

INTRODUCTION – CT 216 SOFTWARE ENGINEERING I

Dr. Enda Barrett



OÉ Gaillimh
NUI Galway

Module overview

2

- 2 hours lectures per week
 - ▣ **Thursday 1PM – 3PM IT 125**
- Software Engineering (**Dr. Enda Barrett**) – S1 + S2 (6 weeks)
- Group project – (**Dr. Enda Barrett**) – S1 + S2
- Formal methods (**Dr. Matthias Nickles**) – Semester 2
- Blackboard
 - ▣ Notes
 - ▣ Problem sheets
 - ▣ Assignment submission
 - ▣ Announcements
- Lab Tutors: TBC
 - ▣ Labs start mid-October – **Friday 12 - 2PM IT 106**



Blackboard

Referenced Texts

3

- ❑ **Software Engineering: A Practitioners Approach** by *Roger Pressman, McGraw-Hill*
- ❑ **Practical Software Engineering: A Case Study Approach** by *Leszek A. Maciaszek, Bruce Lee Liong, Addison Wesley*
- ❑ **Professional Node.js: Building JavaScript-Based Scalable Software** by *Pedro Teixeira*
- ❑ **Learning Node.js: A Hands-On Guide to Building Web Applications in JavaScript** by *Marc Wandschneider*

Module Details

4

- Exam at the end of the year (Summer 2017)
 - ▣ No exam at Christmas
 - ▣ 5 questions answer 3
 - 3 on Software Engineering (EB)
 - 2 on Formal Methods (MN)

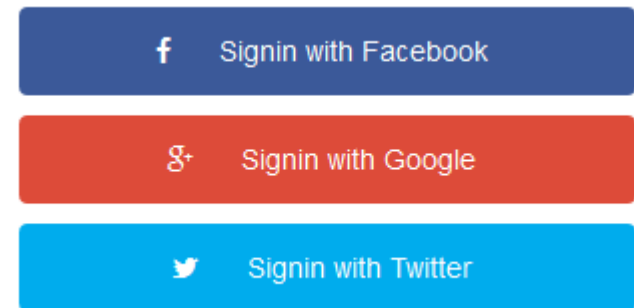
- Group project will account for **40%** of the final mark
 - ▣ Software Engineering deliverable as part of the project
 - Requirements document

Group Projects



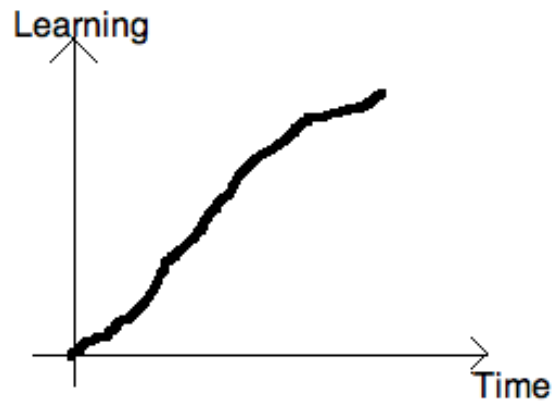
5

- Groups of 4 people
 - ▣ Web based application i.e. You will build a **web application** using (HTML, CSS, JavaScript), you will build the backend in **Node.js** and a data storage component i.e. **MongoDB**
 - ▣ Users will be required to authenticate with the application, preferable using oAuth (Facebook, Google, Twitter).
 - ▣ Past projects included
 - Chat rooms
 - Personal Dashboards
 - Photo sharing application
 - ▣ Some difference between groups!

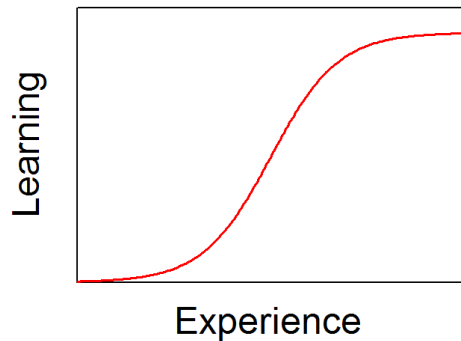


So get thinking about what you would like to do!

- Last year I taught the MeteorJS...



S-Curve (Sigmoid)



Partnership with Blackstone

7

- Hoping to partner with Blackstone LaunchPad



Blackstone
LaunchPad

My details

8

- ❑ Office: IT427
- ❑ Email: Enda.Barrett@nuigalway.ie
- ❑ Phone: 091-493332

Software Development Paradigms

9

- ❑ Waterfall model
 - ▣ Variations of this model, Spiral, Feedback, Overlaps
- ❑ Prototyping
- ❑ Agile Methods
- ❑ eXtreme Programming (XP)

Structured Analysis

10

- Model building activity
 - ▣ Models are created to represent information flow and content within the system
 - ▣ Depict the essence of what must be built

- All about diagrams
 - ▣ Context diagram
 - ▣ Data flow diagrams
 - ▣ Entity relationship diagram

Object Oriented Analysis and Design

11

- Basic principals of object orientation
 - ▣ Encapsulation, Abstraction, Modularity, Hierarchy
- Basic concepts of object orientation
 - ▣ Class, object, operation, attribute, component, generalisation, polymorphism
- UML diagrams
 - ▣ Use case modelling, Activity diagrams, Class diagrams...

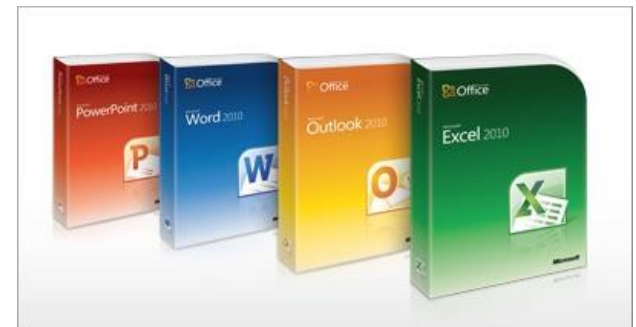
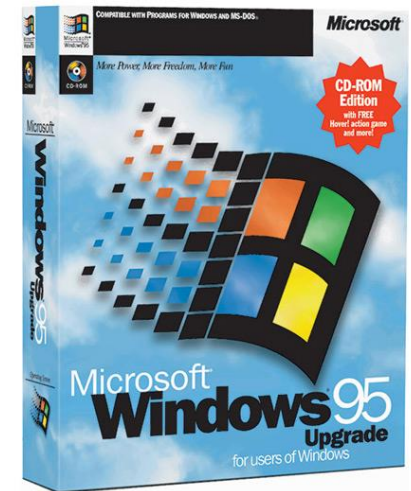
Project technologies

12

- I will cover the technologies that will be used in the project
 - ▣ Database component (MongoDB)
 - ▣ Client side programming (HTML, CSS, JavaScript)
 - ▣ Server side programming (JavaScript (Node.js))
- Sample code, walkthroughs etc.

Software

13



Software is everywhere

14



Not to mention...

15

[Gmail](#) [Images](#)



[Sign in](#)

The Google logo, featuring the word "Google" in its characteristic multi-colored font.

Google Search

I'm Feeling Lucky

And all of these...

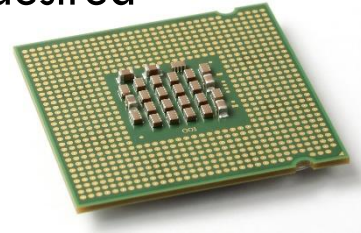
16



What is Software?

17

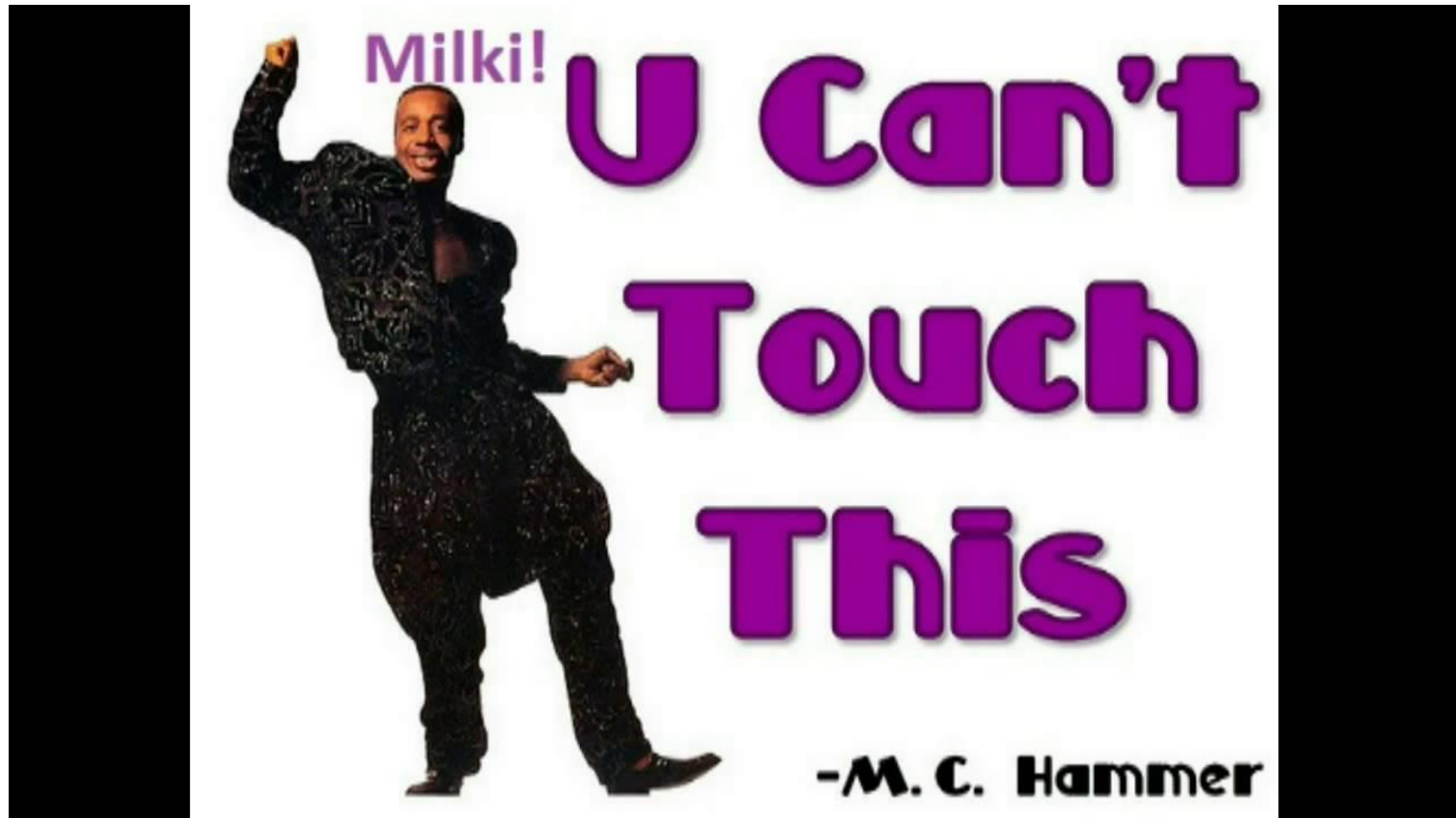
- ❑ A set of instructions that tell your computer what to do!
- ❑ Textbook definition
 - ▣ Instructions (computer programs) that when executed provide desired features, functions and performance
 - ▣ Data structures that enable programs to manipulate the information
 - Arrays, Objects, List, Dictionary, Map
 - ▣ Descriptive information in both Hard Copy and virtual format describing the operations and use
- ❑ Probably one of the most important pieces of technology on the world stage.



What is Software...

18

- ❑ Software is a *logical* rather than *physical* component of computer based systems.



What else does it mean?

19

- You can't taste it...



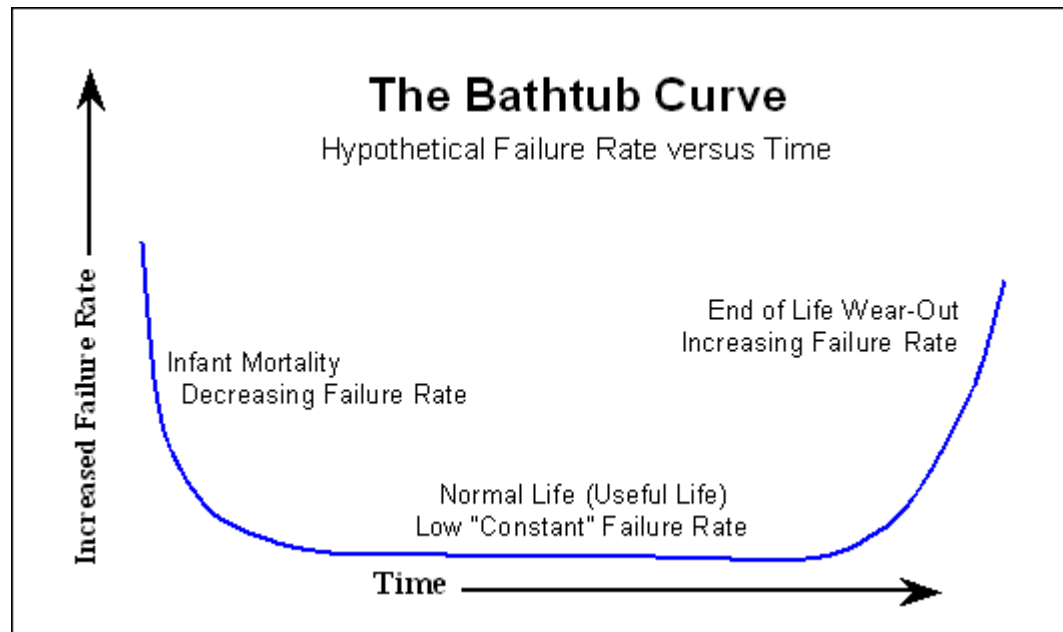
What else does it mean

20

- ❑ Software has characteristics that are considerably different than those of hardware:
 - ▣ Software is developed or engineered; it is not manufactured in the traditional sense.
 - ▣ Software doesn't "wear out".
 - ▣ It is "constructed" using abstract notions of logic (programs/instructions). There is no physical material to degrade or wear-out – so once software functions according to the specification it should remain that way indefinitely.

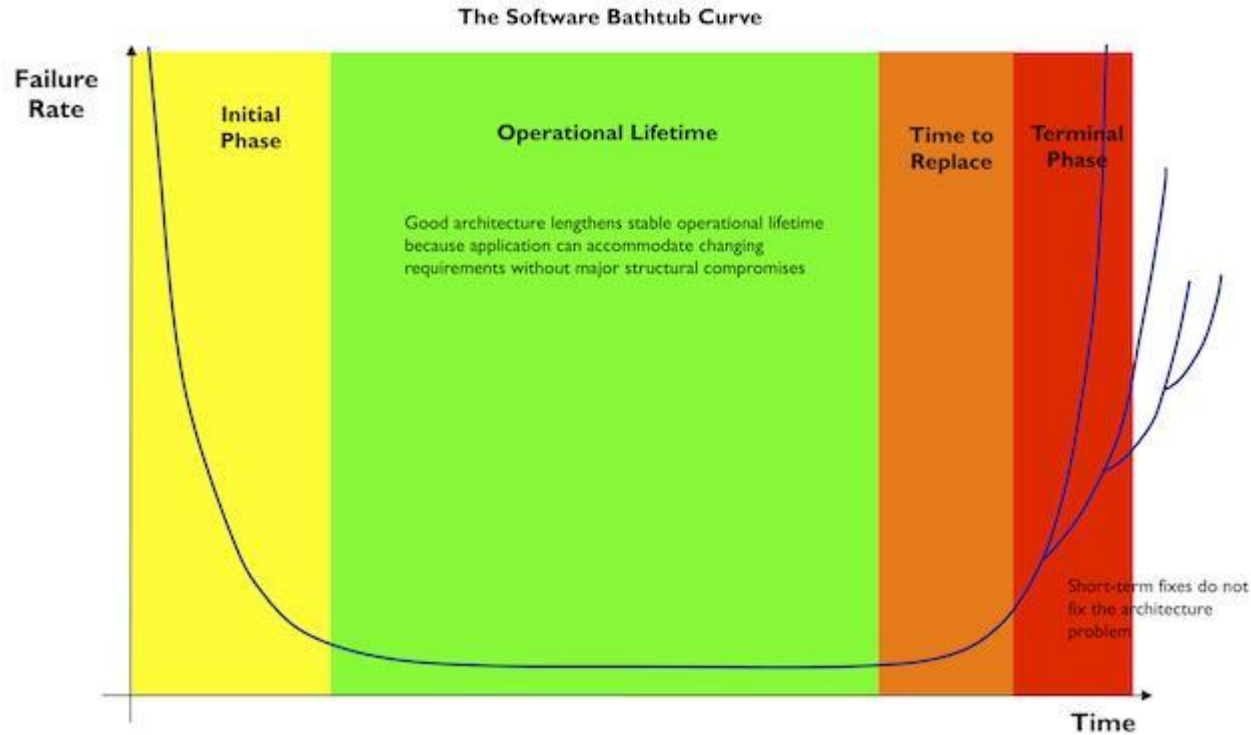
Failure curve for Hardware

21



Failure curve software

22



Software Systems are different

23

- ❑ Exhaustive testing is impossible
 - ▣ All Software is shipped with bugs!
- ❑ Issue only arises in a small number of infrequently encountered use cases
- ❑ Some bugs are features
 - ▣ Behaviour that constitutes a failure in one system may be a feature in another.
- ❑ Microsoft public bug fixing

Examining the manufacturing process...



Why can't software be manufactured?

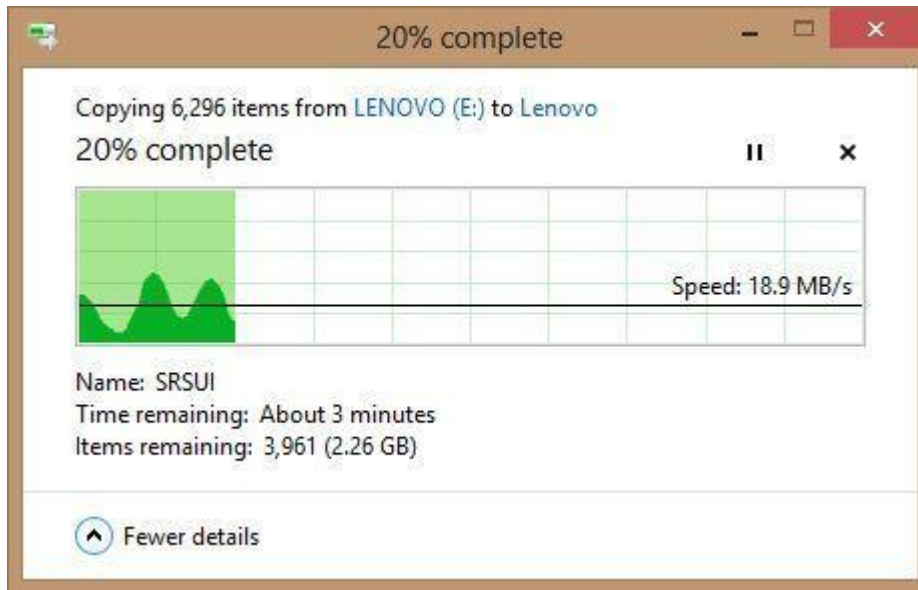
26

Software Development Lifecycle



Software manufacturing...

28



Software Engineering

29

- We generally refer to Software being developed or engineered.
- This module is all about the theory behind that with some practical exercises also.
- Engineering large software systems is hard!



Some notable Software Failures

30

- In February 2003, a computer software blunder at St. Mary's Mercy Medical Centre in Grand Rapids, Michigan, cost the lives of 8,500 patients.



- Patients, who had procedures done from October 25 through December 11 of the previous year, were alive and kicking. However, the glitch, attributed to the hospital's patient management system, notified **Social Security, patient insurance companies, and the patients** themselves, of the “unfortunate” demises.

Knight capital

31

- Knight Capital Group, a market-making firm that until August 2012 had a stellar reputation in its industry, blew all of that in about **30 minutes**. Between 9:30 a.m. and 10 a.m. EST on August 1, the company's trading algorithms got a little buggy and decided to buy high and sell low on 150 different stocks. By the time the bleeding had stopped, KCG had lost \$440 million on trades. By comparison, its market cap is just \$296 million and the loss was four times its 2011 net income.



Ariane 5 rocket

32

- ❑ In essence, the software had tried to cram a 64-bit number into a 16-bit space. The resulting overflow conditions crashed both the primary and backup computers (which were both running the exact same software). The Ariane 5 had cost nearly \$8 billion to develop, and was carrying a \$500 million satellite payload when it exploded.



The “engineering” of Software

HIGH RISK
AREA

33

- The development of large applications in excess of 5000 function points (~500,000 LOC) is one of the most risky undertakings in the modern world (Capers Jones)
- The risks of cancellations or major delays rise rapidly as overall application size increases (Capers Jones)
 - ▣ 65% of large systems (over 1,000,000 LOC) are cancelled before completion
 - ▣ 50% for systems exceeding half a million LOC
 - ▣ 25% for those over 100,000 LOC
- Failure or cancelation rate of large software systems is over 20%
- LOC is not used very often in industry any more
 - ▣ Your bonus will not be defined on how many “lines” you’ve written

The “engineering” of Software cont.

34

- ❑ Of completed projects, 2/3 experience schedule delays and cost overruns (Capers Jones)
- ❑ 2/3 of completed projects experience low reliability and quality problems in the first year of deployment (Capers Jones)
- ❑ Software errors in fielded systems typically range from 0.5 to 3.0 occurrences per 1000 lines of code (Bell Labs survey)

Expected/Delivered benefit from Software Development

35

- ❑ Software development projects deteriorate during the lifetime of the project
 - ▣ Take twice the estimated time
 - ▣ Cost twice the original sum
 - ▣ Deliver half of what was promised
- ❑ Delivered/Expected ratio is 1:8
- ❑ Software Engineering to the rescue



Types of problems in Software Projects

36

- ❑ Feature (scope) creep
 - ▣ Engineering manager “You can add in something when you take something else off the list”
- ❑ Thrashing
 - ▣ Novices are more guilty than experts, as experts realise quicker when they are out of their depth
- ❑ Tolerating
 - ▣ Implementing temporary “workarounds” which grow and grow
- ❑ Compromising
 - ▣ Settling for sub-optimal solutions in order to move work along
 - ▣ Serial only and no USB support
- ❑ Integration problems
 - ▣ Lack of communication at development time
- ❑ Overwriting source code
- ❑ Redesign and rewriting during test
- ❑ No documentation of design decisions

What is software engineering?

37

□ What is Software Engineering?

- ▣ “Software Engineering is the field of computer science that deals with the building of software systems that are so large or so complex that they are built by a team or teams of engineers (Ghezzi et. al. 2003)”
- ▣ “Software Engineering encompasses a process, a collection of methods and an array of tools that allow professionals to build high quality computer software” (Pressman)

