Towards True Engineering

Moving Beyond Craft to Professionalism

RBCS

TIME TESTED.
TESTING IMPROVED.

www.RBCS-US.com

Introduction

- The software profession is evolving rapidly, from its birth circa 1950
- Software testing is also evolving
- However, software and software testing are practiced more like crafts rather than real forms of engineering
- The history, progress, and attributes of other forms of engineering can illuminate the path forward
- Let's consider what a true software engineering (and software test engineering) profession will look like, based on other forms of engineering

What is Engineering?

- Engineering is the application of science, math, and technology in the construction of useful things
- Software certainly involves technology
- However, the science and math behind software remain fairly limited
- In the absence of these, we see software over-relying on skills and process, which are the hallmarks of a craft
- Crafts are perfectly good for building dressers and stained-glass windows, but would you want to drive on a bridge or fly on a plane built by craftsmen?

Guilds and Craftsmen

- In early forms of mechanical engineering, such as making arms, practitioners were craftsmen, often organized into guilds
- Crafts are somewhere between art and engineering, and rely on apprenticeship as a path to the status of master craftsman
- Guilds were formed by craftsmen to control entry into the craft
- Swordsmithing, an ancient craft still practiced today, now uses true engineering materials (e.g., 1086 steel) and has no guilds
- In the Middle Ages, swords used in crimes were destroyed because they were inhabited by evil spirits
- In current software engineering, professionals are blamed for process failures that result in bugs

Civil Engineering, Roman Style

- The Romans built amazing structures, some of which survive to this day
- Civil engineering, but with limited science and math
  - No Newtonian physics
  - No calculus
- So, the Romans had to practice a purely empirical form of engineering
- They knew from experience what structures worked for what purposes, but could not explain the math and science behind why they worked
- We're in the same predicament in software engineering
- We can build cool software, but we can't explain why it works—and why it sometimes doesn't

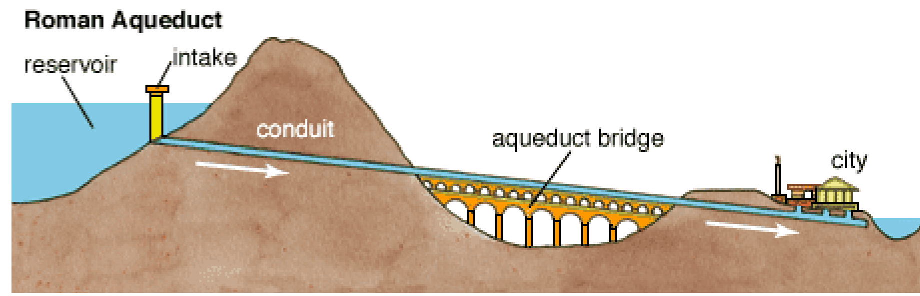
Civil Engineering, Gangnam Style

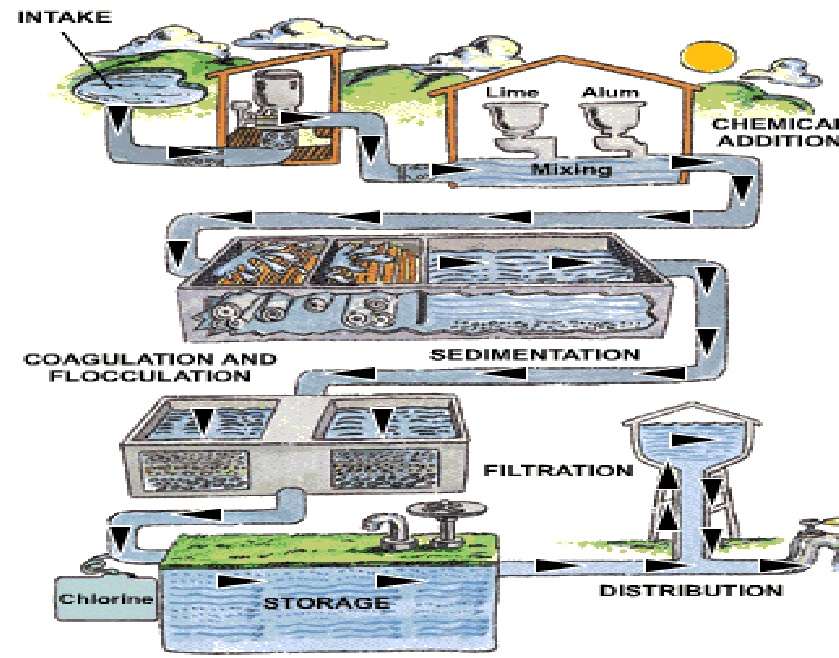
- You can build amazing aqueducts and bridges with empirical civil engineering
- However, you need true engineering to create really amazing things
- For example, consider the Seoul subway system (including now-infamous Gangnam station)

- Longest subway system in the world
- A marvel of technology and civil engineering
- This kind of achievement requires true engineering

Which Would You Rather Drink?

- Contrast unsanitary Roman system with a modern system
- Some elements are similar
- However, science and technology (esp. chemistry and medicine) make the key difference
- Of course, as the Old West saying goes, "Whiskey is for drinking; water is for fighting."





The Role of Modeling

- There's a big challenge for a big bridge in Hong Kong: Typhoons!
- How do you stress-test a bridge?!?
  - Engineers built scale models, including a full replica (1 to 400 scale) of the entire bridge
  - The models were put in a wind tunnel
  - Statistical method (Monte Carlo simulations) were used to simulate various storms and situations
- What made this possible?
  - Engineering materials
  - Mathematical modeling of wind, water, steel, and concrete behavior

- Ongoing comparisons between model predictions and actual response to storm stresses occur
- We're not there yet in software engineering, but we are progressing
- Performance modeling is in practical use today

Engineering Materials

- Software is built by combining words that get translated into instructions for CPUs
- What do other engineers build with?
  - Steel
  - Concrete
  - Glass
  - Plastics
- These standard engineering materials have well-known properties, such as strength, reactivity, melting point
- Their behaviors are also subject to well-understood mathematical formulas
- We don’t have real software engineering materials yet, but we’re moving towards that

Failure Drives Standardization

- Railway bridge collapses in the UK in the 1800s lead to standardization and rules about the materials, processes, etc.
- We have seen and continue to see that in software engineering
- CMMI and ISTQB are examples of that
- However, we continue to have significant disagreements and lack of standardization
- Some worry about premature standardization
- Others feel that any mistakes standardized now would eventually self-correct over time

Self-regulated or Government Regulated?

- In US, engineers are primarily self-regulated, like doctors and lawyers, with governments enforcing those regulations
- Other professions with less status, such as barbers and cosmetologists, are largely government regulated
- From Natl Society of Prof. Engrs. website:
  - "A century ago, anyone could work as an engineer without proof of competency. In order to protect the public health, safety, and welfare, the first engineering licensure law was enacted in 1907 in Wyoming."
  - "Just as the CPA defines the accountant, and a law license defines the lawyer, the PE license tells the public that an engineer has mastered the critical elements of the profession."
- What do we want?
- Certification programs are part of demonstrating an ability to self-regulate

Impossible to Get There in 150 Years?

- Some naysayers might pooh-pooh the idea of making software engineering a true form of engineering in 150 years
- Really? Consider…
  - December 17, 1903
  - April 12, 1961
  - March 2, 1969
  - April 12, 1981
- In less than 150 years, aeronautical engineering has progressed beyond what anyone would have believed 150 years ago




A Vision of Software Engineering in 2100

- Relegation of "crafts" approaches to software hobbyists and personal projects only
- Engineering materials in the form of standard, tested, reusable components with known properties
- Math to represent software behavior
- Modeling (which will be made more accurate from the math)
- Adoption of continuously refined, relevant standards
- Government recognition of reputable certifications in software development and testing

Conclusions

- We've come a long way in the last 60 years, but software engineering still has a long way to go
- The focus on process and skills can only take us so far
- We need the science, the math, and the engineering materials
- We also need the leadership of our profession to step up and lead the transformation into a true engineering discipline

To Contact RBCS

For almost twenty years, RBCS has delivered consulting, outsourcing and training services for software and hardware testing and quality. Employing the industry's most experienced and recognized consultants, RBCS conducts product testing, builds and improves testing groups and hires testing staff for hundreds of clients worldwide. Ranging from Fortune 20 companies to start-ups, RBCS clients save time and money through improved product development, decreased tech support calls, improved corporate reputation and more. To learn more about RBCS, visit www.rbcs-us.com.

Address:	RBCS, Inc. 31520 Beck Road Bulverde, TX 78163-3911 USA
Phone:	+1 (830) 438-4830
Fax:	+1 (830) 438-4831
E-mail:	info@rbcs-us.com
Web:	www.rbcs-us.com