Is software different?
- Methodology for building software very different
- Fewer reviews
- Fewer people involved in components
- Software is far more complicated
- Different change and update models

Scalability
- Working with large software systems is inherently different than small systems
- What does 1000000 lines of code look like?
- Difficult to appreciate large scale issues working on a small scale

The bigger picture
- Building software is rarely the true objective
- Even for Microsoft!
- Answering to the customer
  - Complex set of economic and technical constraints

Modeling the Software Process
- Software life cycles
- Waterfalls and Spirals
- Value of modeling
  - Understand process
  - Framework for work process
- Software construction only a portion of the process

Tradeoffs
- Engineering is about resource tradeoffs
- Constraints
  - Money, Time, Resources, Personnel, Environment
- Where to cut back
Risk Management
- Developing software is planning under uncertainty
- Source of risks
- Assessing and tolerating risks
- Risk Mitigation

People factors - Teams
- Increased workforce
- Diversity of Skills
- Economics
- Team organization
- Team dynamics

People factors - Motivation
- This single most important factor in determining team success / failure
- Ways to motivate
- Ways to demotivate

Requirements
- Understand what to build
- Many different formal approaches
- Functional and non-functional requirements

Design
- General understanding of design can help the design of software
- Basic design rules
  - Provide a good conceptual model
  - Make things visible
  - Principle of Mapping
  - Principle of Feedback

Quality assurance
- Independence of QA
- Many different aspects of quality
- Deliverable of QA is information
- Write it down
Testing
- Plan for testing from beginning of project
- Bug management process central in shipping a product

What else????