Seminar on Software Engineering

Software Standards

“the good thing about standards is that there are so many of them”

Types of Standards

Coding standards

APIs and protocols

Process standards

Quality standards

Types of Standards

Coding standards

- Capitalize variable names
- Comment block for function
- Where to put braces
- Maximal function length

APIs and protocols

Process standards

Quality standards

Types of Standards

Coding standards

APIs and protocols

- STL and Java libraries
- Special libraries (statistics)
- Access to special hardware
- TCP/IP

Process standards

Quality standards

Types of Standards

Coding standards

APIs and protocols

Process standards

- Software engineering terminology
- Lifecycle models
- Auditing

Quality standards

Types of Standards

Coding standards

APIs and protocols

Process standards

Quality standards

- Guaranteed uptime
- Guaranteed performance
- Quality of results

Types of Standards

Coding standards	Developer mobility
APIs and protocols	Portability and inter-operability
Process standards	Certification, guidelines
Quality standards	Assurances

Software Standards

- Quality standards are rare
  - Hard to specify
  - Vendors typically provide no guarantees
- Process standards dominate
  - Specify what you should do, not what you should achieve
  - Serves as an alternative to real quality standards
  - Used e.g. to select software suppliers
  - Used as insurance against malpractice suits

Taking Standards Seriously

- A standard is a compromise
  - Inputs from industry, regulatory bodies, and maybe academia
  - Result of political process of standardization
  - Subject to economic pressures
- Standards are not necessarily technically superior
- Most important ones are APIs and protocols
  - Software development is largely integration
  - If you're big enough (e.g. Microsoft) you can dictate the de-facto standard