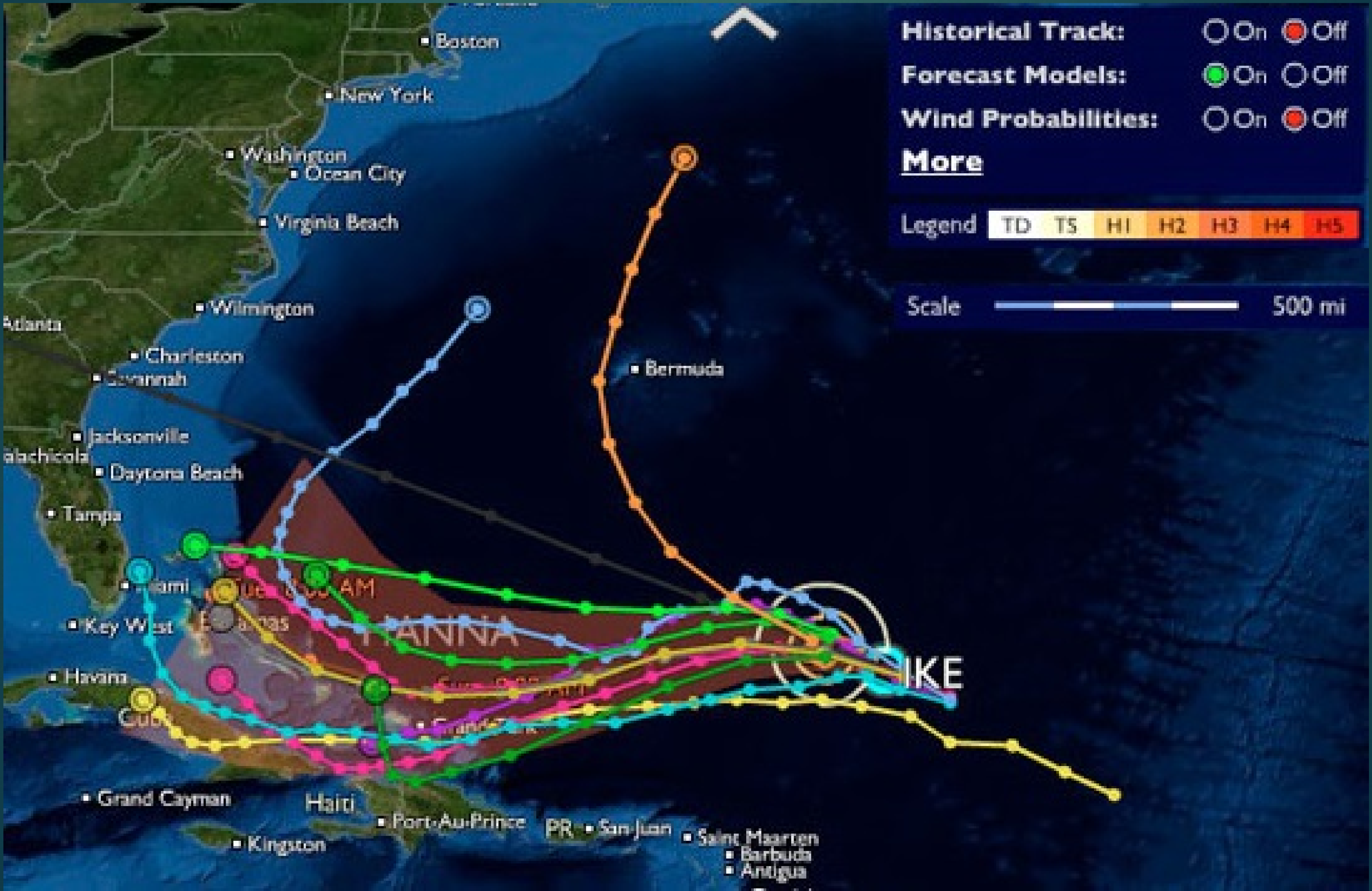
Known Knowns

Known Unknowns

Unknown Unknowns

Good estimates build on knowns

Predictive modeling



Historical Track: ○ On ● Off

Forecast Models: ● On ○ Off

Wind Probabilities: ○ On ● Off

More

Examples of estimating formulas

Metric	Units	Formula
Project Cost Ratio (completed projects)	$ / FP (or SLOC)	(Total Hours * Hourly Cost) + Other Costs Project Functional Size
Support Cost Ratio	$ / 1000 FP (or FTE / app)	(Support Hours * Hourly Cost) + Other Costs Application Functional Size
Repair Cost Ratio	$ / FP (or per fix)	(Repair hours * Hourly Cost) Functional Size of Repair

Today, estimating with good requirements and a solid model is similar to having a mapped minefield…

Estimating *before* requirements

Can you estimate an idea?

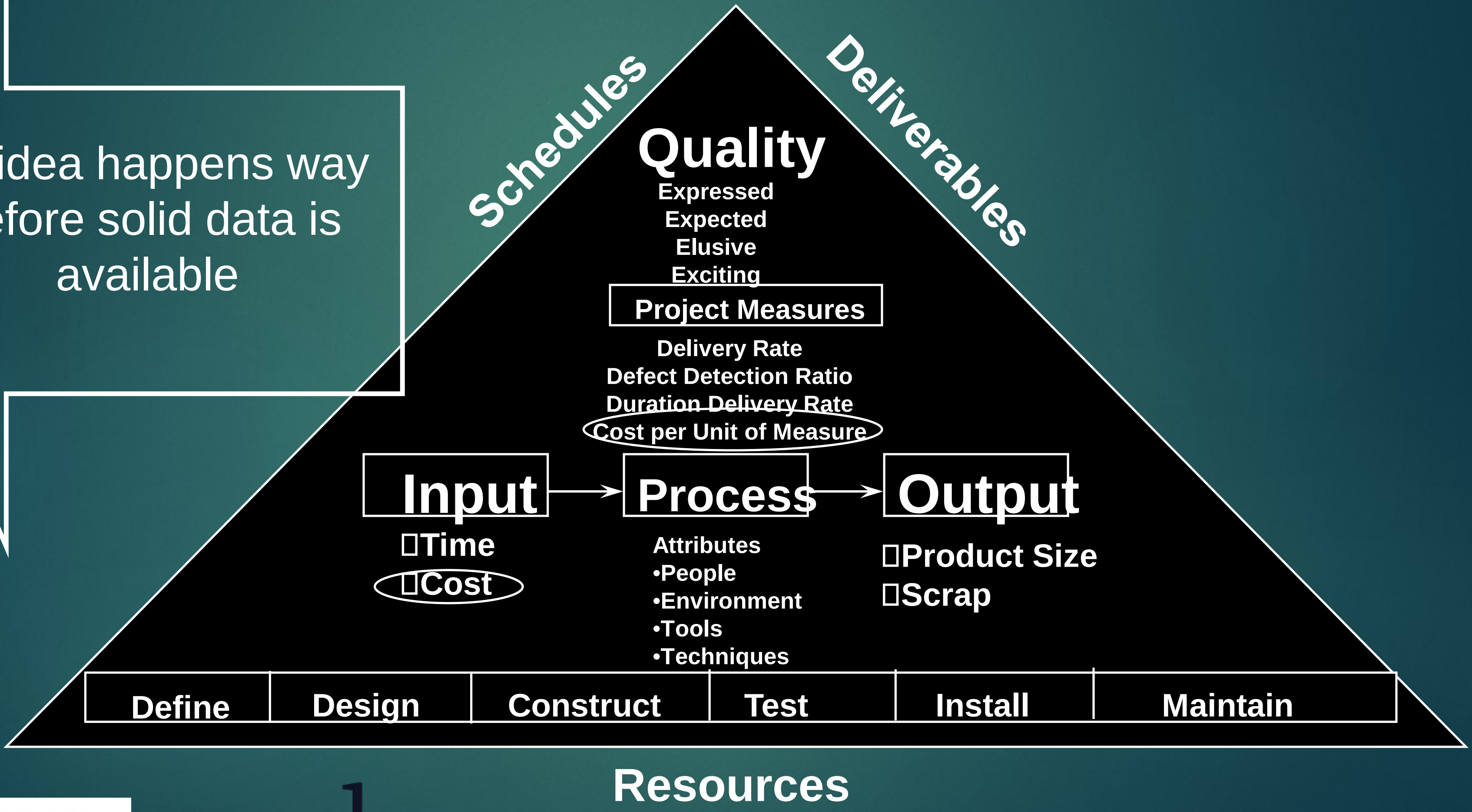
I have an idea… can you give me a quick estimate?

I won't hold you to it…
"ballpark" cost…
Kind-of-like xxx…
I need some #'s…
today

“Right now it’s just an idea…”

The Project Process Model

An idea happens way before solid data is available



Schedules

Deliverables

Quality

Expressed
Expected
Elusive
Exciting

Project Measures

Delivery Rate
Defect Detection Ratio
Duration Delivery Rate
Cost per Unit of Measure

Input → Process → Output

□Time
□Cost

Attributes
•People
•Environment
•Tools
•Techniques

□Product Size
□Scrap

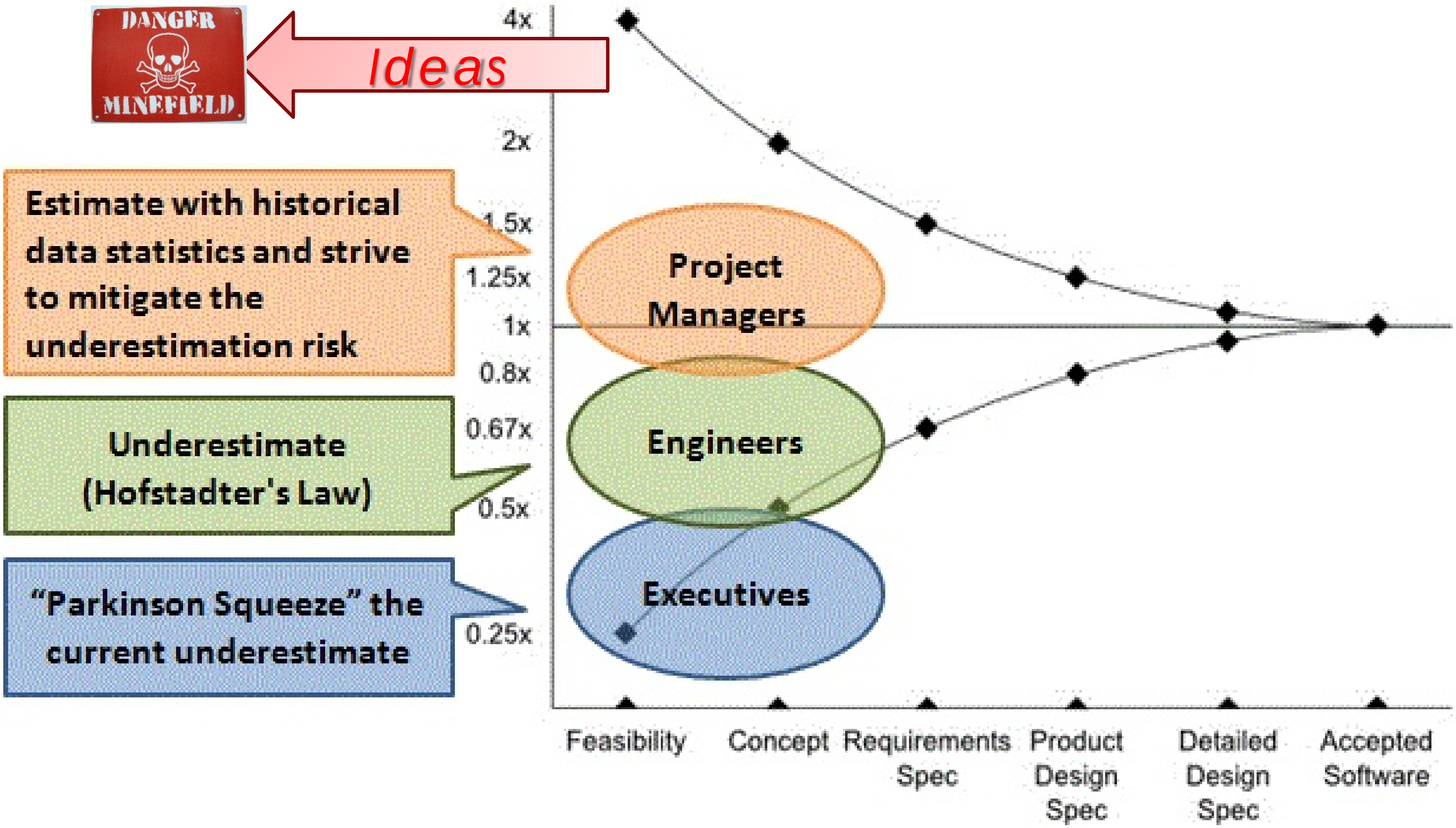
Define | Design | Construct | Test | Install | Maintain

Resources

QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

The Cone Of Uncertainty



Estimate with historical data statistics and strive to mitigate the underestimation risk

Underestimate (Hofstadter's Law)

“Parkinson Squeeze” the current underestimate

@ http://blog.karmona.com

When faced with a napkin, what can you do?

- Refuse to play
- Memory game (“Kinda-sorta”)
- Mutter (“professional ethics”) and leave
- SWAG ($1 million & 6 months)
- Promise (and disappear)

Or…

DYB (do your best) and (gu)estimate

DYB and (gu)estimate

1. Document base assumptions:
   - Functional size
   - Non-functional
   - Technical
   - Historical growth %

2. Document assumptions & do at least two estimates:
   - ROM based on analogy, or expert opinion

3. Save the napkin

1. Document base assumptions

Functional size: (example shortcuts)

- One File Model (# entities * 31 FP each)
- Simple Function Points
- T-shirt sizing (XS - 3X)
- ISBSG (International Software Benchmarking Standards Group) function point profiles
- Patterns (sizing by analogy)

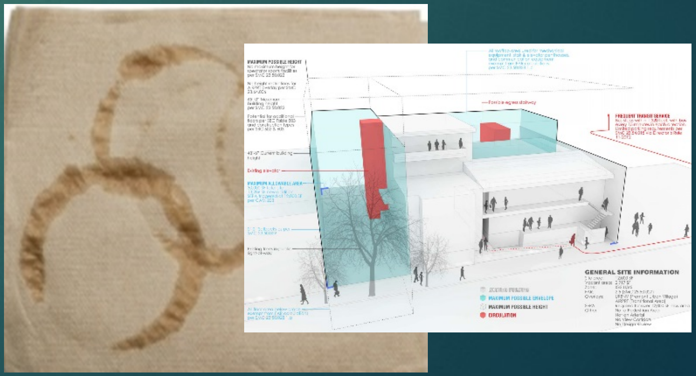
QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

1. Document base assumptions

Non-functional size: ("ilities")

- Document assumptions: standard constraints (e.g., performance, security, special needs)
- Consider adjustment factor or software non-functional assessment (SNAP)
- Overestimate the complexity (seldom less complex) & compare history



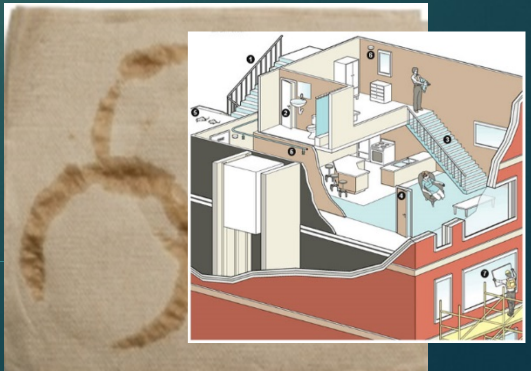
QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

1. Document base assumptions

Technical requirements:

- standard development tool suite for subject area
- tools used on similar systems
- consider "uniqueness" potential (architecture, method, novel subject matter, special skills sets, etc)



1. Document base assumptions

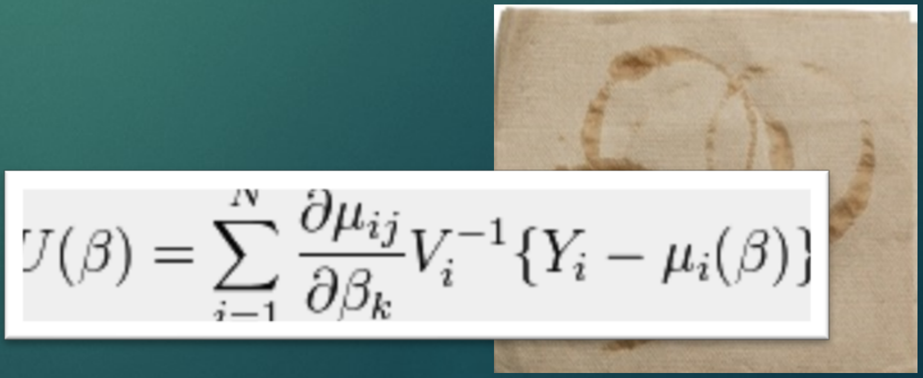
Adjust for uncertainty based on historical %:

- Size growth average 1.5- 2%* per month (18-month project → +30% FP)
- Rework 40% → can double expert opinion
- Risk management → document areas of greatest uncertainty

* Capers Jones, ***Estimating Software Costs: Bringing realism to estimating*** 2nd edition

2. Rough order of magnitude (ROM) equations

- Range of “(gu)estimates” → NO decimals!
- Rules of thumb, high level size
- Supplement with expert opinion (adjusted for uncertainty)
- Label results as “preliminary”
- Save copies of modeling exercises



$$U(\beta) = \sum_{i=1}^{N} \frac{\partial \mu_{ij}}{\partial \beta_k} V_i^{-1}\{Y_i - \mu_i(\beta)\}$$

QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING

3. Save the napkin

- Despite best intent → estimators cannot predict future based on ideas
- (Gu)estimates based on <u>documented assumptions</u> are a step forward
- If (gu)estimate becomes project budget /schedule → scan the napkin for Kick-off meeting

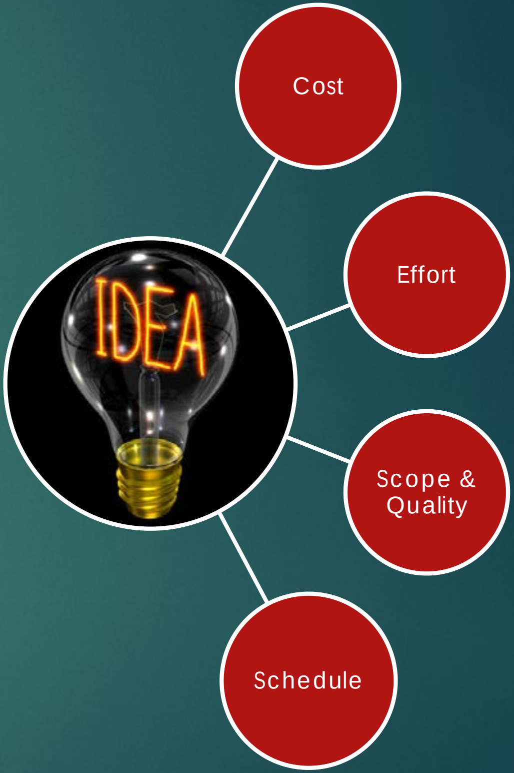
Project ABC
Budget $500 K
Preliminary requirements

QUALITY PLUS TECHNOLOGIES, INC.

CONSULTING

Garbage In,
Garbage Out
YOUR ANALYSIS IS ONLY
AS GOOD AS YOUR DATA

Now you can carefully… navigate the minefield



Cost

Effort

Scope & Quality

Schedule

QUESTIONS?

… THANK YOU

Navigating the Minefield: Successful Software Estimation *before* Requirements are Complete

PRESENTED BY

CAROL A. DEKKERS

EMAIL: CAROLDEKKERS@GMAIL.COM

DANIEL B. FRENCH

EMAIL: DFRENCH@COBEC.COM

QUALITY PLUS TECHNOLOGIES, INC.

cobec CONSULTING