

Levels of Testing

Patrick Oladimeji

Advance topics in Computer Science

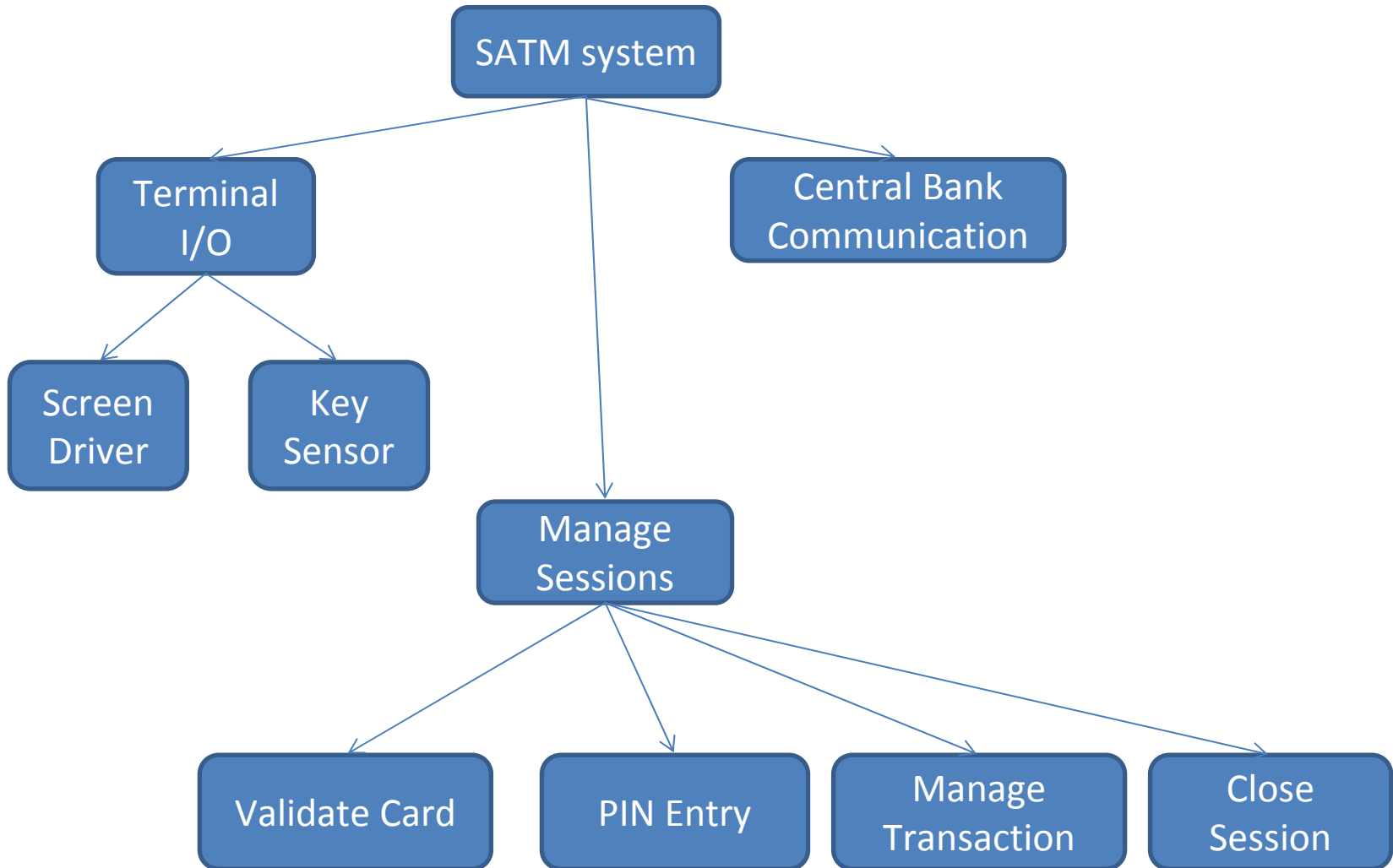
Dr. Markus Roggenbach
Prof. Dr. Holger Schlingloff

University of Wales Swansea Computer Science Department

Contents

1. Different levels of testing.
2. Important factors on testing levels.
 - Factors influencing test scope
 - Why test at different levels?
3. Test levels and software life-cycle models.
 - “V” Model

Decomposition tree of SATM



1. Different Levels of testing

What is a level of test?

- Defined by a given **environment**
- environment is a collection of people, hardware, software, interfaces, data etc.

Attribute	Level			
	Unit	Integration	System	Acceptance
People	Developers	Developers & Testers	Testers	Testers & Users
Hardware O/S	Programmers' Workbench	Programmers' Workbench	System Test Machine or Region	Mirror of Production
Cohabiting Software	None	None	None/Actual	Actual
Interfaces	None	Internal	Simulated & Real	Simulated & Real
Source of Test Data	Manually Created	Manually Created	Production & Manually Created	Production
Volume of Test Data	Small	Small	Large	Large
Strategy	Unit	Groups of Units/Builds	Entire System	Simulated Production

Fig 1 – Sample environment variables

[Systematic Software testing- Rick D. Craig and Stefan P. Jaskiel-2002]

1. Levels of testing

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing
- Regression testing

Contents

1. Different levels of testing.

- **Unit Testing**
- Integration Testing
- System Testing
- Acceptance Testing
- Regression Testing

2. Important factors on testing levels.

- Factors influencing test scope
- Why test at different levels?

3. Test levels and software life-cycle models.

- “V” Model

1.1 Unit Testing

- A unit is smallest testable piece of software
 - can be compiled, linked, loaded
 - e.g functions/procedures, classes, interfaces
 - normally done by programmer
 - Test cases written after coding
- Disadvantage
 - Test cases -written to suit programmer's implementation (not necessarily specification)
- Better to use “Buddy Testing”

1.1 Unit Testing

Buddy Testing

- Team approach to coding and testing
- One programmer codes the other tests and vice versa
 - Test cases - written by tester(before coding starts). Better than single worker approach
 - Objectivity
 - cross-training
 - Models program specification requirement

1.1 Unit Testing

- Normally in programmers IDE (comfort zone)
- Find unit bugs
 - Wrong implementation of functional specs
- SATM Example
 - Testing function procedures e.g. the ValidatePIN() procedure

Contents

1. Different levels of testing.

- Unit Testing
- **Integration Testing**
- System Testing
- Acceptance Testing
- Regression Testing

2. Important factors on testing levels.

- Factors influencing test scope
- Why test at different levels?

3. Test levels and software life-cycle models.

- “V” Model

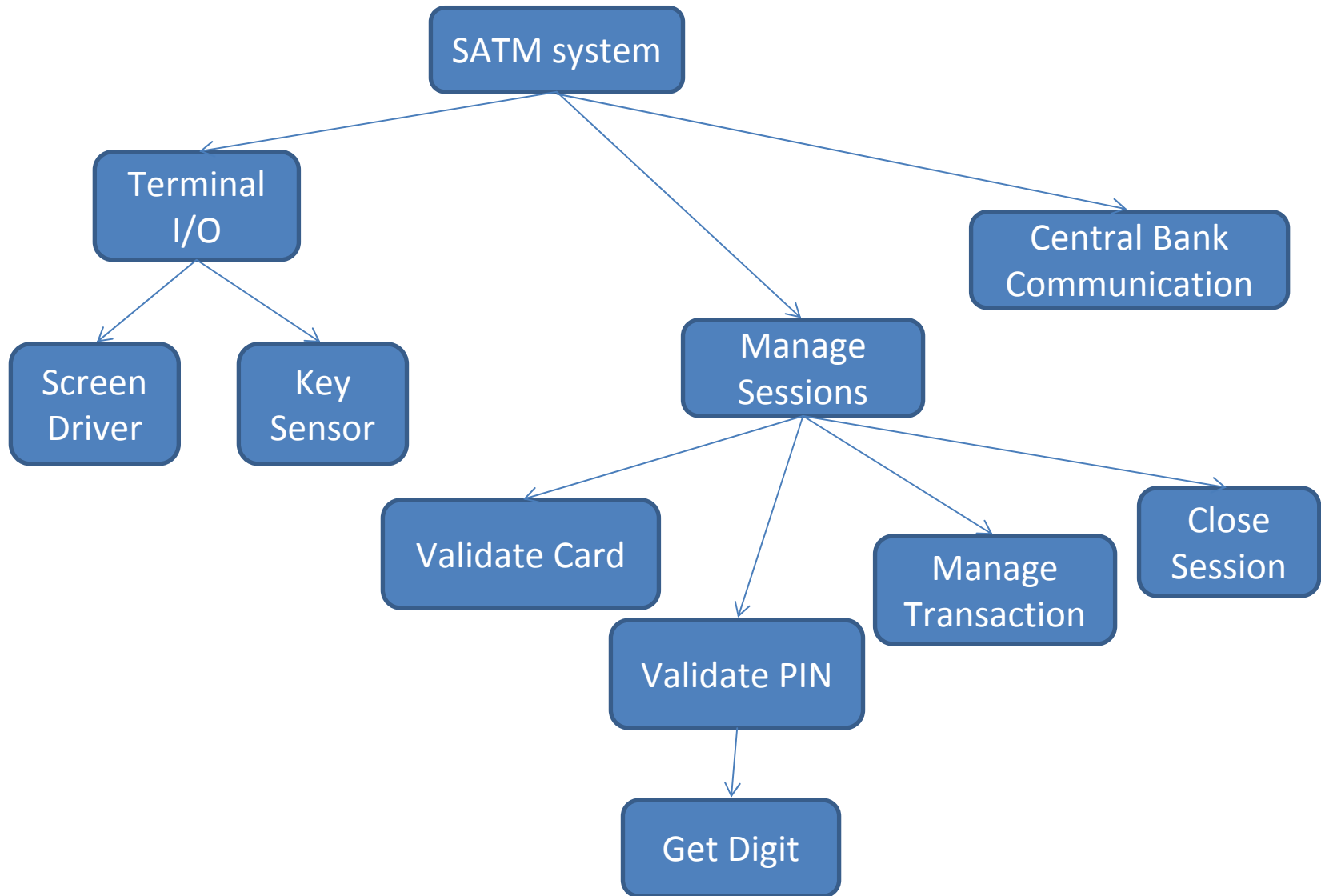
1.2 Integration testing

- Test for correct interaction between system units
- systems - built by merging existing libraries
- modules coded by different people
- Mainly tests the interfaces among units
- Bottom up integration testing
 - Use of drivers
- Top down integration testing
 - Use of stubs

1.2 Integration testing

- Who does integration testing and when is it done?
 - Done by developers/testers
 - Test cases written when detailed specification is ready
 - Test continuous throughout project
- Where is it done?
 - done on programmer's workbench
- Why is it done?
 - Discover inconsistencies in the combination of units.

1.2.1 Stubs and Drivers



Contents

1. Different levels of testing.

- Unit Testing
- Integration Testing
- **System Testing**
- Acceptance Testing
- Regression Testing

2. Important factors on testing levels.

- Factors influencing test scope
- Why test at different levels?

3. Test levels and software life-cycle models.

- “V” Model

1.3 System Testing

- Test of overall interaction of components
- Find disparities between implementation and specification
- Usually where most resources go to
- Involves – load, performance, reliability and security testing

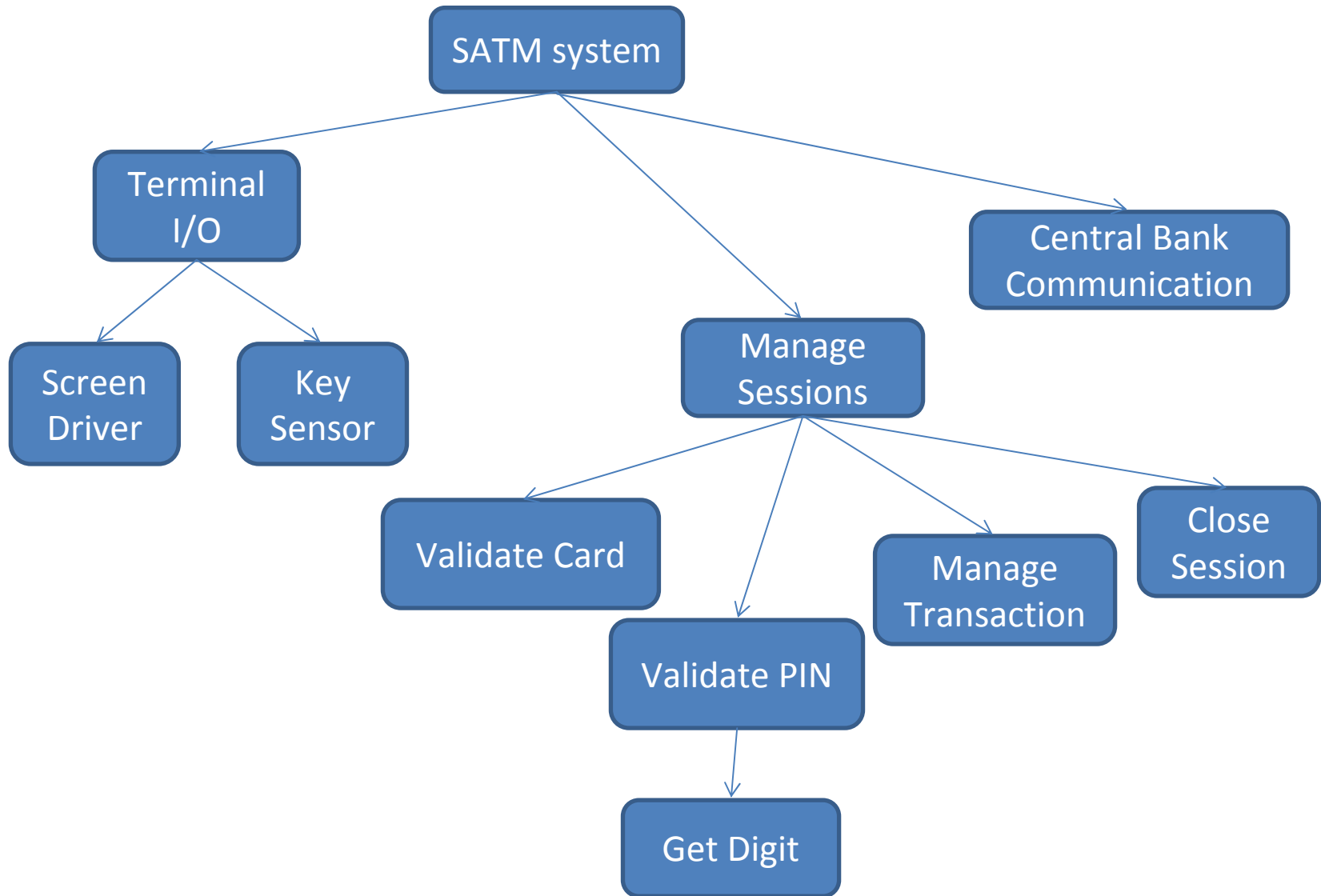
1.3 System Testing

- Who performs system testing and when is it done?
 - Done by the test team
 - Test cases written when high level design spec is ready
- Where is it done?
 - Done on a system test machine
 - Usually in a simulated environment e.g. vmware

1.3.1 System vs. Integration testing

- What vs. How
- Requirement spec-> what
- detailed design spec->how
- System testing functional not structural

1.3.2 Example: System/Integration Testing



Contents

1. Different levels of testing.

- Unit Testing
- Integration Testing
- System Testing
- **Acceptance Testing**
- Regression Testing

2. Important factors on testing levels.

- Factors influencing test scope
- Why test at different levels?

3. Test levels and software life-cycle models.

- “V” Model

1.4 Acceptance Testing

- Demonstrates satisfaction of user
- Users are essential part of process
- Usually merged with System Testing
- Done by test team and customer
- Done in simulated environment/real environment

Contents

1. Different levels of testing.

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing
- **Regression Testing**

2. Important factors on testing levels.

- Factors influencing test scope
- Why test at different levels?

3. Test levels and software life-cycle models.

- “V” Model

1.5 Regression Testing

- On going process throughout testing lifecycle
- New bug-fix breaks previously tested units?
- Perform regression test whenever program changes

Contents

1. Different levels of testing.

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing
- Regression Testing

2. Important factors on testing levels.

- **Factors influencing test scope**
- Why test at different levels?

3. Test levels and software life-cycle models.

- “V” Model

2.1 Factors influencing test scope

- Size of project
- Complexity of project
- Budget for project
- Time scope for project
- Number of staff

2.2 Why test at different levels

- Software development naturally split to phases
- Easily track bugs
- Ensures a working subsystem/ component/ library
- Software reuse more practical

Contents

1. Different levels of testing.

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing
- Regression Testing

2. Important factors on testing levels.

- Factors influencing test scope
- Why test at different levels?

3. **Test levels and software life-cycle models.**

- **“V” Model**

3.1 The “V” model and test levels

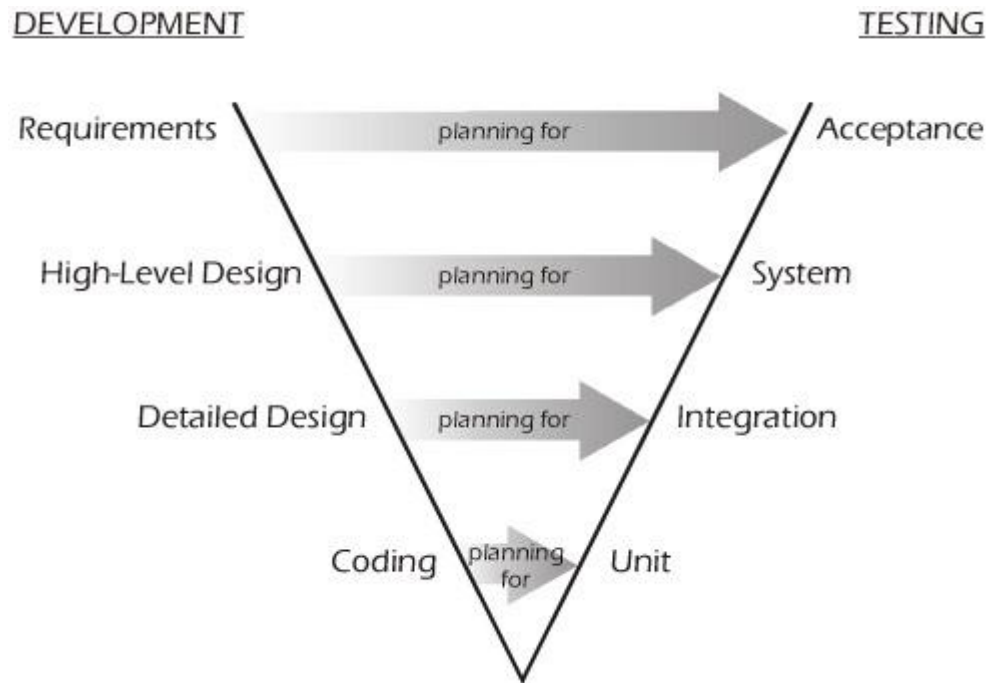


Fig 3.1 levels of testing

[Systematic Software testing- Rick D. Craig and Stefan P. Jaskiel-2002]

Summary

- Different levels of test
 - Unit testing
 - Integration testing
 - System testing
 - Acceptance testing
 - Regression testing
- Factors affecting test scope
- Why test at different levels
- The “V” model and testing