Enterprise Systems
(and extended)

Lecture 01:
Introducing the course

Behzad BORDBAR
Enterprise systems (ES) deals with engineering of

- Complex, large-scale, Internet-based systems consisting of
- Integrated services and components software and hardware
- Dealing with computation, data, transactions to support
- Business processes within and between organizations.
- Ability to process large amount of data
Example : Cloud

G-cloud
Virtualisation and Virtual machine
Virtual network
can handle huge amount of data
can reduce cost and energy
Security of Cloud (collaboration with HP)
How are these hugely distributed systems are engineered?
Example: Big Data

Processing huge amount of data
• Moving to graph-based databases
• Hadoop
• Diagnosability (storm)

How are these hugely distributed systems engineered?
Idea of ES:

Some of (not all) key advances in software:

• Transactions
• Object relational mapping
• Aspect-oriented programming
• Dependency injection

Extended Enterprise Systems, previous years one of:

• Business processes and workflow management
• Service oriented Architecture
• Cloud and Rule Based Systems
• Automated code generation (map noSQL last year)
Lectures deal with Multi-tiered systems:

• But the services can be made of such blocks
• Can be deployed in VMs within cloud
Vision of ES:

Check http://www.cs.bham.ac.uk/~bxb/Teaching.html
Admin info

Check out website:
http://www.cs.bham.ac.uk/~bxb/Teaching.html

• Pay attention assessment different for
  - Enterprise Systems (undergraduate)
  - Extended Enterprise Systems (MSc)

Don’t forget to sign in when attending a lab session

Exercises are not marked, they are to assist you in learning

Module page has copies of slides, source code announcements, … at the website
You will see two examples

1) Book shop which is used within the course of lectures

2) An Asset Management System example which is used in the exercises
   - Very simplified (Why?)
   - Small for you to work with

3) A music management system for your assessment
Assessment for the lectures

I will explain how it is done!

Course work (70% of mark for MSc and 100% for Undergraduates)

Three examinations

• JDBC
• Hibernate
• Spring
Be Aware!

• ES deals with complex technologies
• Is about complex technology such as ORM, component based software engineering, Aspects, Dependency Injection,...

As a developer, a complex technology is your friend! YOU can build cheaper systems which are robust and are easier to maintain

• Open source: you must work your way around it... So you must work hard!

• You will learn ES only by doing the exercises and lots of programming.
Be Aware!

• ES deals with important technologies

Important?

• Is about complex technology such as ORM, component based software engineering, Aspects, Dependency Injection,...

As a developer, a complex technology is your friend! YOU can build cheaper systems which are robust and are easier to maintain

• Open source: you must work your way around it...
So you must work hard!

ES is ONLY suitable for people with strong JAVA programming skills- no resit (MSc)!
Assessment of group proj. (ONLY MSc)

- Divide u to groups of 5ish students

There are two deliverables:

1) 30-45 minutes presentation/lecture (last week of the term)

2) a complete piece of software

- Weekly supervision meetings with me
- you will develop a working software and demonstrate it during the last week of the term. This heavy work!
What is the presentation/lecture about?

Describes your findings and their relationship with the rest of the course:

• motivation behind the creation of the technologies, its strength and weaknesses
• competing technologies.
• Architecture of any tool any API,
• any methodology used and explain your personal evaluation of the technology.
Important dates:

• See
  http://www.cs.bham.ac.uk/~bxb/Teaching.html

• Let us go through them
Topics

• Service oriented Architectures: learn about Business Processes, Workflow Management, BPEL, BPMN and development of SOA applications via Oracle JDeveloper or Websphere or any other (may be open source) framework. Then you will develop a working system.

• Programming in Cloud: Detailed study a Cloud platform provided by a cloud vendor during which you will explore the platform in details, learn its API and develop a working application.

• **Big data:** learn about mapReduce and Hadoop, PIG, ... or platform dependent developments twitter (storm),... <<< my fav. this year
Part 1 JDBC

We assume following installed on your machines:

- Java (at least 6)
- PostgreSQL

However, very little of what follows is specific to PostgreSQL and almost no changes are required, except for the connection details, for Microsoft Access or MySQL and Apache Derby, SQLite (embedded database)
PostgreSQL

PostgreSQL is free, open source multi OS DBMS

If installing PostgreSQL on your own machine, the version of the JDBC driver and PostgreSQL must match. See http://jdbc.postgresql.org/download.html.
PostgreSQL

• Linux: almost always comes with the distribution. For ubuntu see https://help.ubuntu.com/community/PostgreSQL
• Windows: http://pginstaller.projects.postgresql.org
• Macs: Different ports of PostgreSQL to OSX:
  http://www.macports.org/
  http://www.postgresqlformac.com
  http://www.entropy.ch/software/mac-osx/postgresql

Not being a mac user, I have not tried any!

Any experience?
Home work

• Install PostgreSQL and make it work (if you want to use your own machine- good idea to do this anyway)
• Learn how to use it: start, stop, run commands, use of GUI (pgAdmin),…
• Learn basic SQL: SELECT, …
• Do a couple tutorials
• I expect you to be good fluent in PostgreSQL by next week.
IDEs

Learn to use an IDE,
Eclipse (www.eclipse.org),
NetBeans (netbeans.org)
IntelliJ IDE (www.jetbrains.com/idea/).

**Choose an IDE and learn how to use it properly**

I use Eclipse (installed on the School machines)

- For your copy go to www.eclipse.org/downloads/, look for the link for “Eclipse IDE for Java Developers”
- Lars Vogel has a good tutorial starting from basics, see http://www.vogella.de/articles/Eclipse/article.html
- MarK Dexter has recorded six elementary lectures (each 15 mins). See http://eclipsedailystandard.sourceforge.net/workbench01/lesson01.html for the first lecture.