Chapter 1. Intro

Software costs more to maintain than it does to develop.

* 1. **Professional software development**

Program != Software

* Diversity

- There are many different types of software system and there is no universal set of software techniques that is applicable to all of these.

* Fundamentals

- Some fundamental principles apply to all types of software system, irrespective of the development techniques used:

1. Systems should be developed using a managed and understood development process.
2. Dependability and performance are important for all types of system.
3. Understanding and managing the software specification and requirements are important.
4. Reuse software that has already been developed rather than write new software.

* Web

- Software reuse - Idealization! Not so simple!

- Incremental and agile development - Web-based systems should be developed and delivered incrementally.

- Service-oriented systems

* 1. **Software engineering ethics**
* The ACM/IEEE Code of Ethics

- ACM/IEEE - CS Joint Task Force on Software Engineering Ethics and Professional Practices

* Ethical principals

1. Public
2. Client and employer
3. Product
4. Judgement
5. Management
6. Profession
7. Colleagues
8. Self

Chapter 2. Software Processes

Software process descriptions - the activities in processes such as specifying a data model, designing a user interface, etc.

Plan-driven and agile processes

* Most practical processes include elements of both plan-driven and agile approaches.
* There are no right or wrong software processes.

**2.1 Software process models**

The waterfall model - Plan-driven model

Incremental development - May be plan-driven or agile

Integration and configuration - The system is assembled from existing configurable components. May be plan-driven or agile.

* Waterfall model phases

- There are separate identified phases in the waterfall model:

1. Requirements analysis and definition
2. System and software design
3. Implementation and unit testing
4. Integration and system testing
5. Operation and maintenance

- Main drawback: the difficulty of accommodating change after the process in underway

* Incremental development

- when we work**incrementally** we are adding piece by piece but expect that each piece is fully finished. Thus keep on adding the pieces until it’s complete.

* Incremental development problems

- The process is not visible.

- System structure tends to degrade as new increments are added.

Unless time and money is spent on refactoring to improve the software, regular change tends to corrupt its structure.

**2.2 Process activities**

Real software processes are inter-leaved sequences of technical, collaborative and managerial activities.

The four basic process activities of specification, development, validation and evolution are organized differently in different development processes.