Our Evolution around Estimation

Lenny Fenster
Microsoft Services Apps CTO
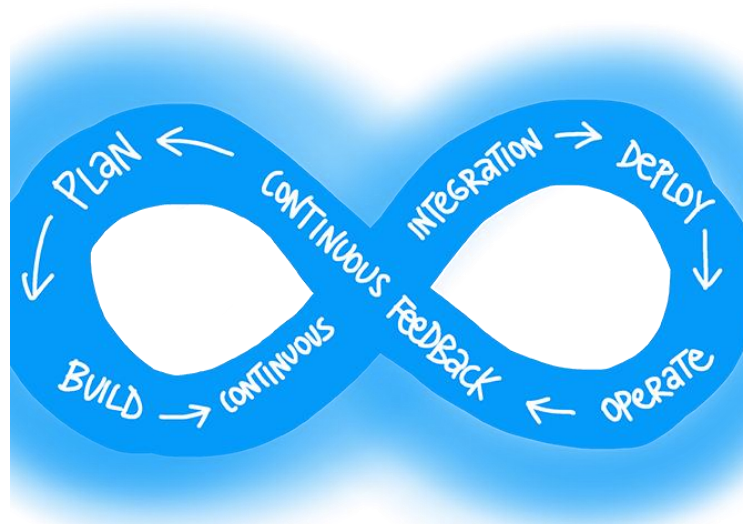
Microsoft

Agenda

- ▶ Background

- ▶ What pain points did we experience?

- ▶ What solutions have you developed to address pain points?

- ▶ What impact does this have on your requirements gathering efficiencies?

What do we do?

Who are we?

Over 1500 people serving as deep technical advisors for some of the largest software development projects around the world

Architects
- Define Solution design and estimate
- Create BE document and the SoW
- Lead for large and complex software development project
- Technical Quality Reviews

Premier Field Engineers
- Isolate and correct problems
- Debug, performance tune, troubleshoot
- Conduct design and supportability reviews
- Migrate, configure and test solutions

Consultants
- Ramp up quickly on customer applications, and operations
- Develop deep understanding of implications of the developed solution
- Deliver high quality engagements

Who are we?

Over 1500 people serving as deep technical advisors for some of the largest software development projects around the world

Architects

- **Define Solution design and estimate**
- Create BE document and the SoW
- Lead for large and complex software development project
- Technical Quality Reviews

Where were we?

Architect estimation problems:

- Under pressure to deliver estimates on time and within budget

- Not experts at using requirements to build user stories and properly size

- Not experts at using estimation tools

- Stability and complexity of existing estimation tools greatly affects consistent usage and results

- Using alternative tools such as MS Excel or MS Project to develop estimates

- Lack of the right customer information often results in inconsistent requirements.

- An overall lack of estimation guidance and a slow turnaround time perpetuates the current situation.

Apps Architects
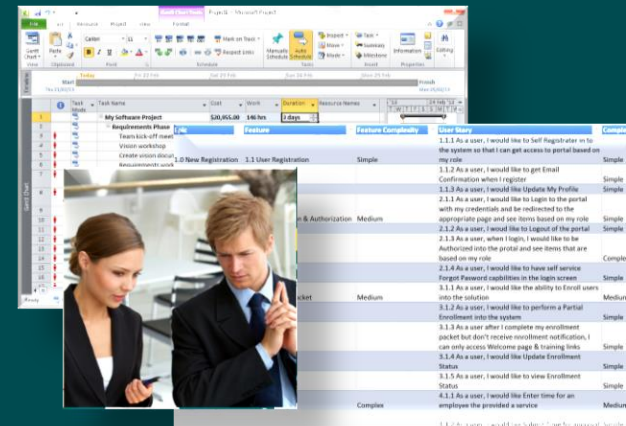
Defensible estimates

Estimation tools and methods

Requirements

Pain point: Non-defensible results

- ▶ Often show large variations across estimates and actuals
- ▶ Often not defensible or speaking to best practices
- ▶ Estimates are often not based on history
- ▶ Are not based on industry trends

Defensible results:

- ▶ Are based on a scientific approach that results in time and effort
- ▶ Come from requirements
- ▶ Are sized based on the appropriate estimation technique
- ▶ Tie back to the solution, customer, team, and technology constraints
- ▶ Explore a variety of potential outcomes and readily answer "what ifs"
- ▶ Are no longer dependent on architect's estimate "gut" feeling
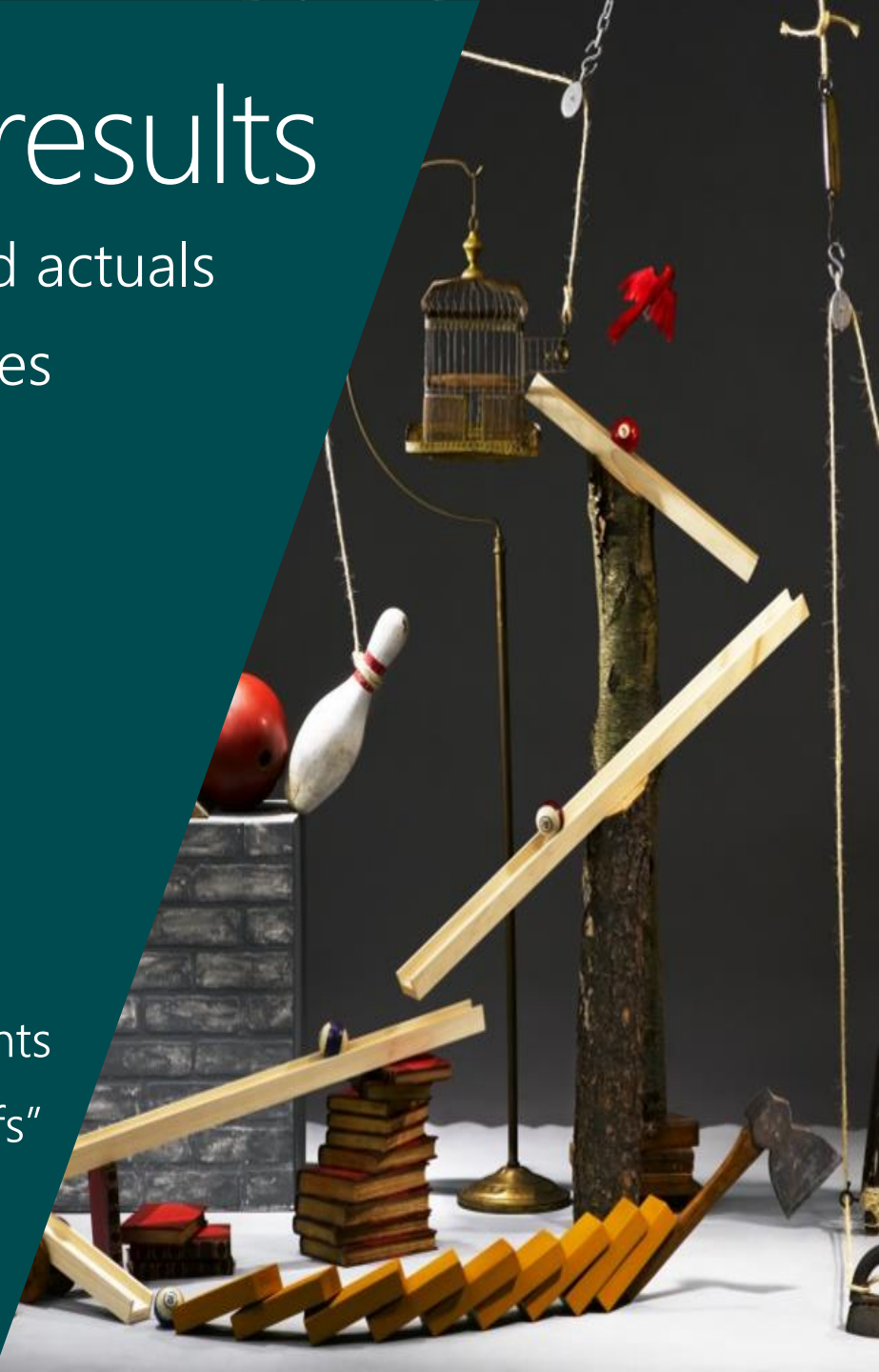
Our opportunity

Separate Estimation from Scoping

▶ Architects focus architecture ☺

    ▶ Requirements analysis

    ▶ Solution approach

    ▶ Determination of complexity and size

▶ A working environment that:

    ▶ Is not predicated on architects being estimation tool experts

    ▶ Reduces dependency on estimation tooling

    ▶ Establishes a dedicated a team of estimation tool, OSE and Software Lifecycle Management (SLIM) experts

    ▶ Removes OSE certification requirement for architects

    ▶ Enables consistent and "defensible" estimates derived by a collaboration between the domain architect and the CET

Apps architects
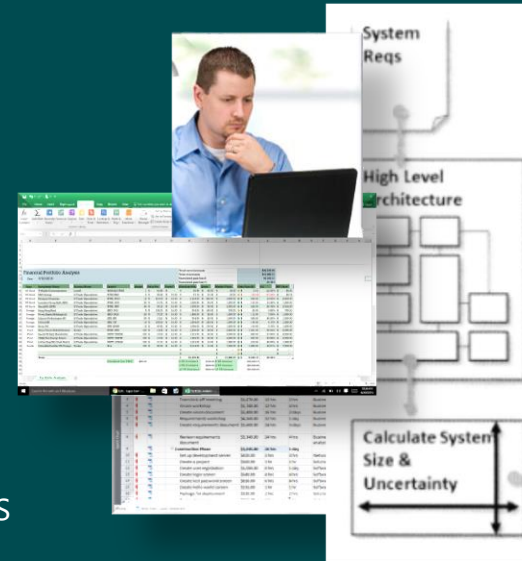
Central Estimation Team (CET)

Defensible estimates

Centralized estimation pursuit flow

Pursuit architect assigned
Architect is informed that they are assigned to support a pursuit

Requirements
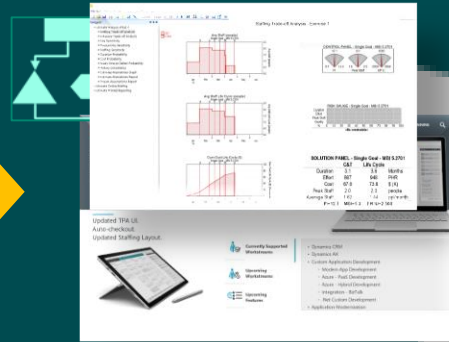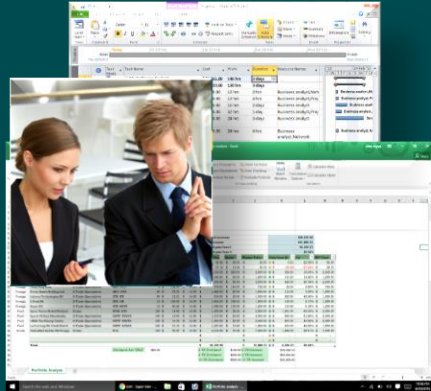Architect works with customer to collect and analyze requirements, determine high-level solution approach, complexity, and size. Architect decomposes solution and creates input package (as EFUs) for CET. Estimation expert from CET assigned to work with pursuit architect

SME EFU review
SME reviews EFUs for consistency, alignment to patterns, and IP reuse
• SME performs EFUs review

Creation of pursuit estimate (CET)
Architect closely collaborates with the CET estimation expert. The CET estimation expert uses the estimation tools based on the input requirements provided by the pursuit architect
• CET takes EFUs and models into SLIM

Review and acceptance of estimate
Architect reviews and accepts the estimate. If additional iterations are required, the architect works closely with the estimation tool expert on the revised estimate. The architect is accountable for, and ultimately takes ownership of, the estimate and is prepared to defend the estimate.
• CET takes SLIM output and builds resource plan

Apps Estimation
The Road to Recovery

PAST	TRANSITION	PRESENT/FUTURE

How we worked before…

▶ Architects estimated using various techniques – Project Work Breakdown Structure (WBS), custom spreadsheets

▶ Architects needed to know how to use the different estimation tools and techniques – One Services Estimator (Use Case Points [UCP,] WBS) in addition to the technology, requirements, and customer domain

▶ Effectiveness of estimates were not measured

How we work now/tomorrow…

▶ Take advantage of Centralized Estimation Team (CET)

▶ Use top-down estimation approach based on industry and past engagement experience

▶ Architects focus on the requirements, technical components – not the estimation tool

▶ Actuals are fed to the CET to improve estimation models

▶ Continue to use One Services Estimator (OSE) WBS as back up

Pain Point Examples

▶ Slow turn-around time

▶ Lack of consistency

▶ Estimates are not defensible

▶ Cannot compare against actuals

▶ Not enough time for requirements elicitation

▶ Estimation tool quality sub-optimal

Benefit Examples

▶ Faster turn-around time using CET

▶ Consistent and defensible estimates

▶ Actuals are measured against estimates

▶ Estimates are left to be calculated by the tool and more defensible

▶ Architect can focus on requirements, architecture, and solution

▶ Leverages SLIM for scientific and data-driven approach

Thank you