

IBM Software

# UK Innovate 2010

The Rational Software Conference



Smarter software for a smarter planet.



IBM Software

# UK Innovate 2010

The Rational Software Conference

## Experiences with Agile Requirements @ Scale

Dee Wauchope / Pete Franklin

[dee.wauchope@capgemini.com](mailto:dee.wauchope@capgemini.com)

[peter.franklin@capgemini.com](mailto:peter.franklin@capgemini.com)



Smarter software for a smarter planet.



## Who we are

Dee Wauchope

Delivery Process Team Lead

Pete Franklin

Head of Process & Enablers

*Leading the development and implementation of Agile RUP as Process Engineers, Mentors & Training Facilitators*



# What we're going to talk about

The 'traditional' view

Challenges @ scale

Agile RUP

Agile requirements @ scale

Conclusion



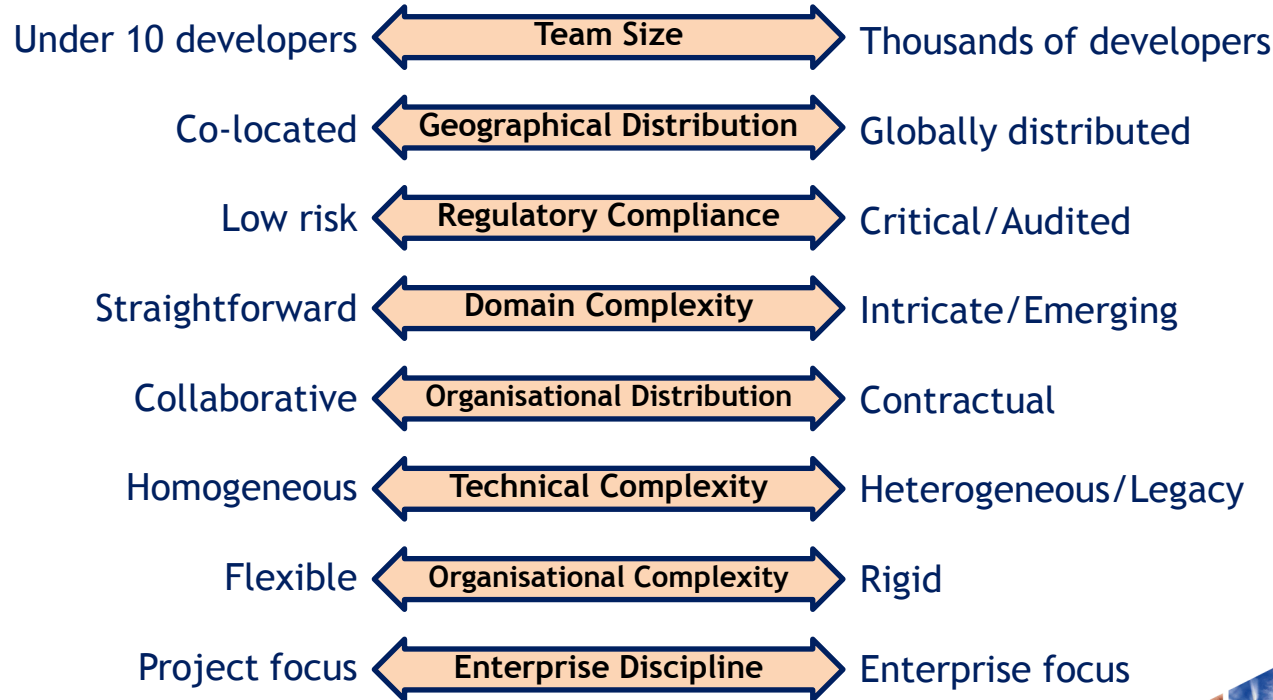
# The 'traditional' view



# The Capgemini experience



# Challenges @ scale



Agile Scaling Model (IBM/Ambler)



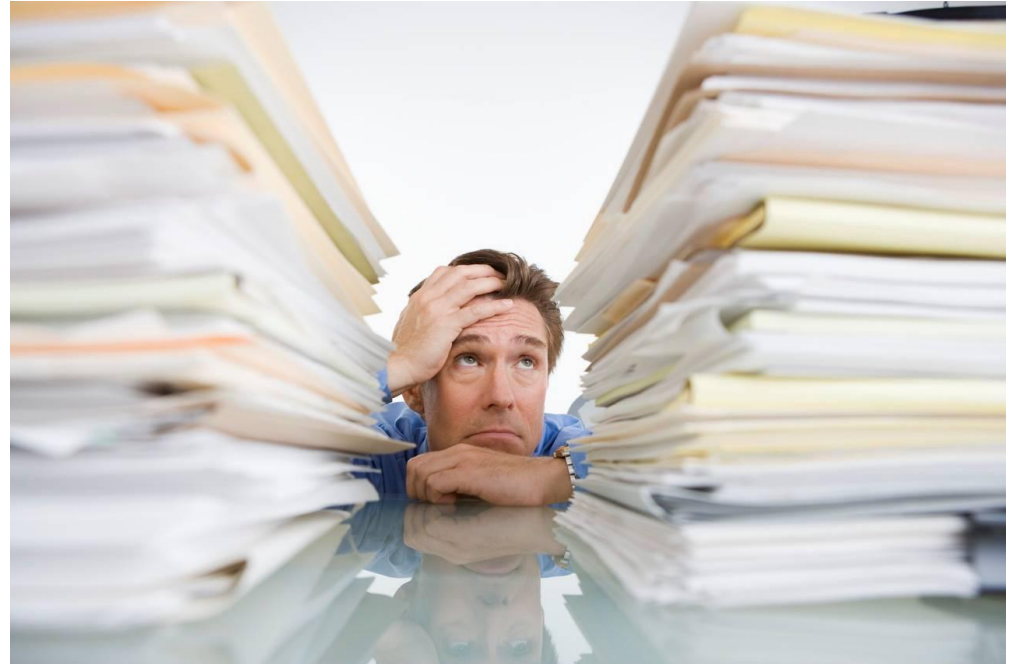


# Large Requirements Set

Thousands of stories?

Prioritisation

Mental limits





Large teams

How to subdivide?

How to collaborate?

How to manage knowledge?



## Location, location

- Geographic separation
- Culture and Context
- Time zones



# Agile RUP



+



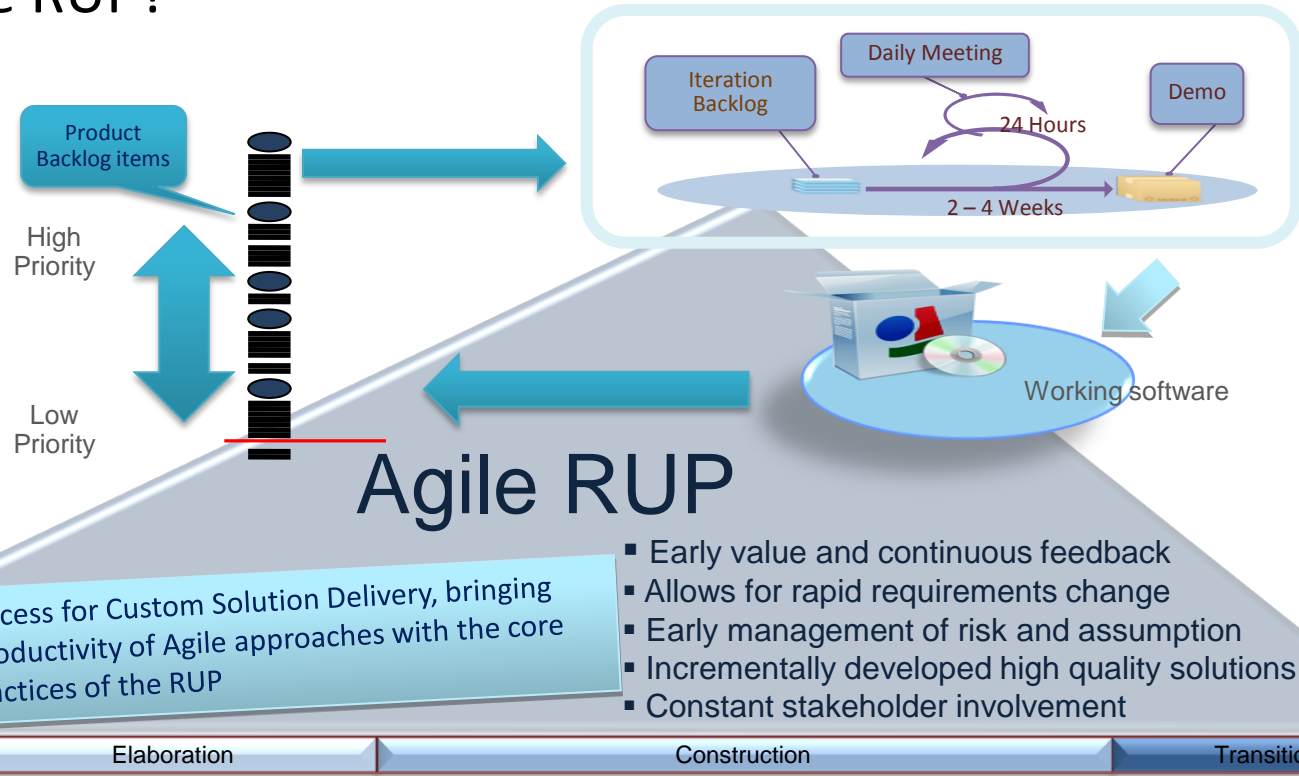
+



=



# What is Agile RUP?



Our standard and proven process for Custom Solution Delivery, bringing together the flexibility and productivity of Agile approaches with the core practices of the RUP

- Early value and continuous feedback
- Allows for rapid requirements change
- Early management of risk and assumption
- Incrementally developed high quality solutions
- Constant stakeholder involvement



## Analyst mindset



*{Noun}: a set of beliefs or a way of thinking that determines one's behaviour, outlook and mental attitude*





# Generalising specialists



# Embracing change

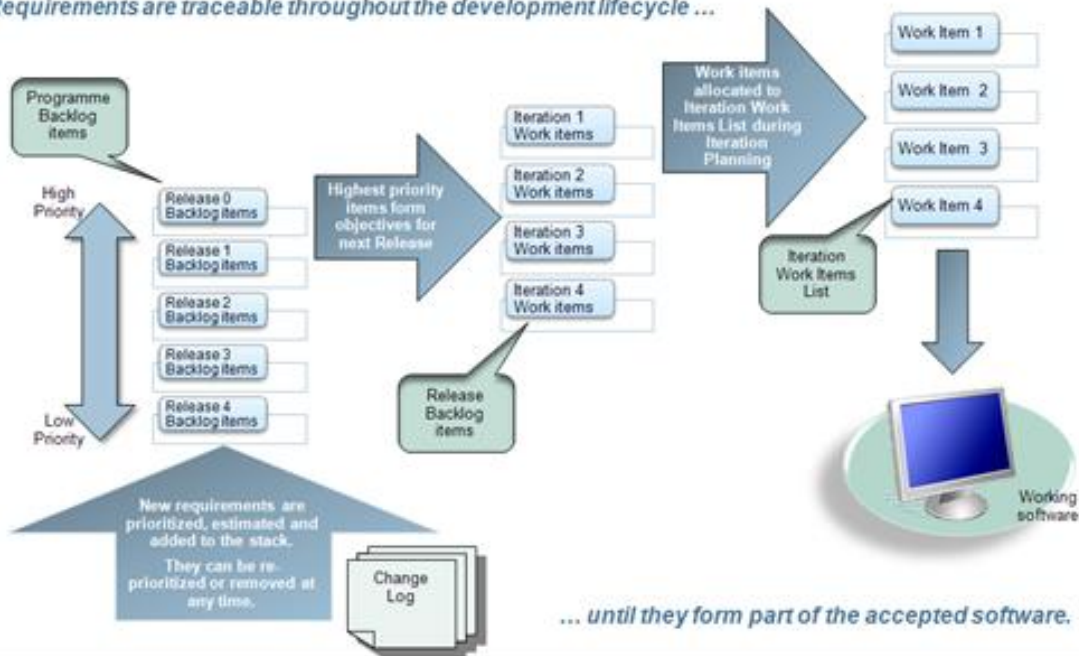




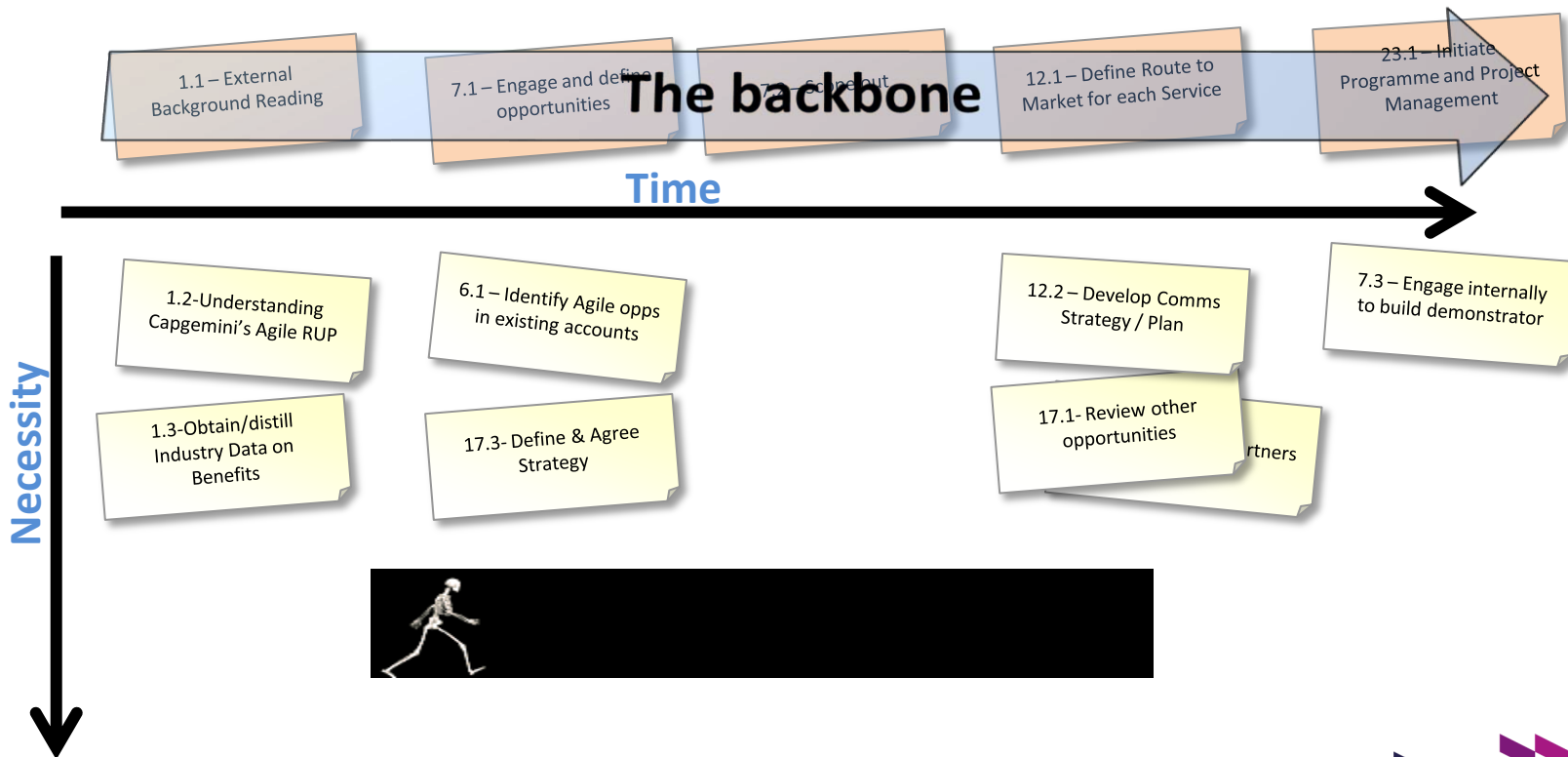
# Requirements @ Scale

The Backlog items are mapped to the client requirements and used to perform iteration planning

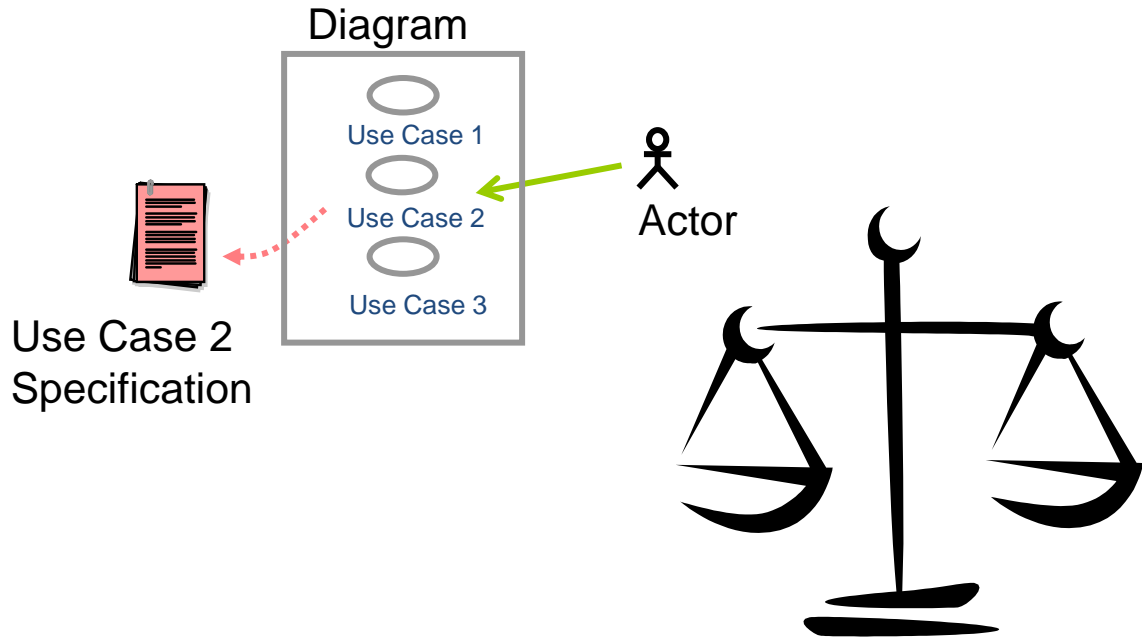
*Requirements are traceable throughout the development lifecycle ...*



# The walking skeleton



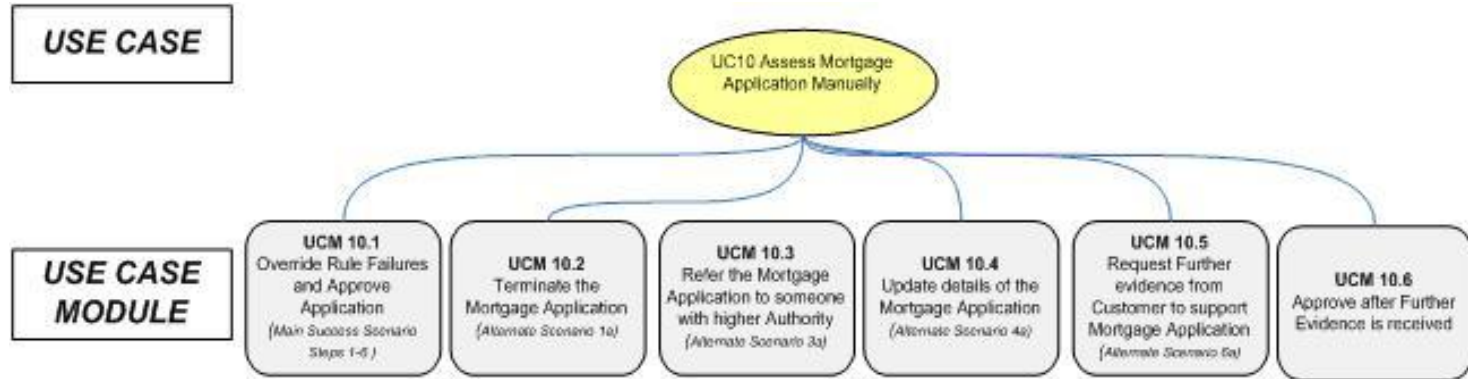
# Use cases vs. user stories



“As a *<type of user>*, I want *<some goal>* [so that *<some reason>*].”  
[Optional]



# Use case modules



# Identifying use case modules

Identify key scenarios – those associated with the highest risks.

Group flows with similar themes.

Identify Test Conditions and allocate to Use Case Modules.

Ensure that the Use Case Modules are sized to fit into an Iteration. Use Case Modules should be flexible enough to be split up if too big or combined to create sensibly sized work items. Thus, a simple Use Case may be split into the following Use Case Modules:

- Main success scenario and functional Test Conditions

- Main flow with specified alternate scenarios and functional Test Conditions

- Performance and stress Test Conditions for Main flow



# Level of documentation

Just enough, just in time

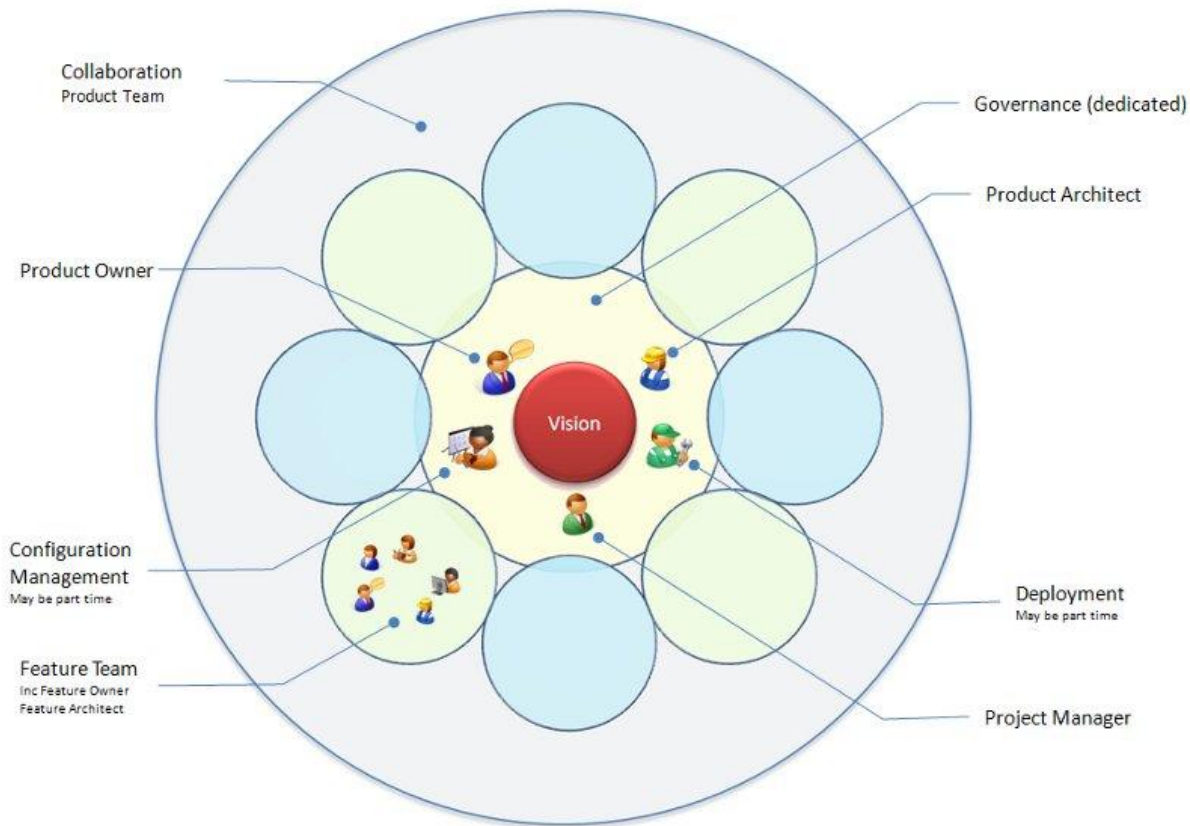


# Scaling teams

Organise by features

Governance

Support



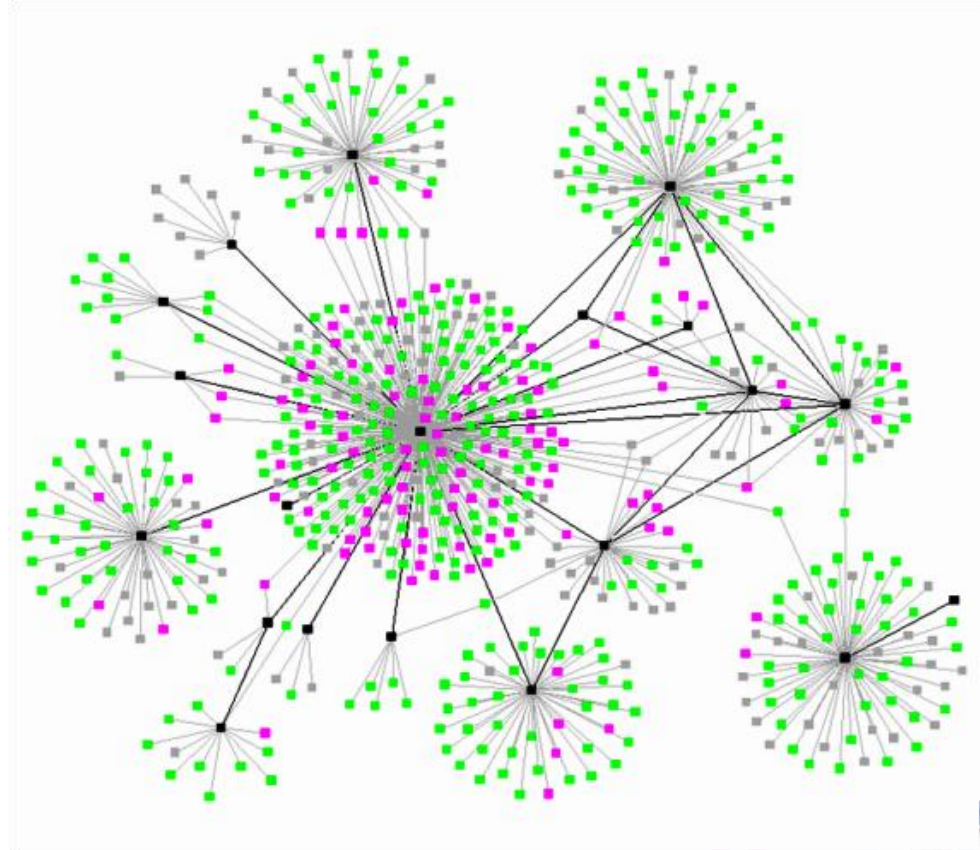


## Distribution

Communicate – by all possible means

Get a travel budget

Don't forget Context

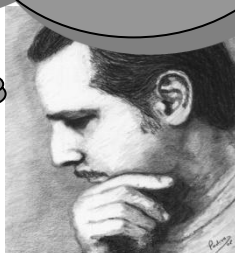


## So how was it for our analysts?

I'm not sure what this means for my deliverables. Do I still need to produce definitive Use cases and get them signed off?

Do I need to write-up detailed features/modules as well? If so, how will I get this done while also supporting the feature teams?

And I want to be an Analyst not a Tester.



Before

I'll need a reasonable view of requirements before entering the iteration – but I like being able to refine some of the details once we get started. The users love it – and this really helps me in getting quick decisions and clarifications. Doing some testing works better than I feared. Everybody helped out to achieve our goals, and there was a tremendous team spirit.



After

# Conclusion



# QUESTIONS





© Copyright IBM Corporation 2010. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.



IBM Software

# UK Innovate 2010

The Rational Software Conference



Smarter software for a smarter planet.

