SOFTWARE ENGINEERING

University of Waterloo

Software Engineering Profession and Discipline

Joanne Atlee
Director of Software Engineering
University of Waterloo

An extension and update of David Notkin's CRA Snowbird presentation on Software Engineering Licensing and Certification, from a Canadian perspective.

Approaches to Software Quality

- Focus on the product
- Focus on the process to build the product
- Focus on the people who build the product
  - Software engineering education
  - Licensing of software engineers

Topics

- Education
  - Efforts to define a body of knowledge
- Licensing of software engineers as P.Eng.s
  - Provide some basic facts
  - Present the varying opinions and the rationales behind them
- Discussion

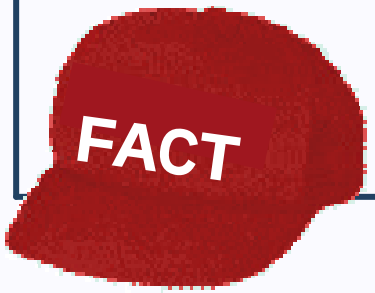
I will try to focus on each topic separately, but they are intertwined, so it will be hard.

Multiple Hats

Hat idea borrowed from David Notkin

SE Programs in Canada

SE Degrees Offered by CS	SE Degrees Offered by Engineering	CS/SE Options	CE/SE Options
Waterloo ↔ Joint	Waterloo	Waterloo ↔ Joint	Waterloo
New Brunswick ↔ Joint	New Brunswick	Victoria ↔ Joint	Victoria
McGill*	McGill*	UBC	UBC
Saskatchewan (CSAC)	Ottawa (CEAB)	Toronto	Toronto
	McMaster (CEAB)	Alberta (CSAC)	Alberta
	Western (CEAB)	Queen's	Queen's
	Calgary	Calgary	RMC
	Concordia	Simon Fraser	Regina
		Western	Ecole Poly
		Windsor	Sherbrooke
		Brock	ETS
		Memorial	
		Montreal	



***Subject to approve by Ministry of Education**

SE is a specialty within Engineering

SE is an engineering discipline that specializes in the design of software artifacts....

...and is distinct from Computer Science

- Requires an appreciation for other engineering disciplines
  - software might have to interact with other engineered components
- Requires an understanding of how the physical world behaves
  - software might have to react to or control the physical world
- Emphasizes teamwork

SE is a specialty within Engineering

Degree programs offered with engineering titles should be accredited as engineering programs

- non-accredited engineering programs weaken standards intended to protect the public.
- could be viewed as fraudulent by graduates who find that the degree they earned does not satsify the academic requirements for a P.Eng. licence.

SE Evolved from Computer Science

- Computer science has an engineering flavour, in that computer scientists design and build software artifacts.
- The discipline of software engineering was founded primarily by computer scientists
- Software engineering advances over the past 30+ years have been primarily from the results of computer scientist researchers.
- Computer scientists know more about building software than any other group.

SE is unlike other Engineering Disciplines

- Software behaviour can have many discontinuities
  - which makes it much more difficult to model and evaluate than a (non-discrete) physical component
- Traditional engineering programs instill no appreciation for the complexity of software problems.
  - software is viewed as being malleable and inexpensive to change
- Engineering curriculum cannot include serious computing science depth and still maintain a shared experience with other branches of engineering.

SE = CS + Eng

Computer Science

- data management
- data transformations
- design patterns
- algorithm paradigms
- programming languages
- human-computer interfaces

Engineering

- disciplined process
- large, integrated systems
- coordinated teams
- non-functional properties
  (e.g., performance, reliability, maintainability, ease of use)

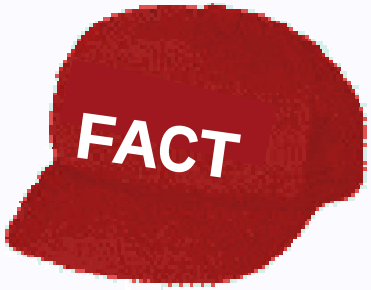
Waterloo's SE program is designed to be accredited by both Eng. and CS

CS & CE

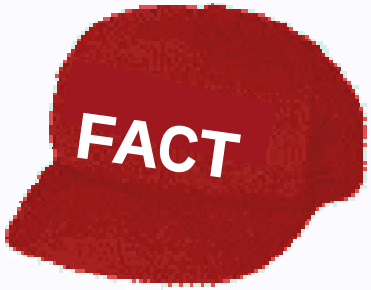
- programming principles
- data structures & alg.
- digital circuits
- digital computers
- programming languages
- dynamic systems
- databases
- concurrency
- operating systems
- networks and security
- performance evaluation

Software Engineering

- abstraction and specification
- project management
- requirements
- software architecture
- testing and quality assurance
- human-computer interfaces

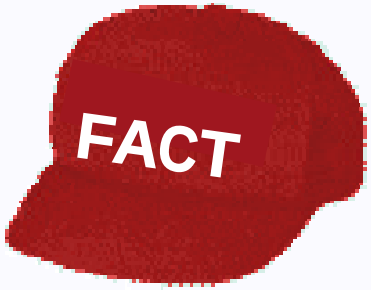


SE Body of Knowledge



- Accreditation and licensing require the definition of a core body of knowledge of principles, practices, technical knowledge (CS, Eng., and SE knowledge)
- This core body of knowledge should reflect good practices that are applicable to most SE products, are achievable, and provide some assurance of quality
- The practices need not be guaranteed to produce perfect software

Canadian SE Bodies of Knowledge

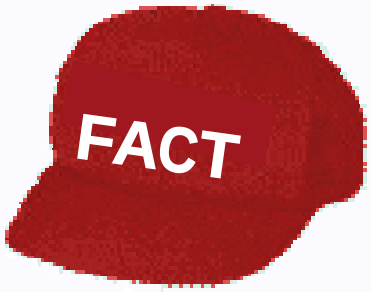


	CEQB*	PEO	BC	OIQ	MAC	UoO	UWO	UW	UNB
Discrete Math	-	✓	-	✓	1	1	1	2	2
Data Struct. & Alg.	✓	✓	✓	✓	1	2	3	2	2
Software Design	✓	✓	✓	✓	2	3	3	2	3
Testing, Reliability	-	✓	-	✓	1	1	1	1	1
Digital Systems	✓	✓	✓	✓	1	1	1	1	1
Computer Arch.	✓	✓	✓	✓	2	2	1	1	2
Operating Systems	✓	✓	✓	✓	1	1	2	2	1
File and Database	✓	-	✓	✓	1	1	1	1	1
Systems, Control	-	✓	-	-	2	-	-	1	1

* Proposal

Licensing Software Engineers

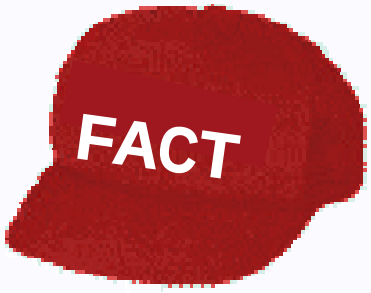
- What is it?
- Current status?
- To what degree should it become mandatory?



Licensing and Certification

- The general purpose of *certification* and *licensing* is to verify and provide assurances about the competency of those being certified/licensed.

- Provincial and state governments mandate the licensing of certain professionals who are legally required to practice at a level consistent with public safety (doctors, lawyers, professional engineers).

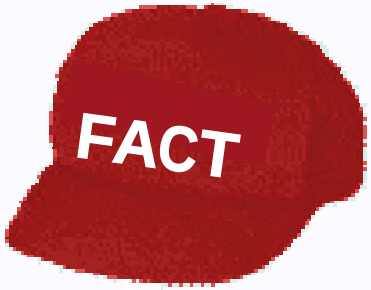


Licensing and Certification

- A professional society may voluntarily operate a certification program to authenticate practitioners' competency and to inspire public confidence in the profession.

  CIPS Information Systems Professional (I.S.P.), IEEE Software Engineering Certification

- Some software companies offer product- or vendor-specific certifications that assess a user's proficiency in using the company's products.



Professional Engineers (P.Eng.)

FACT

In Ontario, any practising engineer must be licensed as a P.Eng.

Anyone who practises engineering without a license or a temporary license is guilty of an offense and may be fined.

Most provinces define “engineering” by *practice* (what an engineer does) rather than by *title*

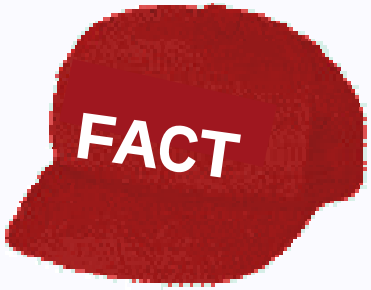
Professional Engineers Act

In Ontario, the practice of professional engineering is defined in the **Professional Engineers Act** and comprises three tests:

1. Any act of designing, composing, evaluating, advising, reporting, directing or supervising;

2. Wherein the safeguarding of life, health, property or the public welfare is concerned, and

3. Requires the application of engineering principles, but does not include practising as a natural scientist.



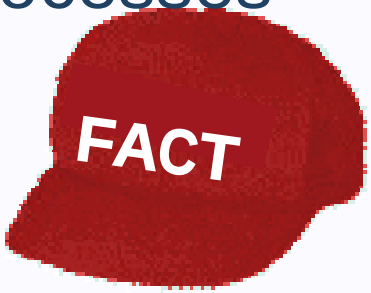
Enforcement varies

- The PEO does not have the resources to enforce the licensing of every practicing engineer

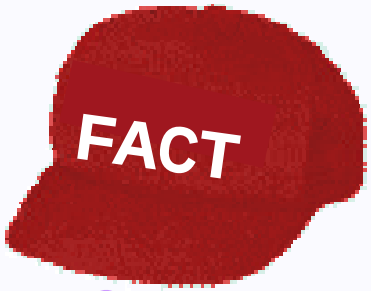
- Licensing is most prevalent among civil engineers -- for reasons that would affect software engineers

  - they deal with government employees
  - they sell their services directly to the public

  neither of whom can be expected to assess quality of the engineer's products or processes



Getting a P.Eng. Licence



CEAB degree

CEAB-equivalent degree

Technologist or equivalent degree

Confirmatory Exams

4 Basic Studies
8-12 Advanced Examinations

4 Years acceptable Work Experience

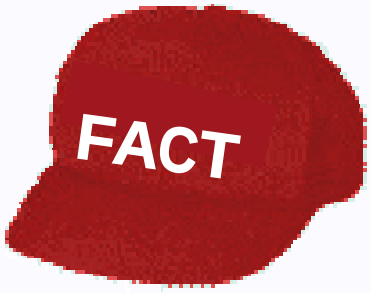
10 Years acceptable Work Experience

Professional Practice Examination

Professional Engineer (P.Eng.)

A P.Eng. is evaluated within a specific engineering discipline, but is licensed as a general “Engineer”.

A P.Eng. practising beyond his or her area of competence may be subject to disciplinary action by the provincial professional engineering society.



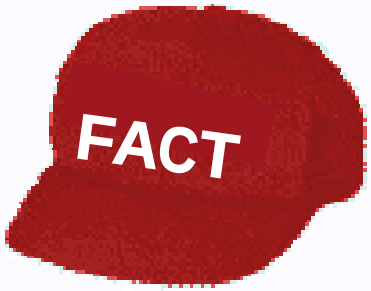
Current Status

Provinces that are licensing software engineers as P.Eng.s

- Ontario, British Columbia, Alberta

States that advertise they are licensing software engineers as P.E.s

- Texas



Arguments for Licensing Software Engineers as P.Eng.s

- The practice of software engineering falls under the Professional Engineers Act
- Licensing software engineers would improve software quality by raising the competence of practitioner
  - Would encourage education
  - Would encourage the application of best practices – even among the non-licensed
  - Would provide assurances to the public

Other arguments for Licensing Software Engineers

- Would act as a driver to improve our engineering of software and our education of software engineers
- Would advance the professionalization of software engineering

Arguments against Licensing Software Engineers as P.Eng.s

- Software engineering should be practised by whomever is best, and there is no evidence that engineers are the best
- Licensing software engineers would not improve software quality, and would provide false assurances to the public about software quality
- There is no widely accepted body of knowledge defining competency in software engineering
- Public safety can best be protected by certifying software products rather than software producers

Long-term Forecast

- I predict that within my professional lifetime, the development of some software systems will be restricted to P.Eng.s who are competent to practice software engineering.
  - safety-critical software
  - software components in engineered products
  - software that models or controls the physical world

- Licensing of software engineers will become serious only when the public demands it
  - possibly after the catestrophic failure of some software system

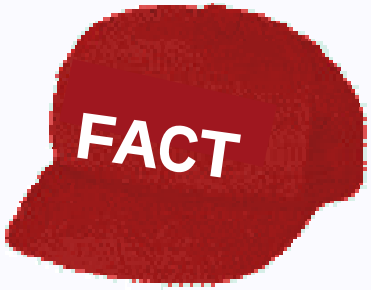
Role of Computer Scientists

- The demand for well-qualified software specialists far outstrips the supply of software engineering P.Eng.s, and will do so for the foreseeable future.

- One possibility would be to legislate the roles that computer scientists vs. software engineers play in developing critical software systems
  - a la the distinction between architects and structural engineers
  - exempt from the Professional Engineers Act those software systems that have little interaction with the physical world (e.g., financial systems, information systems, telecommunications systems,)

Temporary solution

- Until there is a critical mass of P.Eng.s trained as software engineers, computer scientists should be allowed to work on such critical systems.
- Liberalizing the rules for obtaining a limited licence would help.
- A Limited Licence is a licence issued to an individual who, as a result of ten or more years of specialized experience, has developed competence in a certain area of professional engineering.



Insert discussion here

References

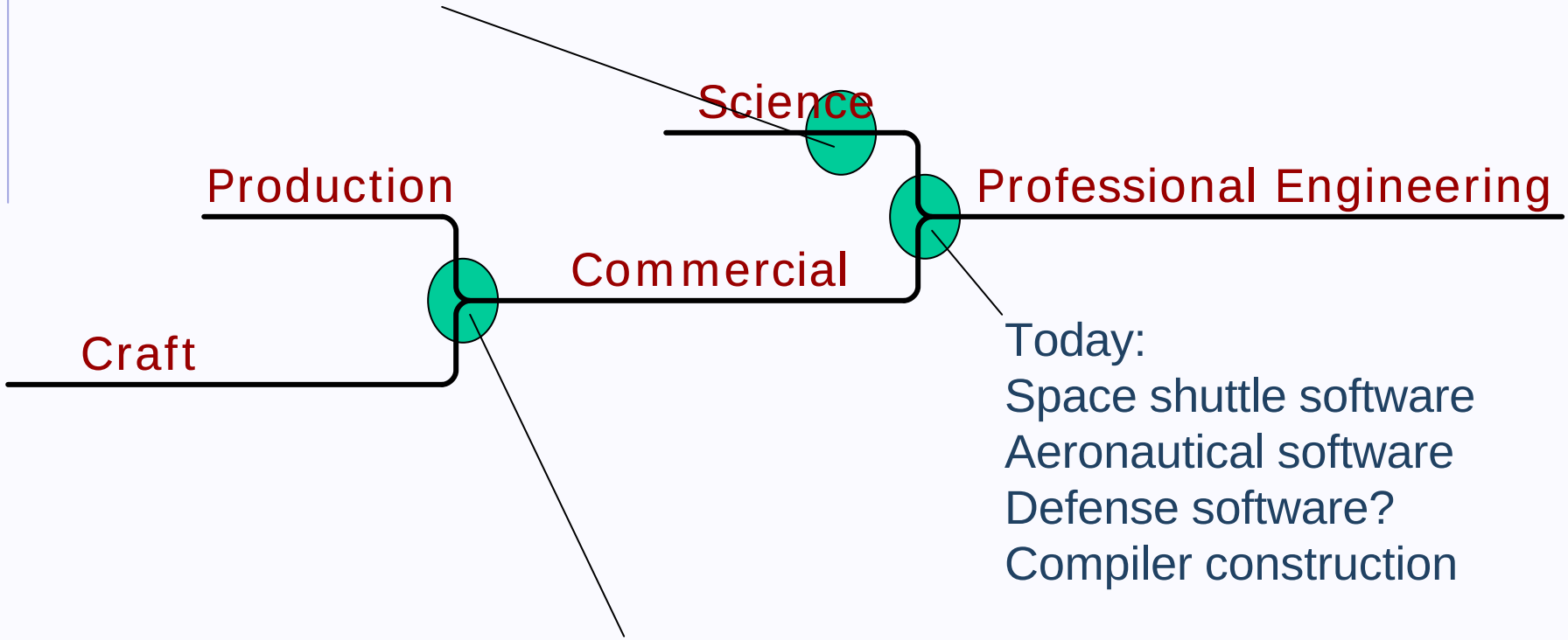
- Allen, F. and Hawthorn, P., “ACM Task Form on Professional Licensing in Software Engineering”, May 1999.
  http://www.acm.org/serving/se_policy/report.html
- CEQB, “Core and Supplementary Bodies of Knowledge for Software Engineering: A report prepared for the CCPE as part of the CEQB Body of Knowledge Development Pilot Project”, Draft version 0.4, Sept. 5, 2001.
- Knight, J., Leveson, N. et al., “ACM Task Force on Licensing of Software Engineers Working on Safety-Critical Software”, Draft July 2000.
  http://www.acm.org/serving/se_policy/safety_critical.pdf
- Notkin,David. “Software Engineering Licensing and Certification”, presentation given at CRA Conference at Snowbird, 2000.
  http://www.cra.org/Activities/snowbird/00/notkin-crawk3-5.pdf
- Parnas, D.L., "Two Positions on Licensing", Panel position statement in Proceedings of the 4th IEEE International Conference on Requirements Engineering , June, 2000.

References

- *The Professional Engineers Act,* Statutes of Ontario, 1990, Chap. P.28
  http://192.75.156.68/DBLaws/Statues/English/90p28_e.htm
- Shaw, Mary, “Prospects for an Engineering Discipline of Software”, *IEEE Software*, November 1990
- UW Software Engineering program
  http://www.softeng.uwaterloo.ca

Mary Shaw's Evolution of an Engineering Discipline

1960s: data structures, algorithms
1990s: design patterns, architecture patterns



Today:
Space shuttle software
Aeronautical software
Defense software?
Compiler construction

1980s: software development processes

SWEBOK (IEEE)

FACT

A multi-year project to formulate a Body of Knowledge that should be mastered by practicing software engineers.

- Normative practices that are applicable to most software products
- Intended to form the basis for Specialization Examinations that P.E. applicants (in the States) take after satisfying their work experience requirements.

Certificates of Authorization

P.Eng.s and corporations who provide engineering services to the general public are required to obtain a Certificate of Authorization.

- Companies must identify the person who will assume responsibility for supervising engineering work and must obtain professional liability insurance.

- An individual engineer who offers engineering services to the public must either obtain liability insurance or must disclose to all clients, in writing, the lack of insurance.