

Two words:
improve workflow

Testdriven utveckling

Magnus Jonsson
Siemens Medical Solutions

Soarian

- Stort projekt, ca 400 personer i projektet
- Distribuerad utveckling i USA, Indien och Sverige
- Web baserat lösning med admin client
- Service orienterad arkitektur
- C++, MS SQL
- UI - Java script utan logik
- Admin client VB

3 Koncept

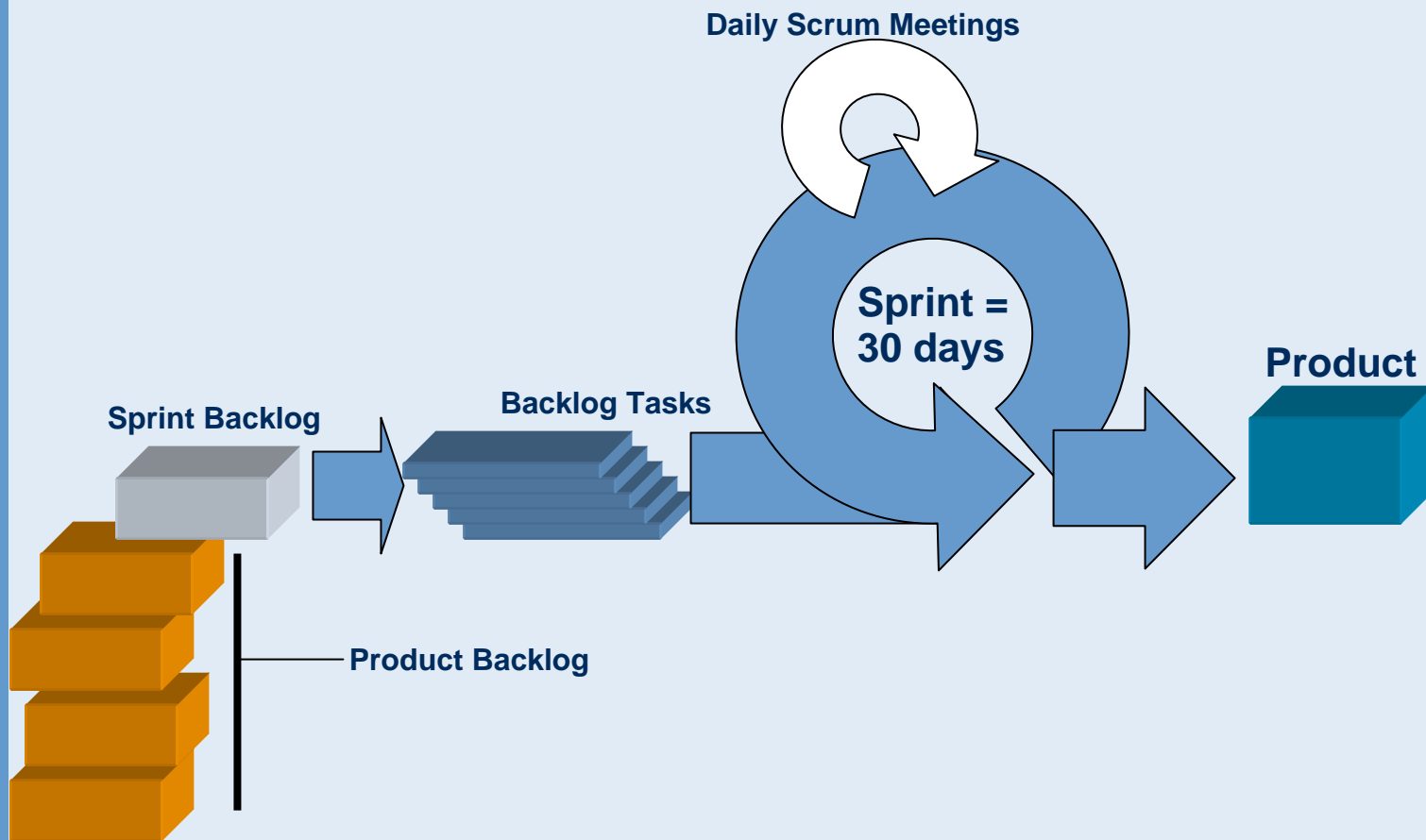
- Agile med Scrum
- Continuous integration
- Test Driven Utveckling
- Verktyg som stöd

Agile with Scrum

- Agile - lively, active och swift
- Scrum – vidare utveckling av Iterativ/Incremental utveckling
- Scrum består av flera sprintar
- Arbetar i 30 dagars sprintar med ett "fixed scope"
- Små team 4 till 10 personer

Scrum Overview

Überblick

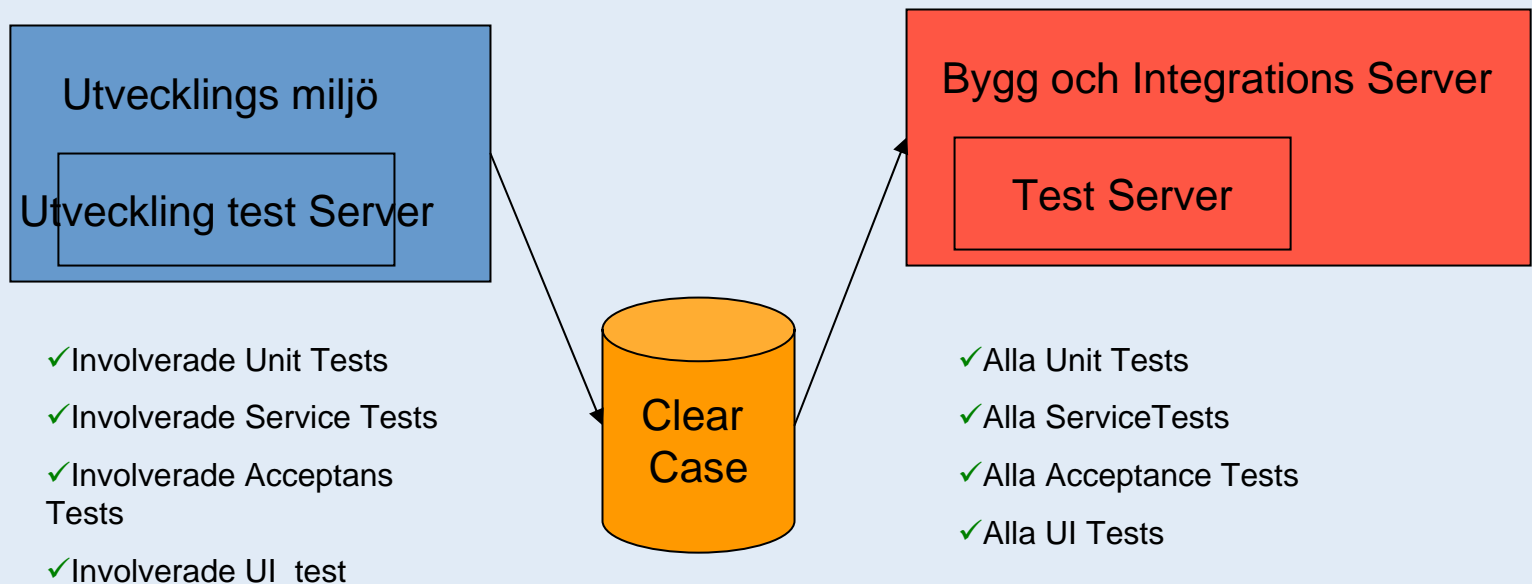


Continous integration

- Integrerar koden ofta med 'all' annan kod
- Bygger schemalagt (natt) och automatiskt vid incheckning av ny körbar och testad kod
- Automatiska regressions tester - snabb feedback
- Genom röda tester kan man se kod-förändringar i andra moduler snabbt
 - ◆ Cruise control – triggar
 - ◆ Ant script – bygger

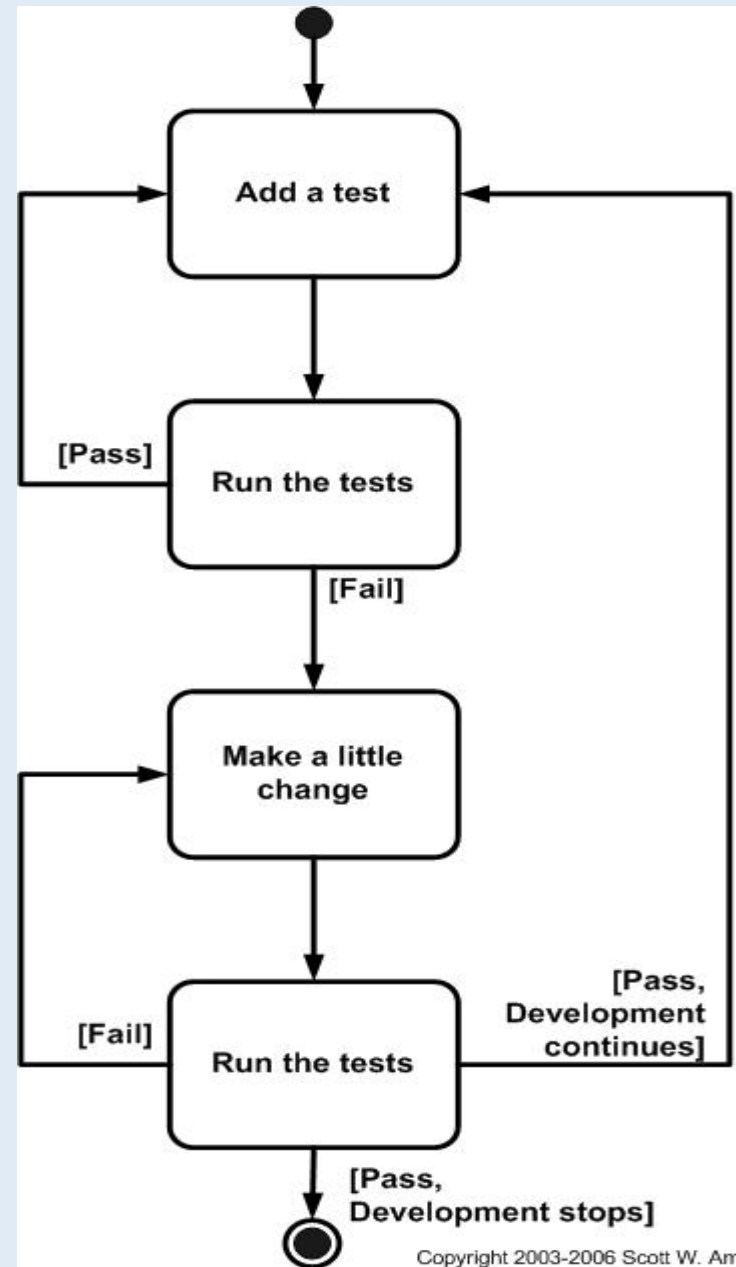
Continuous Integration

- Automatisk Kompilering och körning av test vid kodförändring

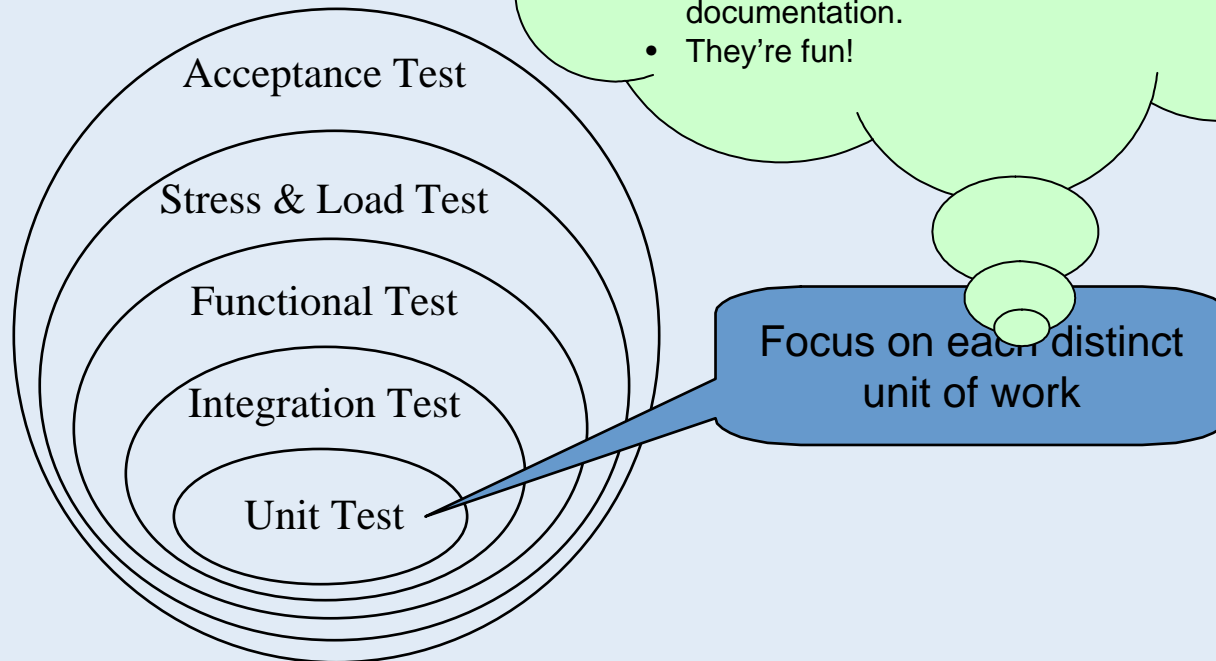


Test Driven Utveckling

- Test :
 - ◆ styr utveckling
 - ◆ Acceptans tester som specificerar en funktion inte validerar
 - ◆ som design
- Specificerar och implementerar unit tester innan man kodar
- Viktigt med 'Code reviews'



The testing model



TDD och Continuous integration

- Vad är den verkliga fördelen av TDD?
- Test i focus
- Utvecklare tänker test när de utvecklar – mer kund focus
- Produkt analyser får större teknisk inblick
 - ◆ Genom att delta i framtagandet av acceptance tester
- Ökar team känslan
- Bättre kvalitet

Fitnessse

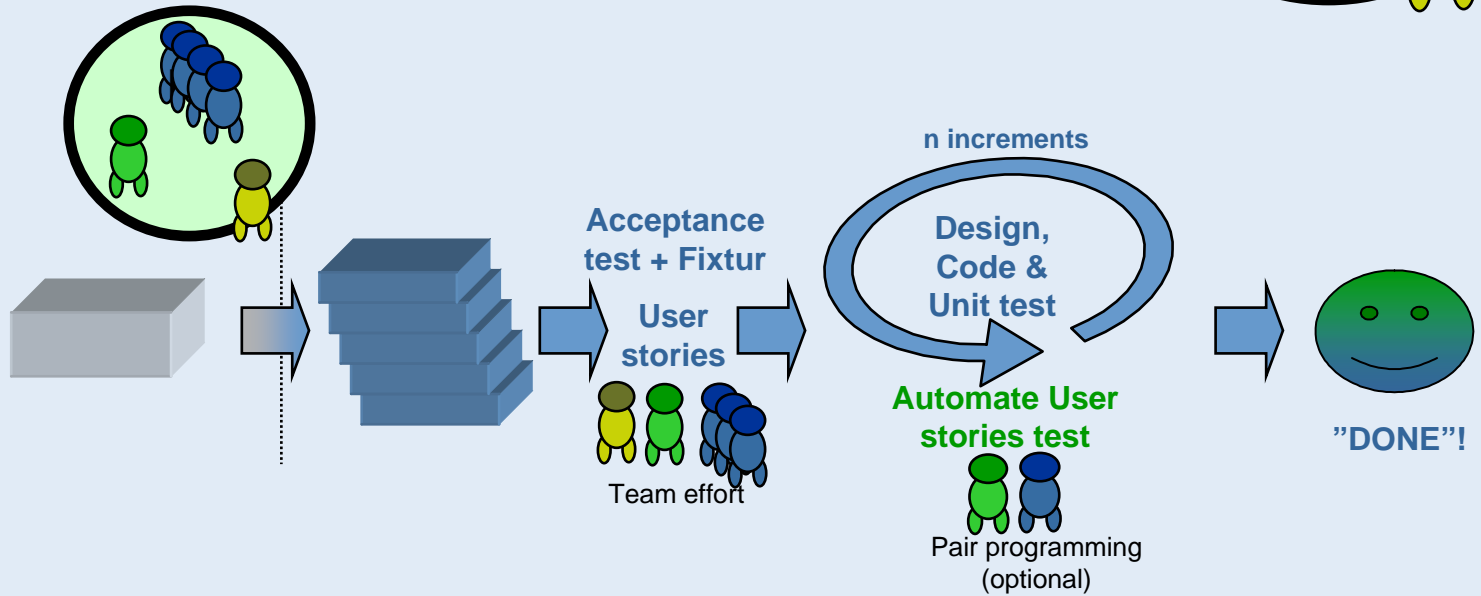
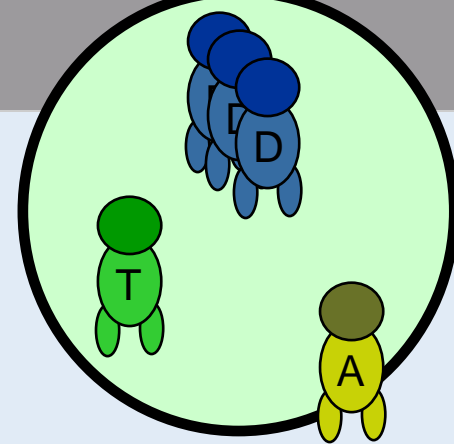
- En Wiki – för att specificera test fall och data
- En Webbserver
- Ett enkelt test verktyg som
 - ◆ Har stöd för att sätta upp test data
 - ◆ Automatiskt jämför vad mjukvaran skall utföra och vad den faktiskt utför
 - ◆ Har stöd för att ”rensa” test data efter sig
- Kopplar testfall med kod m.h.a. fixturer
- Finns för både java och C++
- Ett forum för förståelse av funktionalitet

Exempel Test Driven Utveckling Allergy team

- Typer av test
 - ◆ Unit tester – ren logic in Både SP och Service
 - ◆ Stored Procedure – som helhet
 - ◆ Service – som helhet
 - ◆ Acceptance – säkerställa förväntad funktionalitet
 - Service nivå och ner till DB lagret
 - ◆ UI – Feature Integration Test (Manuella) – säkerställa förväntad funktionalitet

DAM Allergy team

- Utvecklings Sprint -



Exempel Test Driven Utveckling

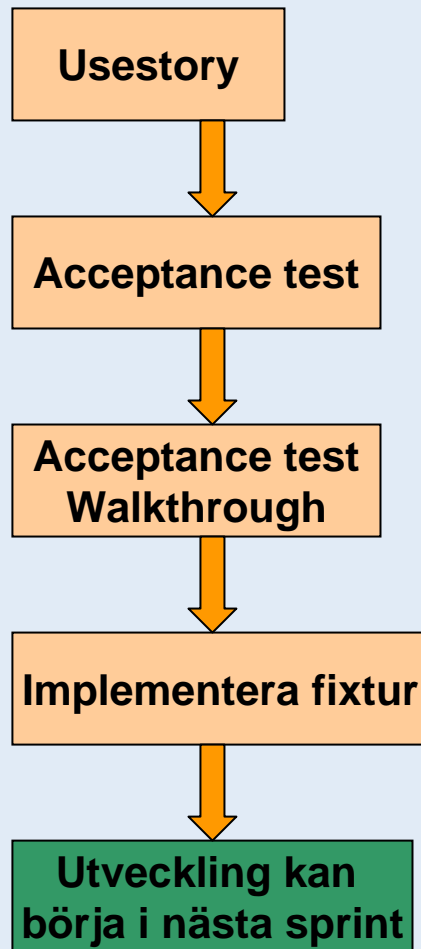


- User Stories– S1, S2
- Acceptance test - S1, S2
- Unit test och Kod –S1
- Fixtur implementeras -S1, S2
- FIT test skapas -S1, S2
- FIT test körs S1 funktioner



- User Stories– S3
- Acceptance test - S3
- Fixtur implementeras – S3
- Unit test och Kod – S2
- FIT test skapas –S3
- FIT test körs S2 funktioner

Arbetsflöde



- User story – PA
- Acceptance tester – PA eller Testare
- Walkthrough av Acceptance tester – Alla involverade parter
- Fixtur – Teknisk testare

Länkar:

- TDD:
- <http://www.agiledata.org/essays/tdd.html>
- http://en.wikipedia.org/wiki/Test_driven_development
- <http://www.testdriven.com>
- Tools:
- Fitnessse: <http://www.fitnessse.org>
- J-Unit, Cpp Test tools, SQL-Unit, DB-Unit

Q & A

