Advanced Software Development & Engineering
Theme Introduction

Complexity

- Software Development is Complex
  - *Consisting of interconnected or interwoven parts; composite (Oxford Dictionary)*
- *Complex as in :*
  - *Composed of third party artifacts*
  - *Combining different environments, information sources, communication media*
  - *Written by a team of people with their own thought process*

Complication

- Software development need not be complicated
  - *Not easy to understand or analyze (Oxford Dictionary)*
- Complicated as in :
  - Difficult to understand or setup
  - Difficult to improve
  - Difficult to exploit

Our objective

- Presenting expert solutions :
  - Many problems are recurrent
  - And many of those have been solved in a generic and efficient manner
- Presenting the tools we use for the job
  - To help you organise your work
  - To give you ideas
- Exposing our best practices (bear with us)
- Remain platform agnostic as far as possible

Syllabus

- Design block
  - Introduction to Entreprise Computing
  - Design patterns
- Integration block
  - Security In Computer Applications
  - Iterative Development
- Maintenance block
  - Debugging Techniques
  - Code Review Best Practices

Introduction to Enterprise Computing

- Enterprise : *An undertaking, especially one of some scope, complication, and risk (Cambridge Dictionary)*
- Enterprise aims at dealing with complexity while remaining open :
  - Distributed, multi-tier environment
  - Diversity of information sources
- Ready made solutions exist (J2EE, .Net)
- Knowing about existing solutions may avoid you reinventing the wheel

Design patterns

- Efficient conceptual reuse
- Aimed of object oriented languages
- Expert solutions that solve your problems
- Help to understand how a system works
- Help to identify a solution and how to extend it

Security In Computer Applications

- Apprehending “software security”
- Evaluating security risks
- How to create secure software
- Advices and warnings for developers (what to do, not to do)
- What is the best usage of cryptography ?

Iterative Development

- Deals with complexity little by little
- Helps you manage teamwork
- Makes your software easier to maintain and integrate
- Bundles best practices in one neat package

Semi-Interactive Demo

- **Thursday at 3pm**
  Build integration demonstration
  - Turn an existing java application into an integrated build
  - Cover all steps of iterative development :
    - Compilation
    - Testing
    - Deployment
    - Automated Build
  - You can join by following the instructions on:
    - http://tinyurl.com/5h256
  - “Java experience required, web application development a plus”

Debugging Techniques

- Debugging methods to :
  - Localise...
  - Evaluate...
  - And repair defects
- Compiler features
- Debugging and tracing tools

Code Review Best Practices

- Efficient understanding of third-party code
- Learning from existing code
- Automation of code reading using
  - Standard OS tools
  - Dedicated code documentation tools

Panel discussion

- **Thursday 4.30pm**
  Theme : "Are novel Software Development techniques relevant to HEP?“
  - iCSC lecturers
  - 2 Senior Panelists
  - You the audience