Evaluation of an agile application development approach with 4GL tools in an offshored IT-Supply scenario

Dr. Michael Linke

Title & Abstract

- **<u>Title:</u>** Evaluation of an agile application development approach with 4GL tools in an offshored IT-Supply scenario
- **Related research topics**
  - ☐ IT > Program Management
  - ☐ IT > Software Engineering
  - ☐ Social Sciences > Collaboration Mechanisms
- **Abstract**

The cost pressure for IT-organisations has grown considerably to deliver projects, software, capabilities and features faster and in a more cost-efficient way. Agile software development methods, like Scrum, in cooperation with tools and methods of (partly) automated code generation can be interpreted as an answer to these prevailing challenges. Within this evaluation study in a mixed SAP IT environment a conception for the combined usage of 4GL software tools and agile software development methods together with offshored software developers within different business domains was developed and therefore executed. The results showed that an average cost reduction between 40% and 80% regarding to the overall project setup were within the range of practical realization, if a decent project and communication governance would be in place.

Agenda

- Motivation and Introduction
- Agile software methods
- Communication and Collaboration as enablers
- CASE, 4GL and Code Generators
- Conclusion
- Literature
- Q&A

- The *cost pressure* for IT-organisations has grown considerably to deliver projects, software, capabilities and features faster and in a more cost-efficient way, and this not only since the financial crisis of the last years.

- Also the increasing spread of mobile applications for mobiles or smart phones – the number of them actually already exceeds the number of stationary personal computers [Westney1995] – as well as the associated importance of world-wide available "Apps", contributes to the increased expectation users on IT organisations.

- Together with an increasing *consumerization* of hardware [Hackenson2008], that is the use of business software on private end devices and versa, as well as a likewise increasing mixture work and life habits, has led

- to a *21st century customer* [Takeuchi1986] who demands and asks for reactions to inquiries an requests, to some extent also real-time.

→ Demand for „fast" and „cheap" IT

Motivation and Introduction > Dimensions

- Demand for „fast" and „cheap" IT
- Dimensions
  - Fast and efficient overall process (PM)
  - Fast and efficient software engineering itself (SE)

Source.: [Windev, Scriptcase, Microsoft]

Motivation and Introduction > Current market response PM

Source.: [Collabnet, Telerik, PMI]

Agenda

- Motivation and Introduction
- Agile software methods
- Communication and Collaboration as enablers
- CASE, 4GL and Code Generators
- Conclusion
- Literature
- Q&A

Agile software methods > Overview of the process

Source: [Northeaster University, College of Engineering]

R = Requirement S = Software
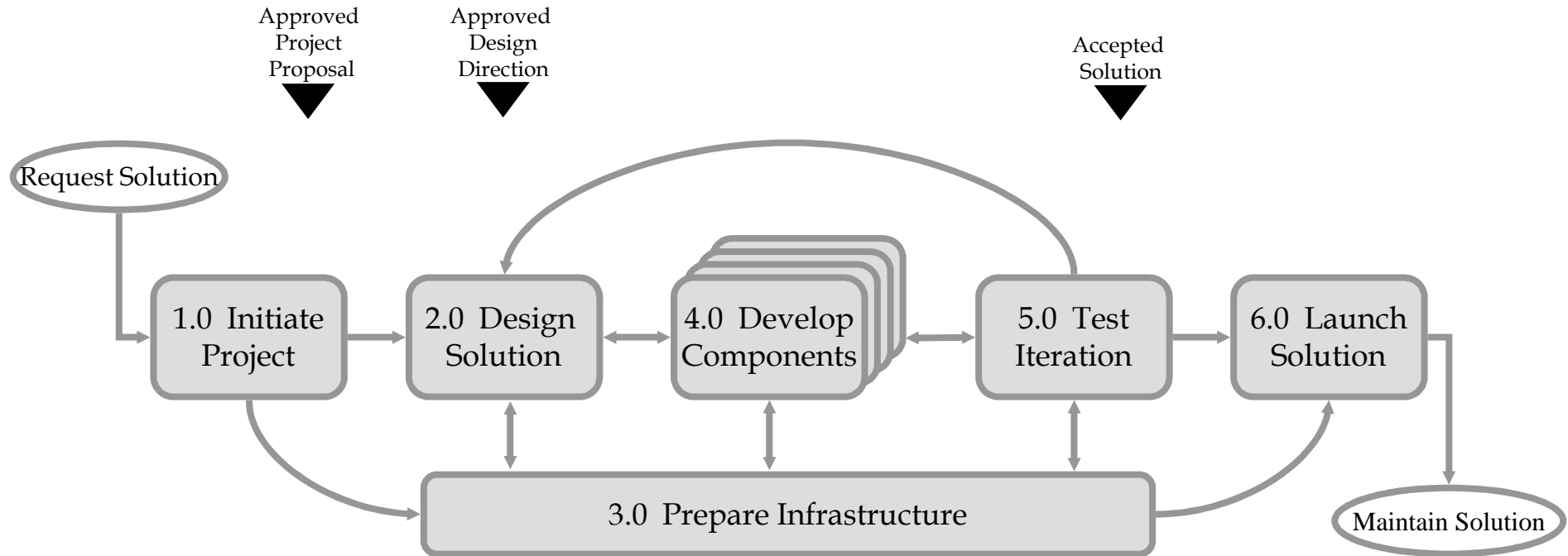
Agenda

- Motivation and Introduction
- Agile software methods
- Communication and Collaboration as enablers
- CASE, 4GL and Code Generators
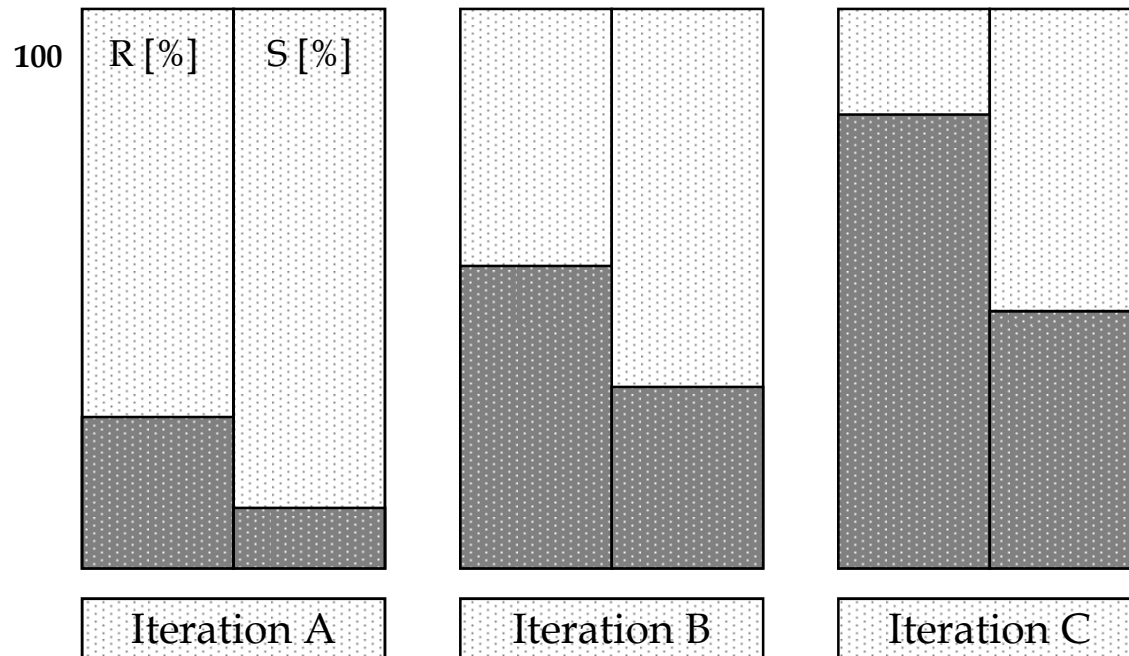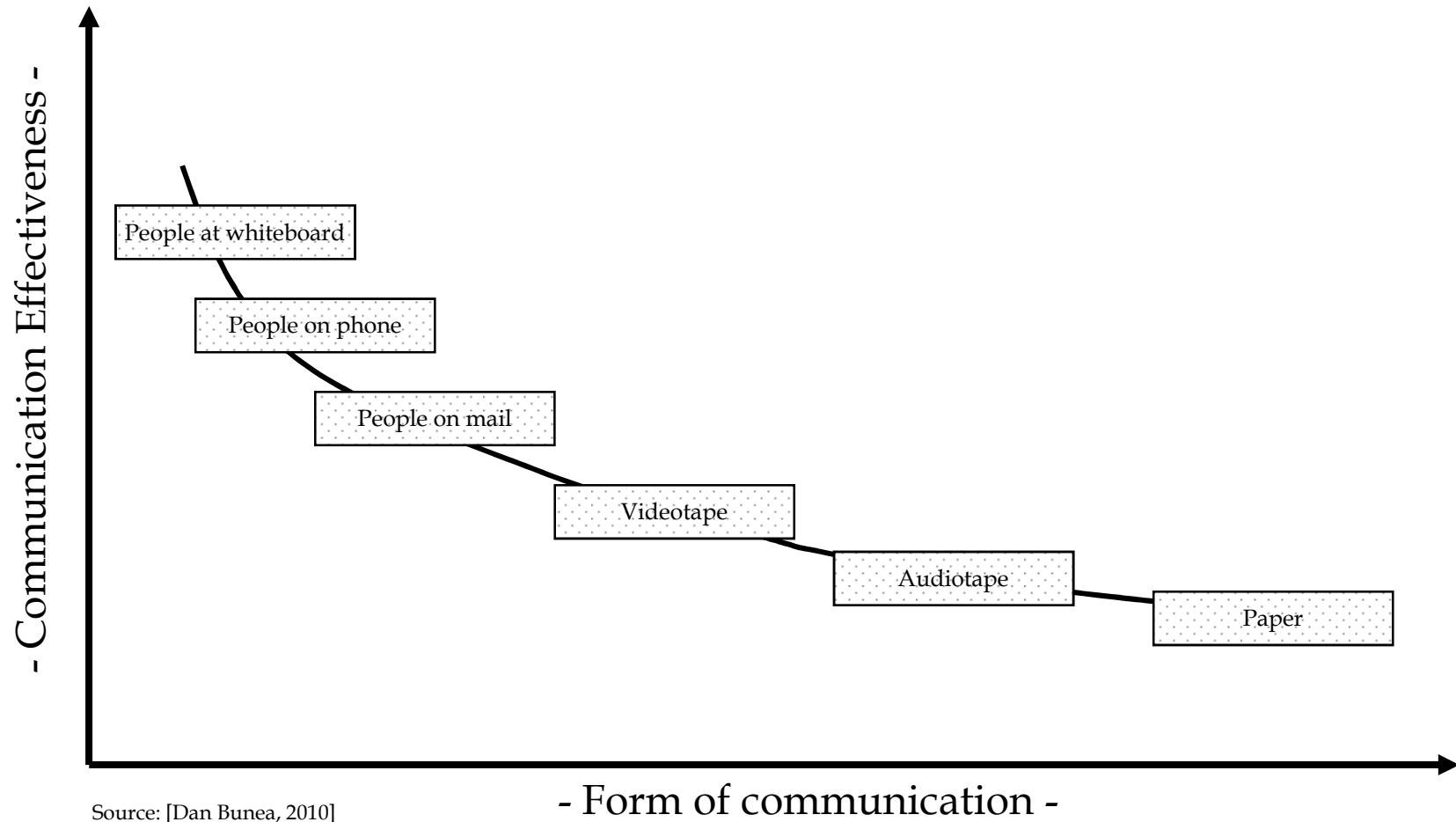- Conclusion
- Literature
- Q&A

- Communication Effectiveness -

People at whiteboard

People on phone

People on mail

Videotape

Audiotape

Paper

Source: [Dan Bunea, 2010]

- Form of communication -

Communication and Collaboration as enablers > Technical setup

@ Forum & Mailstore

AUTO-Notifier

@ Bugtracker

AUTO-Notifier

Content Repository

**New role of Technical Business Analyst in a „Hub & Spoke"
Architecture**

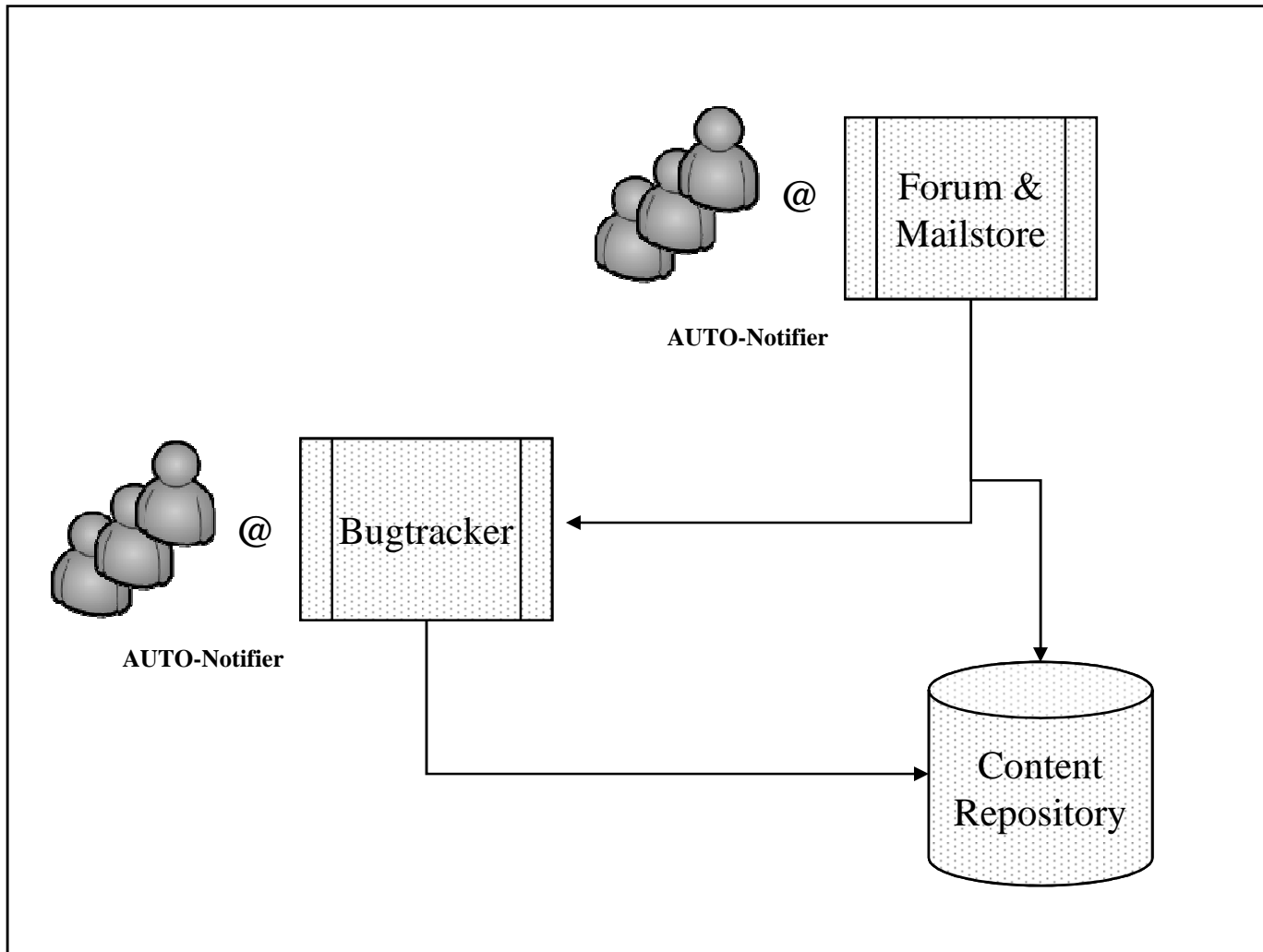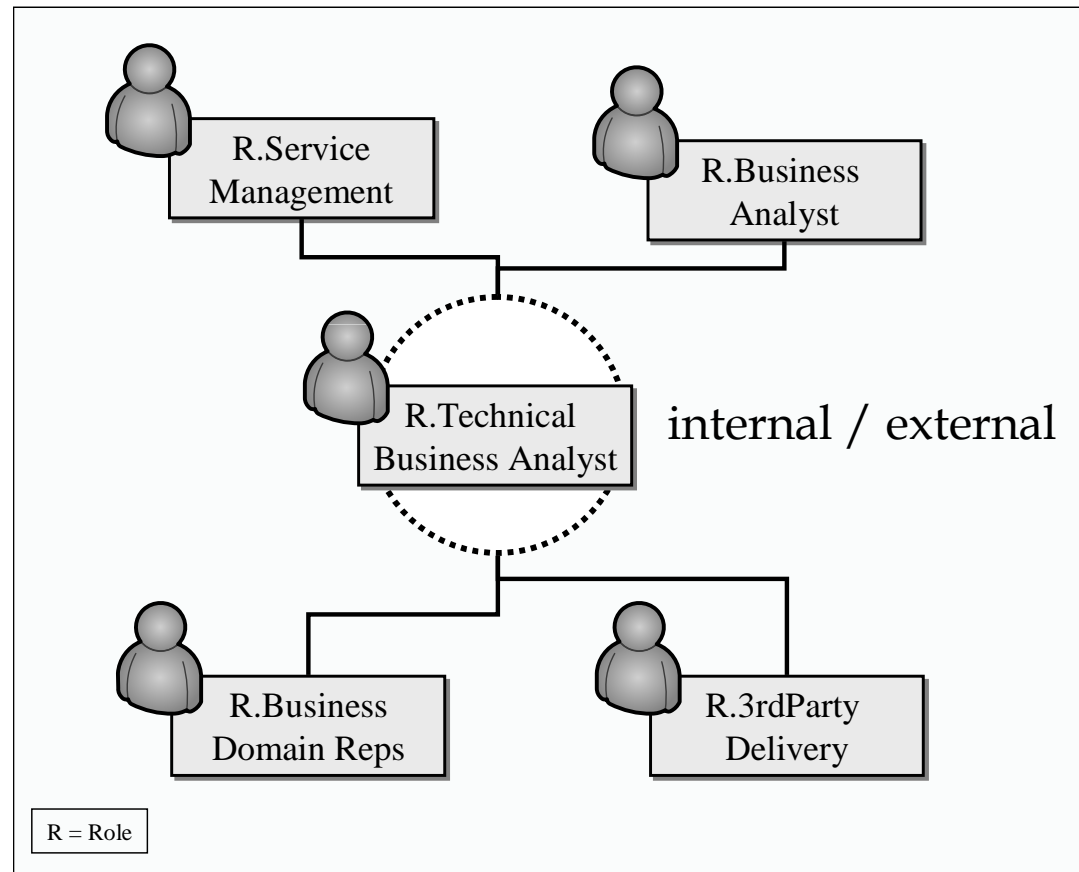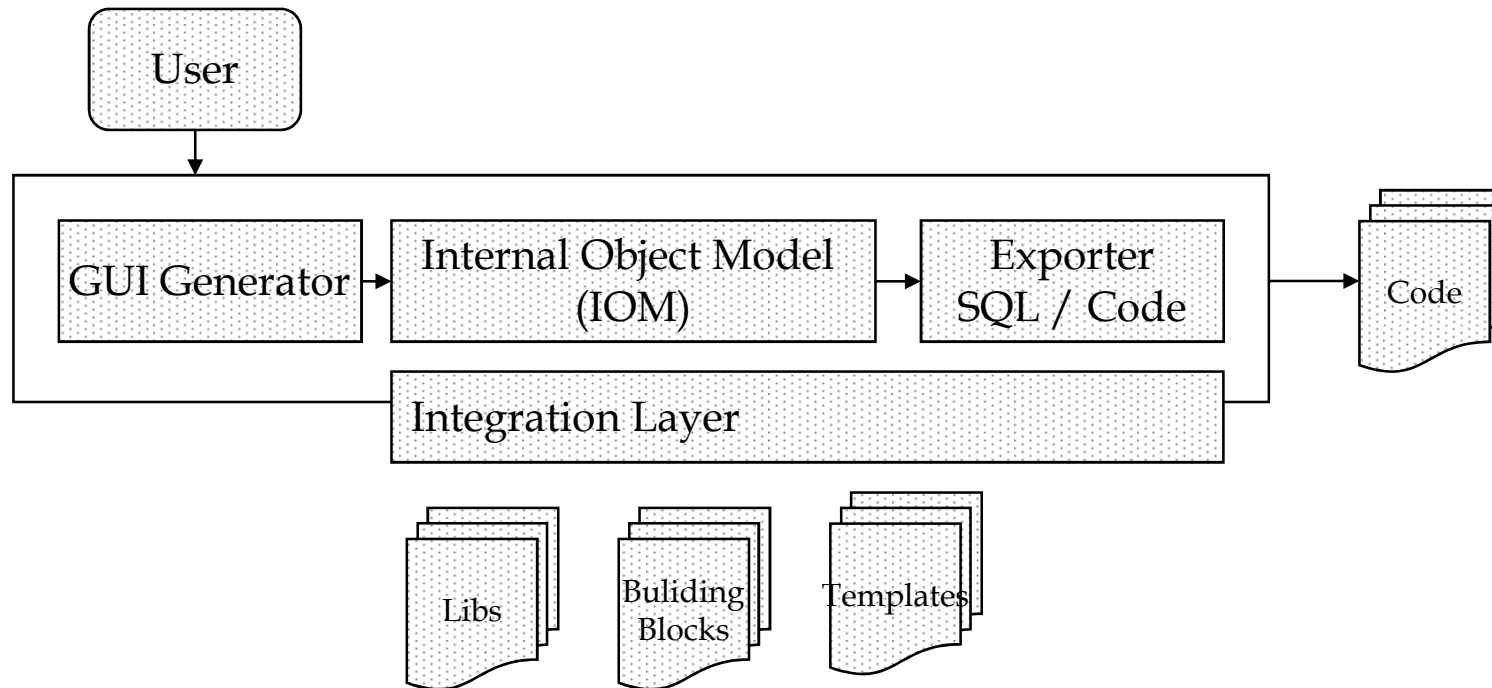Fig.: Hub & Spoke roles of the Tcehnical Business Analyst (Source: Own composition)

Agenda

- Motivation and Introduction
- Agile software methods
- Communication and Collaboration as enablers
- CASE, 4GL and Code Generators
- Conclusion
- Literature
- Q&A

User

GUI Generator → Internal Object Model (IOM) → Exporter SQL / Code → Code

Integration Layer

Libs | Buliding Blocks | Templates

Source: [Own composition]

Source: [Microsoft 2011]

CASE, 4GL and Code Generators > Examined Tool "Scriptcase"
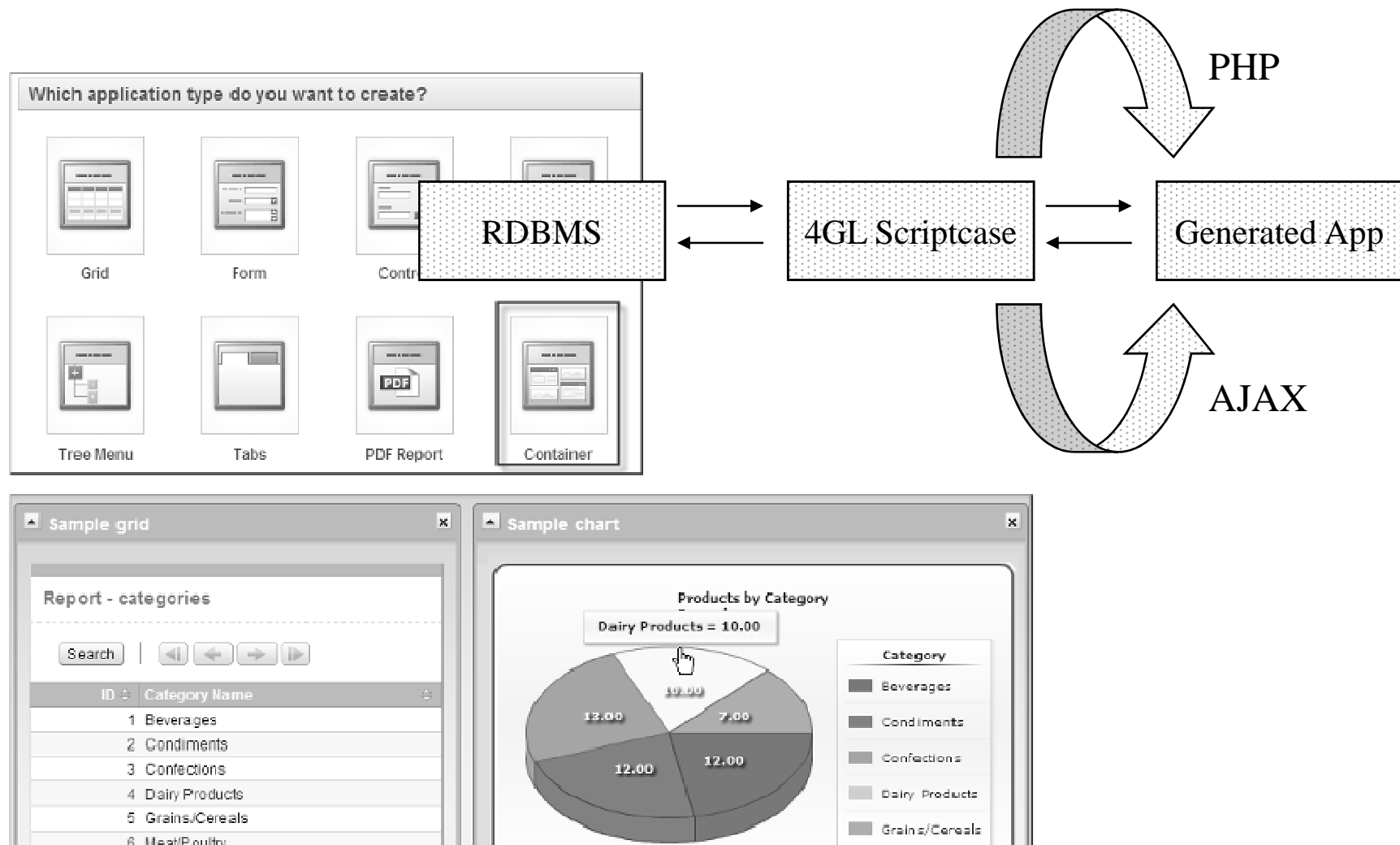
Source: [Scriptcase, 2011]

Period	Build	Technology
1980's	10-20 md	3GL, T.P. Forms, Cobol, Codasyl Database. Everything hard work !
1990's	2-5 md	4GL, Case*Generators, Forms3, Oracle7/8.
>2000	3-6 md	3GL, Hand crafted Pl/Sql HTML, Oracle8.

Source: [Foxopen.net]

Today

FeldThoughts

« Yahoo Finally Got Back Into My Daily Folder | Early Stage Investing in an Uncertain Economy »

What Happened To The 4GL?

View Comments | Tweet | Share | Like

On today, this day of endless blogging / talking (and hopefully downloading) of Google Chrome, I thought I'd go retro and write about my renewed hunt for the 4GL. But first, a little context on how this came about.

This summer I spent some time playing around with Google AppEngine to try and understand it better. It didn't take long before I realized I needed to really understand how to program in Python to do anything. So I spent some time reading about Python and ultimately realized that if I wanted to make any progress, I needed to spend a chunk of time actually learning Python. Fortunately, I found a nice MIT course titled 6.00. I took the equivalent (6.001) in 19 [...] goody – I can re-live my 18-year old stu [...]

I know that there are some companies working on this. In an attempt to be trendy, this is now called PaaS (Platform-as-a-Service). But it's really the reincarnation of the 4GL. And I'm not sure I really want a PaaS – I think I just want a 4GL that works with today's web-based environment.

Source: [Brad Feld]

Agenda

- Title and Abstract
- Status Quo IT divisional paradigms
- Status Quo Open Source (OS) usage
- Conception of an OS centric IT division
- Conclusion
- Literature
- Q&A

Conclusion

- With iterative software development methods it seems to be possible to develop software *faster* than with a development method driven by a traditional V-model.

- By using this general development approach with partly automated code generation, for instance through 4GL-code generators, additional *comparative speed and cost advantages* can be achieved.

- The use of *offshore developers* can be asserted as an additional component for IT-project cost reduction in IT-organisations, because usually they have significantly lower hourly rates compared to onsite developers.

- But at the same time the effort for communication between onsite and offsite resources *increased significantly*, here it was tried to support this at the best through the use of appropriate online collaboration tools.

- Despite the offshore components, iterative GUI reviews were possible through *telepresence* and web-conferencing though, as well as the processing of change requests.

- However, the historically observed shortcomings of the *code quality* of the 4GL-tools generation became as significant as potential topics concerning the guaranteed future of the operating company as well as the additional versioning of the development environment itself.

… start small and smart …

Agenda

- Title and Abstract
- Status Quo IT divisional paradigms
- Status Quo Open Source (OS) usage
- Conception of an OS centric IT division
- Conclusion
- Literature
- Q&A

IJAMSGD | New International Journal

NEW

- We are interested in papers relevant to contemporary business and management challenges.
- Current Call for Editorial Board
- Current Call for Papers