



Stephanie van Dijck

De integrale aanpak maakt complexiteit hanteerbaar

Samenvatting:

Nieuwe projecten nemen toe in complexiteit: afhankelijkheden tussen software componenten, de combinatie van nieuwe en herbruikte software, leveringen van derden: stuk voor stuk uitdagingen die verschillende aspecten van het project waaronder ook de teststrategie onder druk kunnen zetten. Als het team bovendien ook nieuw is, voor elkaar en voor het produkt, dan breidt de complexiteit zich uit op het organisatorische vlak. Tegelijkertijd geeft zo'n beginsituatie een kans om een nieuwe manier van werken te definiëren. Technische complexiteit wordt hanteerbaar wanneer architectuur, integratiestrategie en teststrategie op elkaar worden afgestemd in een 'integrale aanpak'. Door het afstemmen van deze expertises is een project beter onder controle: architectuur bepaalt de onderverdeling van de software in functionele blokken, de integratiestrategie bepaalt hoe de software wordt opgebouwd; de risk-based test strategie zorgt ervoor dat integratie rekening houdt met de gevoelige onderdelen van de software zodat deze in een vroeg stadium onder controle zijn. De testtaken worden verdeeld tussen de ontwikkelaars en het testteam, wat testtijd vermindert en de feedbackloop voor de ontwikkelaars belangrijk verkort. Door het verdelen van de verantwoordelijkheden voor het testen over verschillende teams met de nadruk op het gebruiken van de aanwezige expertise wordt voorkomen dat het testteam in een laat tijdstip van het project toch nog een bottleneck vormt. De integrale aanpak is succesvol toegepast bij het ontwikkelen van de eerste harddisk/DVD recorder binnen Philips: het resultaat was een systeem dat binnen de gestelde tijd, binnen het budget en met vereiste kwaliteit werd opgeleverd. Inmiddels wordt hij ook gebruikt in andere projecten, van klein tot groot.

Dit verhaal legt uit hoe de integrale aanpak in zijn werk gaat en hoe deze in elk project kan worden toegepast. Tevens wordt ingegaan op de verdeling van de testverantwoordelijkheid over de verschillende disciplines in een projectteam.

Biografie:

Stephanie van Dijck is Test Manager bij Philips TASS. Zij heeft meer dan 16 jaar ervaring in alle aspecten van (technische en embedded) software ontwikkeling en is momenteel voorzitter van de Nederlandse examencommissie voor ISEB/ISTQB.

PHILIPS

The Integral Approach

Stephanie van Dijk

Philips TASS B.V.
June 13th, 2006

PHILIPS

Identity

Philips TASS is a leading provider of technical software services and solutions, enabling our customers to bring better products faster to market



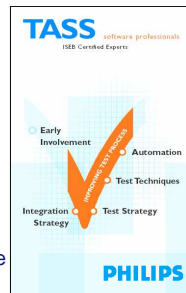
Philips TASS

2

PHILIPS

Focus

- Hw/Sw Interfacing
- Remote Services
- Integration & Testing
 - Increase I&T knowledge (ISEB)
 - Sharing I&T best practices
 - Enable customers to compose reliable software and reduce the risk of product failure



Philips TASS

3

PHILIPS

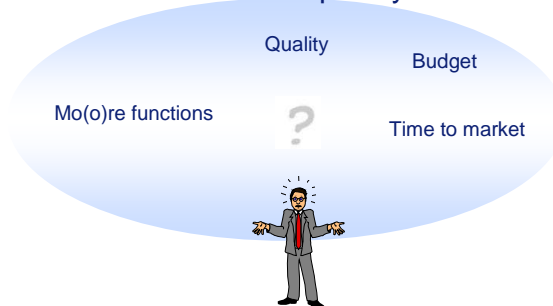
Projects become more and more complex, with consequences for controllability and quality

Philips TASS

4

PHILIPS

What Determines Complexity?

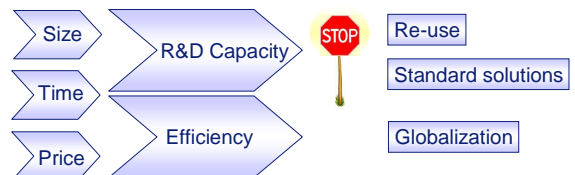


Philips TASS

5

PHILIPS

Solutions



Philips TASS

6

Consequences

- Program concept: #different products from a single functionality/subsystem pool
- Mainline concept: #similar products from a single baseline

Solution == Architecture?

- Division in subsystems
- Development of subsystems at any location
- Interfaces need careful definition



Solution == Integration? (1)

Delivery-based integration:

- Each subsystem develops autonomously
- Each subsystem has its own release schedule

Result: product team cannot prepare for integration without temporary software...

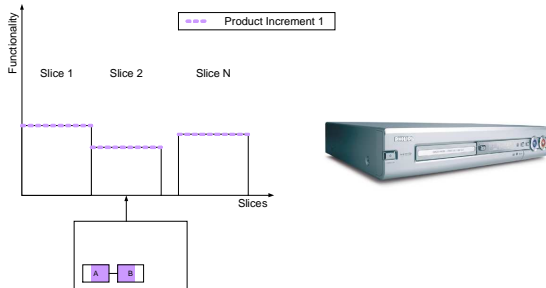
Solution = Integration? (2)

Incremental integration:

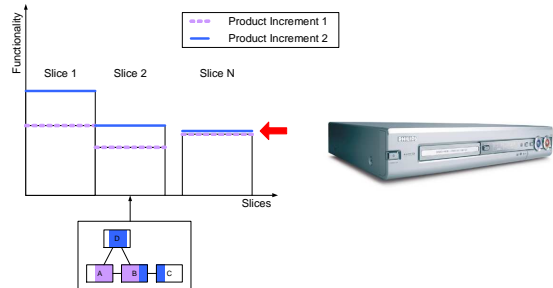
- Each subsystem develops autonomously.
- Release schedule aligned with main project.
- Different release levels.

Result: project team can prepare integration, old versions available in case of delay.

Incremental Integration (1)



Incremental Integration (2)



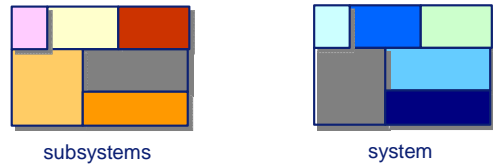
Architecture + Integration == enough? (1)

- No: what about the quality?
- Test Strategy needed to test the important stuff early, or:



Architecture + Integration == enough? (2)

If each subsystem has their own responsibility for the test approach:
Chance on gaps and unnecessary overlaps!



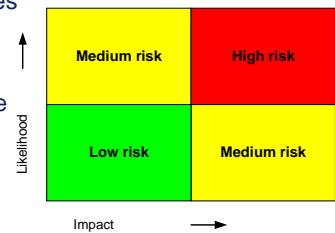
Test Strategy (1)

The test strategy must cover the product under development AND each subsystem:

- Test levels aligned
- Intake testing by product test team

Test Strategy (2)

- Risk based
 - product risk matrix
 - test techniques
 - Coverage
 - Level of independence

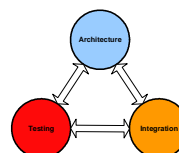


An architectural driven solution is not enough. Architecture, integration and testing must be considered together in an integral approach.

Integral Approach: Expertises

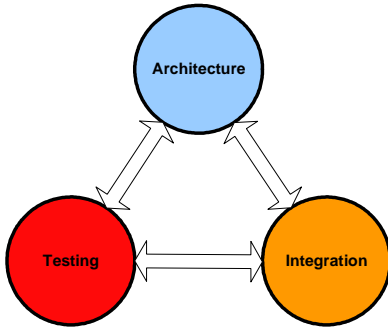
Each area is an expertise:

- Architecture == decomposition
- Integration == assembly
- Test == quality, risk control



These need to be aligned to keep the project under control

Integral Approach: Interrelationships



Decentralized Approach: Test

- Subsystem teams test their own subsystem (domain knowledge available).
- Product test by product test team.

A decentralized approach prevents the product test team becoming a bottleneck in the project.

Decentralized Approach: Test Levels

- Subsystem teams
 - Component tests
 - Integration tests (increments)
- Product team
 - System tests
 - Intake test
 - Daily confidence test
 - Automated regression & smoke test



Integral Approach: Summary

- Combine Architecture, Integration and test expertise to keep control of the project.
- Decentralized approach prevents bottlenecks.
- Integral and decentralized approach combined: spend effort as effectively as possible.

Integral Approach: Summary

Results	Success factors	Foundation
<ul style="list-style-type: none"> • Control • Quality insight • Flexibility • Efficiency 	<ul style="list-style-type: none"> • Management commitment • Skilled people • Test automation • Decentralized approach 	<ul style="list-style-type: none"> • Architecture • Integration strategy • Test strategy • ... <u>aligned</u> !!

Questions?



