



Lars Wahlberg



92-97 Saab Military Aircraft
97-99 FMV
99-06 OMX
06-10 Cinnober FT
10- ... AddQ

lars.wahlberg@addq.se

*Bilder på Engelska, blir så
mycket Svengelska annars!*

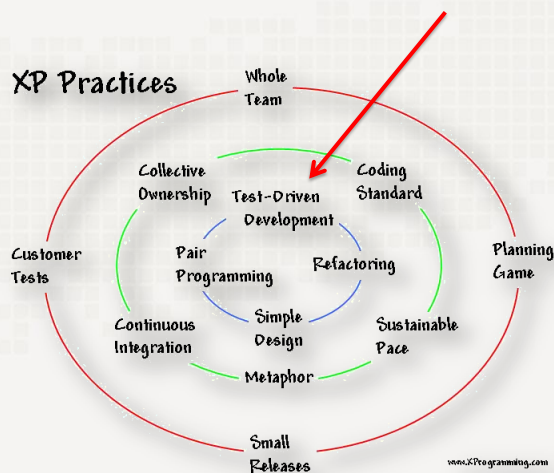
Add
CONSULTING

What is TDD?

- Test method? Not primarily, but we get tests 😊
- Technique for **design** of software
- A way to **document** software
- Should be learned and **practiced**



XP Practises

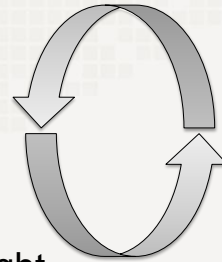


www.xprogramming.com



TDD Basics

- RED – Write a test
- GREEN – Make it run
- REFACTOR – Make it right



Add
CONSULTING

Write a test ●

- How would you like the operation?
- Write a story!
- Invent the interface you wish you had!
- Include all elements needed for answer!


Add
CONSULTING

Make it run

- Quickly make it Green!
- Use simple (dirty) solution!
- Quick green excuses all sins!
- But only for a moment!



Make it right

- Remove introduced duplication
 - Code
 - Data
 - Structure
 - Logic
 - Etc
- Make sure it still is Green 



Demo



TDD in essence

Architectural Driven Design



Clean Code



Then ...



Make it work

Test Driven Design



Make it work



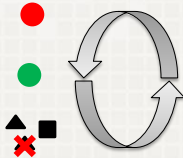
Then ...



Clean Code



TDD Master small steps!



How small iterations/steps **CAN YOU** do?

BE ABLE to do small (silly) steps!

But you **DO NOT** always need to use it!



Yes We Can!

Add
CONSULTING

Test Infected

Once infected you might change attitude ...

- 100% OK Tests
- Tests instead of debugging
- Be able to refactor aggressively with confidence (=sleep well) !
- Want others to write tests!



Add
CONSULTING

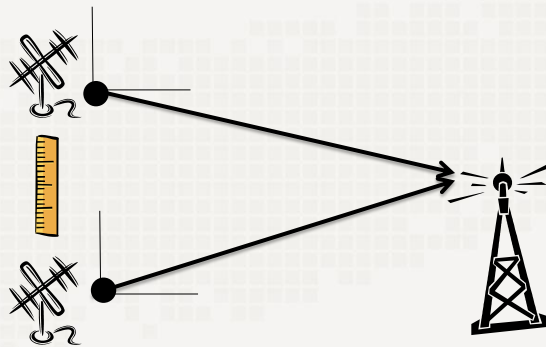
Dependency & Duplication

Duplication

Dependency



Triangulation



See the problem from another direction!



Benefits



- Studies indicate that Developers that using TDD are more productive
- Programmers rarely need to use debuggers
- Interface before implementation
- TDD allows small steps when required
- Unit test code larger, Implementation code smaller
- No more code written than necessary



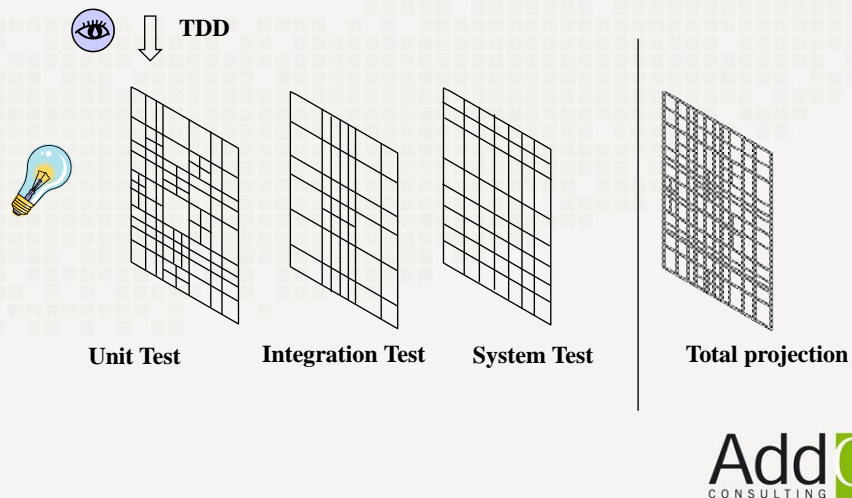
Vulnerabilities



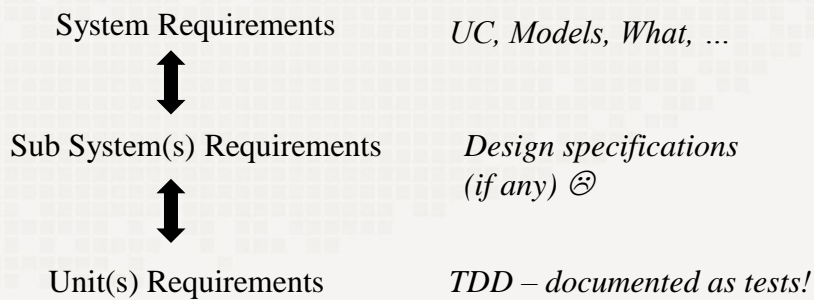
- TDD difficult for applications where functionality resides in many systems
- Tests needs maintenance
- TDD test cases may be influenced by developers idea on structure (i.e. white box)
- If Developers misinterprets the requirement specification for the unit, code and tests are wrong!
- The high number of passing unit tests may bring a false sense of security



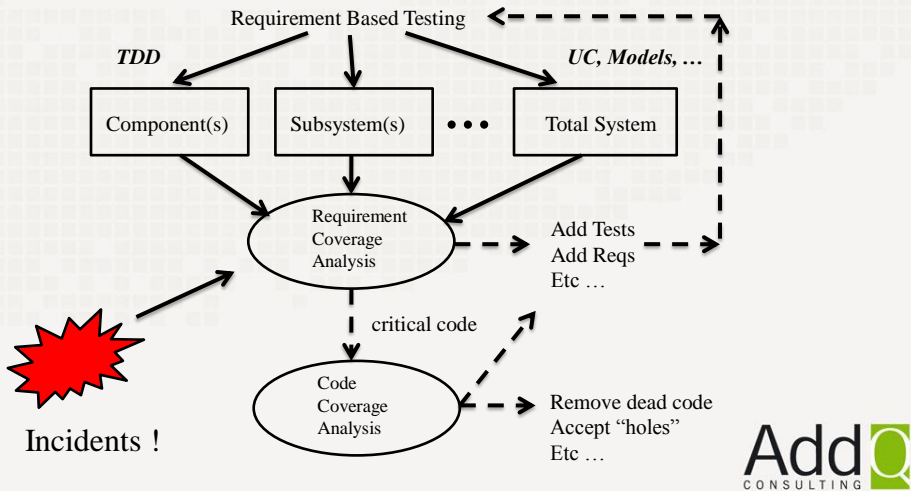
TDD in Test



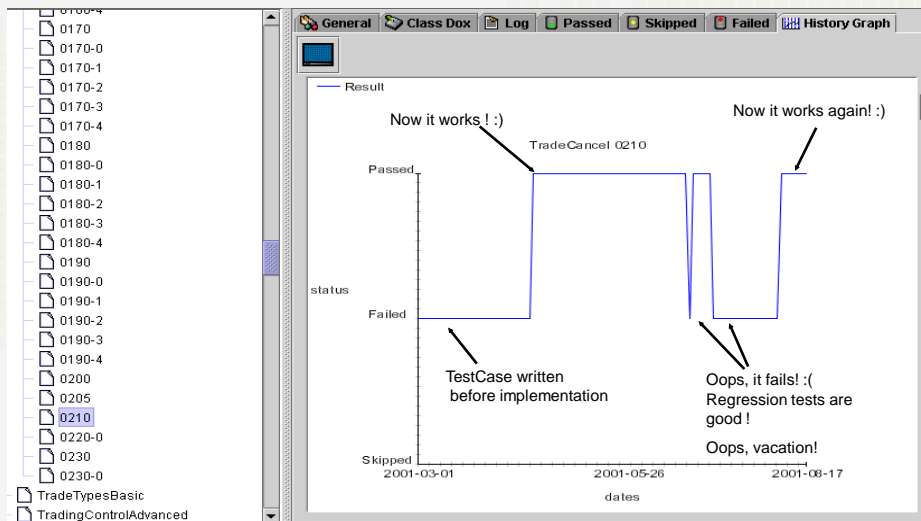
Requirements



Complete Test Process



TDD – System Test



Code Coverage

- Statement Coverage (every line / block)
- Decision Coverage (every if decision)
- *MC/DC Coverage* (used in aerospace)

Execute Requirement Based Tests

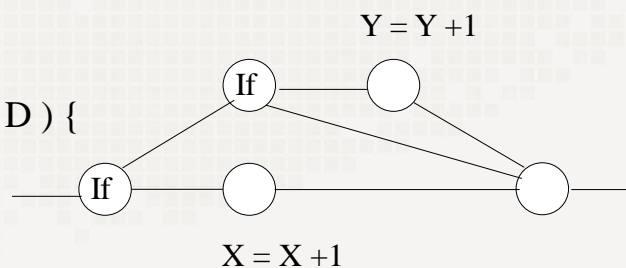
THEN

measure Code Coverage



Code Coverage - Example

```
if (A & B) {
  X=X+1;
} else if (C | D) {
  Y=Y+1;
}
```



Code Coverage - Answers

Tests Needed

Statement Coverage: 2 Tests

Decision Coverage: 3 Tests

MC/DC Coverage: 4 Tests



Emma plugin for Eclipse

The screenshot shows the Eclipse IDE with the Emma plugin. The main editor displays the following Java code:

```

public boolean addAll(int index, Collection c) {
    if (c.isEmpty()) {
        return false;
    } else if (size == index || size == 0) {
        return addAll(c);
    } else {
        Listable succ = getListableAt(index);
        Listable pred = (null == succ) ? null : succ.prev();
        Iterator it = c.iterator();
        while (it.hasNext()) {
            pred = insertListable(pred, succ, it.next());
        }
        return true;
    }
}

```

The bottom panel shows the following coverage data table:

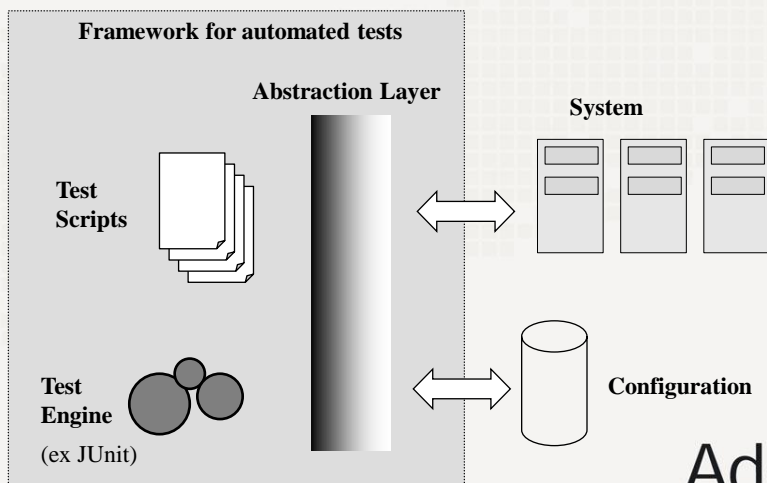
Element	Coverage	Covered Lines	Total Lines
java - commons-collections	79.5 %	10927	13738
org.apache.commons.collections	74.1 %	3942	5183
ArrayStack.java	86.9 %	32	37
BagUtils.java	86.7 %	13	15
BeanMap.java	72.4 %	155	214
BinaryMaps.java	87.6 %	127	145
BoundableBuffer.java	93.2 %	82	88
BufferOverflowException.java	55.6 %	5	9
BufferUtils.java	88.9 %	0	9
BufferUtils.java	30.8 %	4	13
CloseUtils.java	93.9 %	31	33
CollectionUtils.java	92.4 %	290	317
ComparatorUtils.java	84.4 %	3	36
CursorableLinkedList.java	85.4 %	444	520



Demo

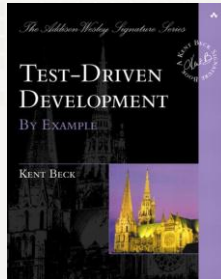


TDD for Abstraction Layer



References

- http://en.wikipedia.org/wiki/Test-driven_development
- Test Driven Development – By Example
Kent Beck



Add
CONSULTING

Summary

- Repeat

RED	Write a test	●
GREEN	Make it run	●
REFACTOR	Make it right	▲ ■
		✘
- Primarily a design method
- But contributes to the total testing
- Gives 100% statement coverage

Add
CONSULTING