

Scaling Up & Scaling Down



Reqtrix
improving software development

Iterative Project Management: A Scalable Approach to Managing Software Development Projects

Iterative software development methodologies offer many benefits for modern software development projects but are often pigeon holed as only being suitable for certain kinds of small to medium sized projects that are prepared to adopt a very technologically focused, informal, hands-off management approach.

Based on many years of experience working with project managers on iterative software development projects, this presentation will present a scalable iterative project management approach, based upon the Unified Process project lifecycle, that allows iterative software development practices to be successfully applied to all sizes of software development projects.

The talk focuses on how to apply iterative project management practices to different sizes of project, how to scale the project management practices to meet the needs of the organisation (including their adoption within a PRINCE2 environment) and how to get started in their application. Ian spoke to us last year on Iterative Development and in response to members requests is expanding his subject this time.

Scaling Up & Scaling Down

Agenda

- w** Background
 - §** A Brief Introduction To Iterative Development
- w** A Typical RUP Project
- w** Scaling Up - Managing Large Projects
- w** Scaling Down - Managing Small Projects
- w** Conclusions

Scaling Up & Scaling Down

The Principles of Scalable Iterative Development

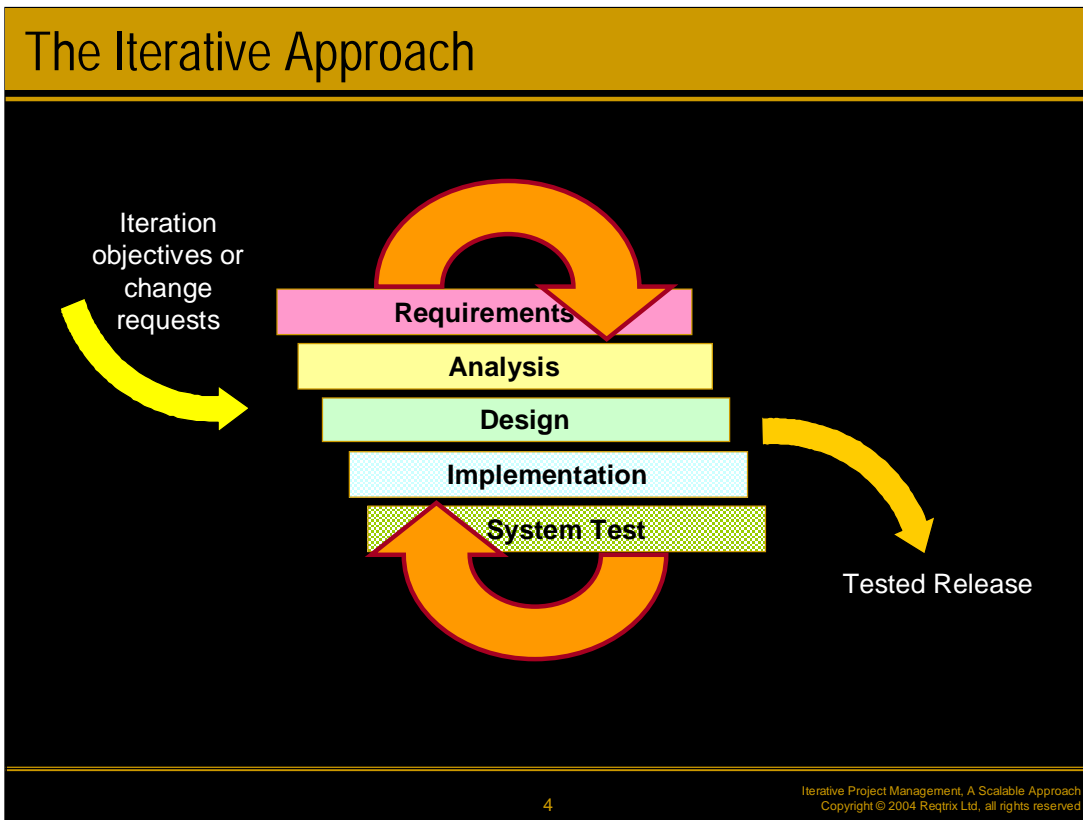
1. Iterative and Incremental
2. Use-case / scenario driven
3. Risk focused
4. Architecture-centric
5. Agile

3

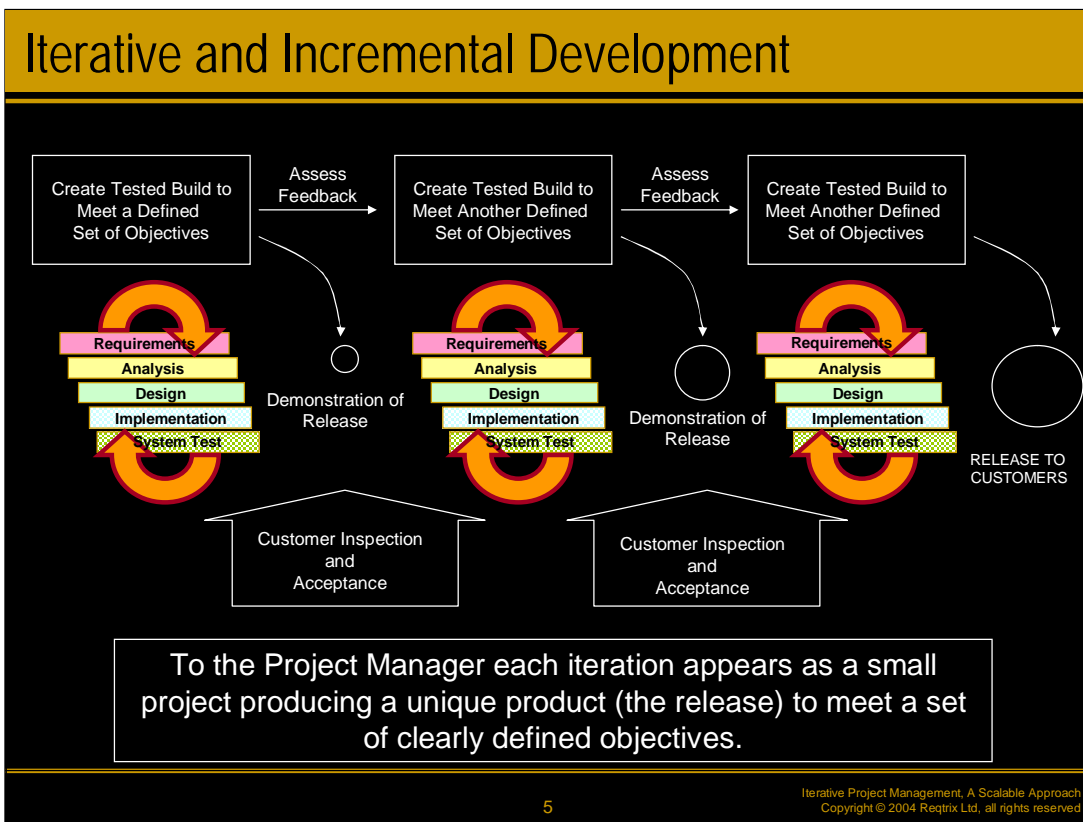
Iterative Project Management. A Scalable Approach
Copyright © 2004 Reqtrix Ltd. all rights reserved

The principles are not in any particular order....

Scaling Up & Scaling Down



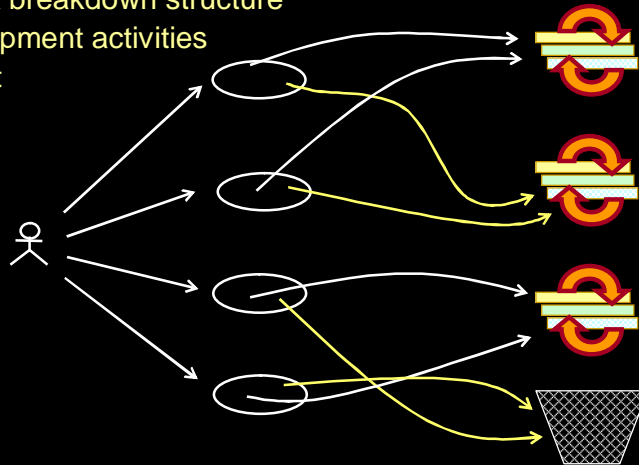
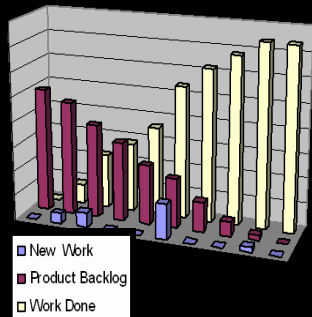
Scaling Up & Scaling Down



Scaling Up & Scaling Down

Use Case and Scenario Based Planning

- Use cases place requirements into context providing the end to end threads required to successfully set the objectives for the iterations
- Use cases are ideal for:
 - structuring the work breakdown structure
 - planning the development activities
 - scope management
 - progress checking



6

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

There are other story and thread based mechanisms that can be used to facilitate the management of iterative software development.

Extreme programming, for example, uses User Stories in the same way that the Unified Process uses use cases.

Scaling Up & Scaling Down

Iterating: The Project Manager's Perspective

Overall Project Management Process

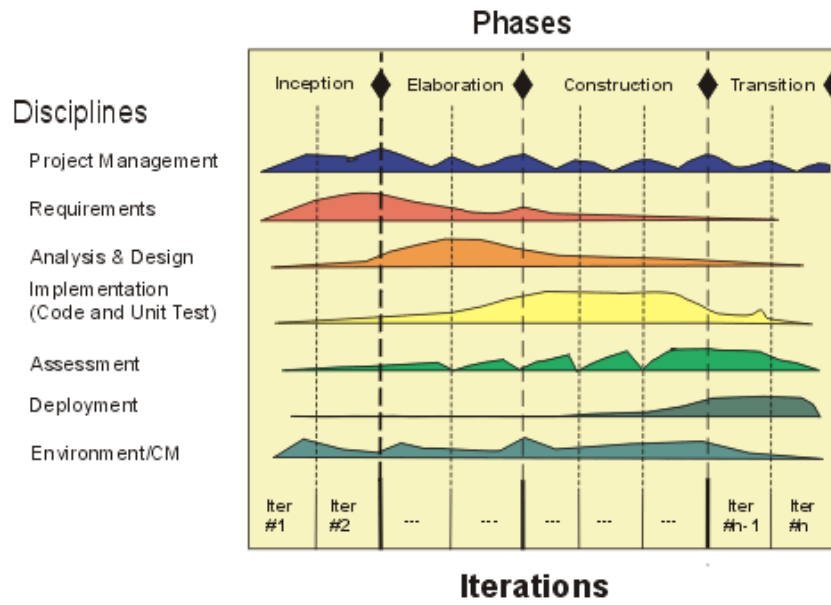


Iteration Management Process

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Scaling Up & Scaling Down

The Unified Process: Anchoring The Lifecycle



Scaling Up & Scaling Down

Phases and their Purpose

Phase	Focus
Inception	Confirm the scope and objectives of the project and <i>bring the business risks under control</i>
Elaboration	Stabilise the product plans and <i>bring the architectural and technical risks under control</i>
Construction	Build the product and <i>bring the logistical, project execution risks under control</i>
Transition	Deliver the product and <i>bring the roll-out risks under control</i>

The phases define a risk-driven lifecycle

9

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

The Phases: An objective measure of the 'state' of the project.

Scaling Up & Scaling Down

Agenda

- w Background
- w A Typical RUP Project
- w Scaling Up - Managing Large Projects
- w Scaling Down - Managing Small Projects
- w Conclusions

Scaling Up & Scaling Down

Project Scale Has Many Dimensions

We will be looking at:

- w Number of software products to be produced
- w Duration
- w Number of project teams
- w Number of people
- w Iteration Length
- w Formality
- w Team Distribution

11

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Size varies by:

- Team size
- Objective
- Amount of overhead
- Team distribution and availability
- Project formality
- External dependencies
- Project environment
- Criticality of errors

Scaling Up & Scaling Down

A Typical Iterative Project

- w Number of software products: 1
- w Single Project Team
- w Duration: 6 – 9 months
- w Number of people: 5 – 30
- w Iteration Length: Typically 2-6 weeks
 - § Most common 4 weeks (30 calendar days)
- w Formality: Medium
- w Team Distribution: Co-located

Scaling Up & Scaling Down

Phases, Effort and Schedule

MBASE & RUP Spiral-based Activity/Phase Effort and Schedule distributions covered by COCOMO II					
Activity / Phase	In or Out of Scope	Percentage of Value Estimated Directly by COCOMO II			
		MBASE		RUP	
		Effort	Schedule	Effort	Schedule
Inception	Out	6% (range 2%-15%)	12.5% (range 2%-30%)	5%	10%
Elaboration	In	24% (range 20%-28%)	37.5% (range 33%-42%)	20%	30%
Construction	In	76% (range 72%-80%)	62.5% (range 58% - 67%)	65%	50%
Transition	Out	12% (range 0%-20%)	12.5% (range 0%-20%)	10%	10%
Totals		118% (range 102%-135%)	125% (range 102%-150%)	100%	100%

Source: COCOMO II Frequently Asked Questions

13

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Scaling Up & Scaling Down

How Many Iterations?

	Total	Inception	Elaboration	Construction	Transition
Small	3	1		1	1
Typical	6	1	2	2	1
Large	10	2	2	4	2

A project may have many more iterations (20 to 40 have been reported) but would go through many cycles.

Scaling Up & Scaling Down

A Strange Coincidence: The Square Root Rule

Project Duration		Iterations	
		Length	Number
Two years	100 weeks	10 weeks	10
One year	49 weeks	7 weeks	7
Nine months	36 weeks	6 weeks	6
Six months	25 weeks	5 weeks	5
Four months	16 weeks	4 weeks	4
Two months	9 weeks	3 weeks	3
One month	4 weeks	2 weeks	2

Source: What if we used common sense?, J Marasco, Rational Edge, Jan 02

Scaling Up & Scaling Down

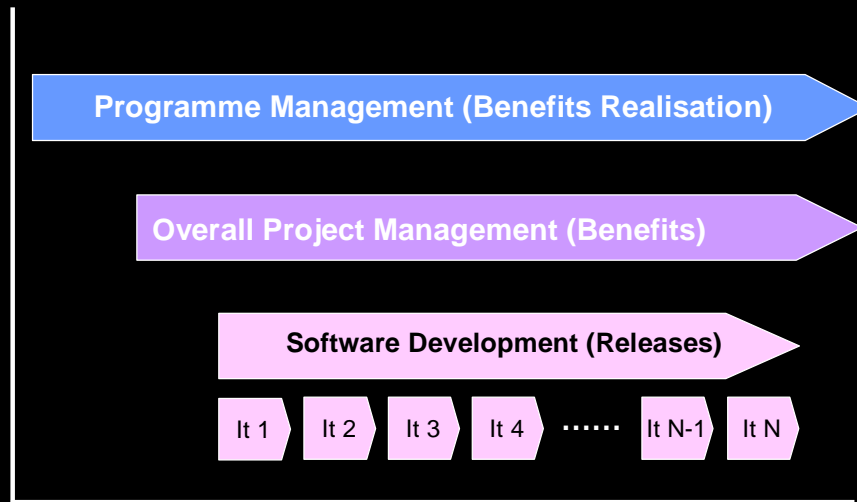
Ignore the Corner Case

- w Software development projects do not exist in isolation
 - w The software is typically developed as part of a larger business project or programme
 - w Multiple releases are needed to deliver the full business benefit

A single release (one pass through the lifecycle) as a stand alone project is a very rare corner case.

Scaling Up & Scaling Down

A Scalable Process: Planning at Multiple Levels



Adapted from Implementing RUP Within a PRINCE2 Environment: Laurence Archer, Oak IT, 2001

17

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

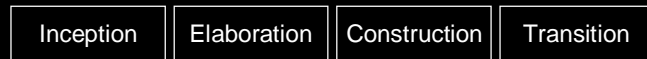
The programme management layer is present in a lot of modern businesses. Where the projects are not parts of programmes then the organizations ‘Strategic Management’ plays the same role.

If you are an IT supplier then you will typically find that the commissioning business has a business project that plays the same role.

Scaling Up & Scaling Down

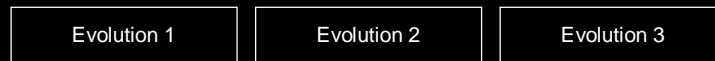
A Scalable Process: Evolution Planning

- w Very few projects deliver a single release
- § One pass through the Unified Process lifecycle



One evolution, deploying one major release

- w Most projects deliver the system in series of major releases
- § Multiple passes through the lifecycle



Multiple evolutions, each deploying a major release

18

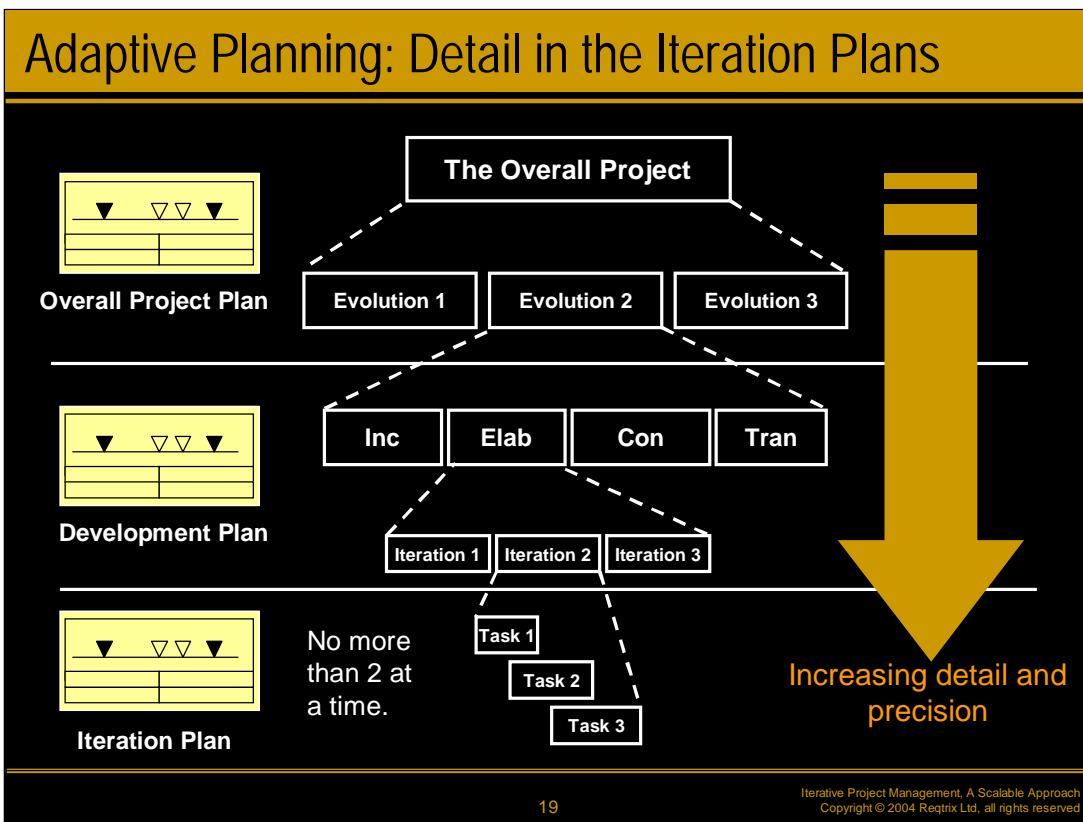
Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Most software development projects deliver the software in more than one ‘major’ release. This entails evolving the system through a series of evolutions, each of which applied the RUP lifecycle.

This is a UP evolution for each major release (Release 1, 2, 3 etc) not for each spot release. During the transition of Release 1 fixes and emergency releases may be required (Releases 1.1, 1.2, 1.3 etc). These would be the result of undertaking Transition iterations as part of the first evolution.

Unlike the phases of the process the Evolutions can overlap. We will look at this in Module 4: Phase Planning and Assessment.

Scaling Up & Scaling Down

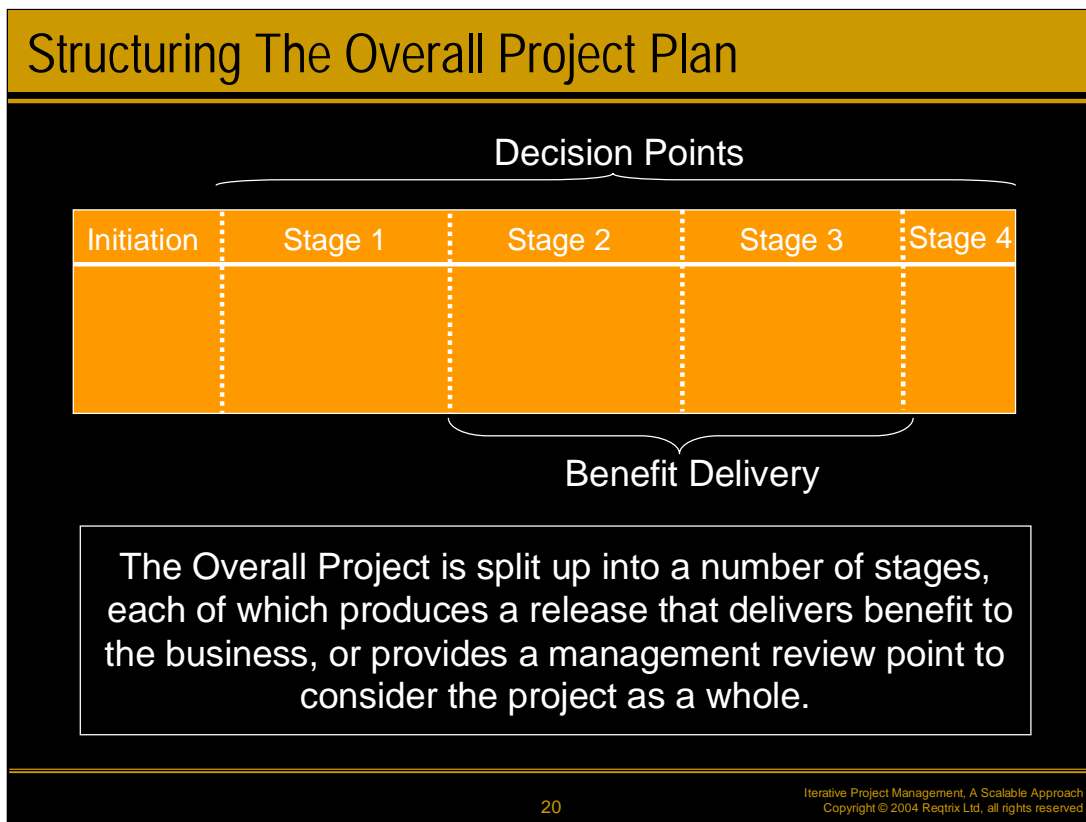


There will be one overall project plan for the project as a whole.

There will be one development plan for the current evolution.

There will be an iteration plan for the current iteration. As this iteration progresses and its results start to become apparent the planning of the next iteration starts.

Scaling Up & Scaling Down



The Overall Project is split up into a number of stages, each of which delivers benefit to the business.

The stages

Partitions of the project with decision points

Management Stage

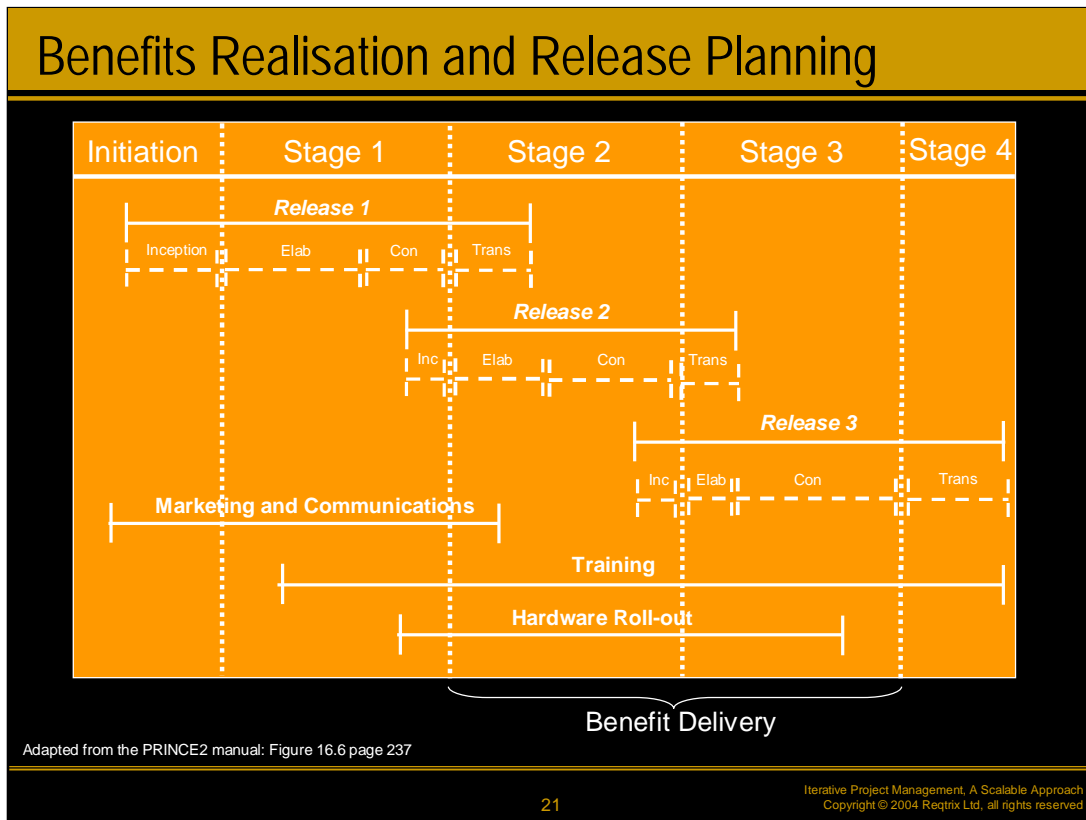
Equates to the commitment of resources and authority to spend

Technical Stage

Equates to the use of a particular set of specialist skills

For example software development skills

Scaling Up & Scaling Down



Typically the software is delivered to the business in a number of major releases. Each major release is considered to be a full RUP evolution and typically the end of its Construction Phase aligns with the end of a stage.

Scaling Up & Scaling Down

Typical Project: Lessons Learned

- w There is no such thing as a typical project
 - w The published numbers should be used to challenge, not create, plans
- w The only predictable split in a project is between the Elaboration and Construction Phases
- w Plan at all three layers
 - w Release planning is as important as phase and iteration planning
- w You cannot iterate successfully without an overall project plan to provide context to the iteration results
- w PRINCE2 provides a proven method for the management of the overall project

Scaling Up & Scaling Down

Agenda

- w Background
- w A Question of Scale
- w A Typical RUP Project
- w Scaling Up - Managing Large Projects
- w Scaling Down - Managing Small Projects
- w Conclusions

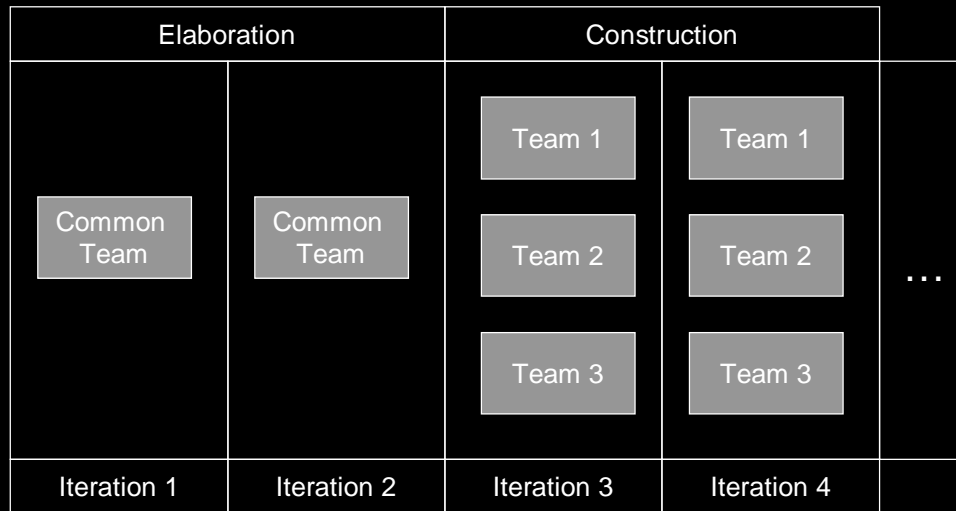
Scaling Up & Scaling Down

Large Software Development Efforts

- w Number of software products: >1
- w Multiple Project Teams
- w Multiple Releases
- w Duration: > 9 months
- w Number of people: > 30
- w Release Cycle: 6 – 12 months
- w Iteration Length: > 4 weeks
- w Formality: > Medium
- w Team Distribution: Dispersed

Scaling Up & Scaling Down

Scaling Up: Multi-site, Multi-team Development



Adapted from: Agile and Iterative Development – A Manager's Guide, Larman, Addison Wesley, 2004

25

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

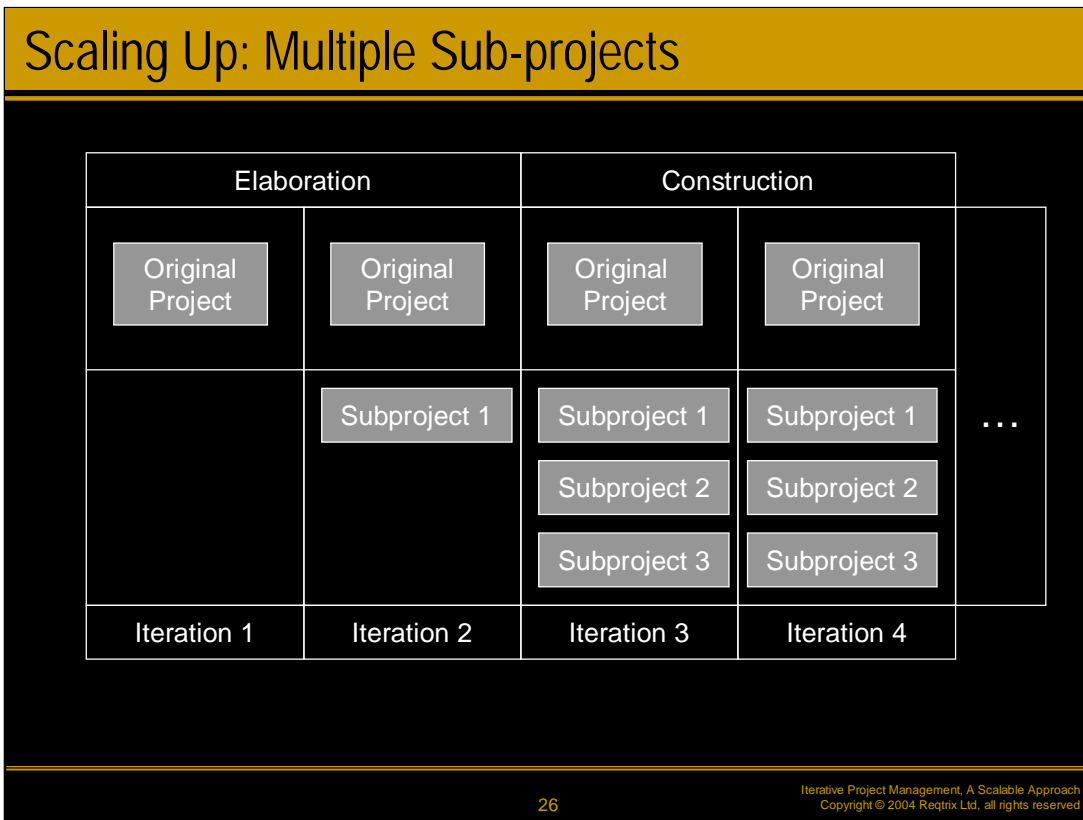
Start with a small-team (preferably co-located) who will create the architecture, undertaking the Inception and Elaboration Phase.

Members of the common team go on lead the sub-teams in the construction phase.

The construction teams are typically made up of specialist skills aligned to sub-systems or disciplines.

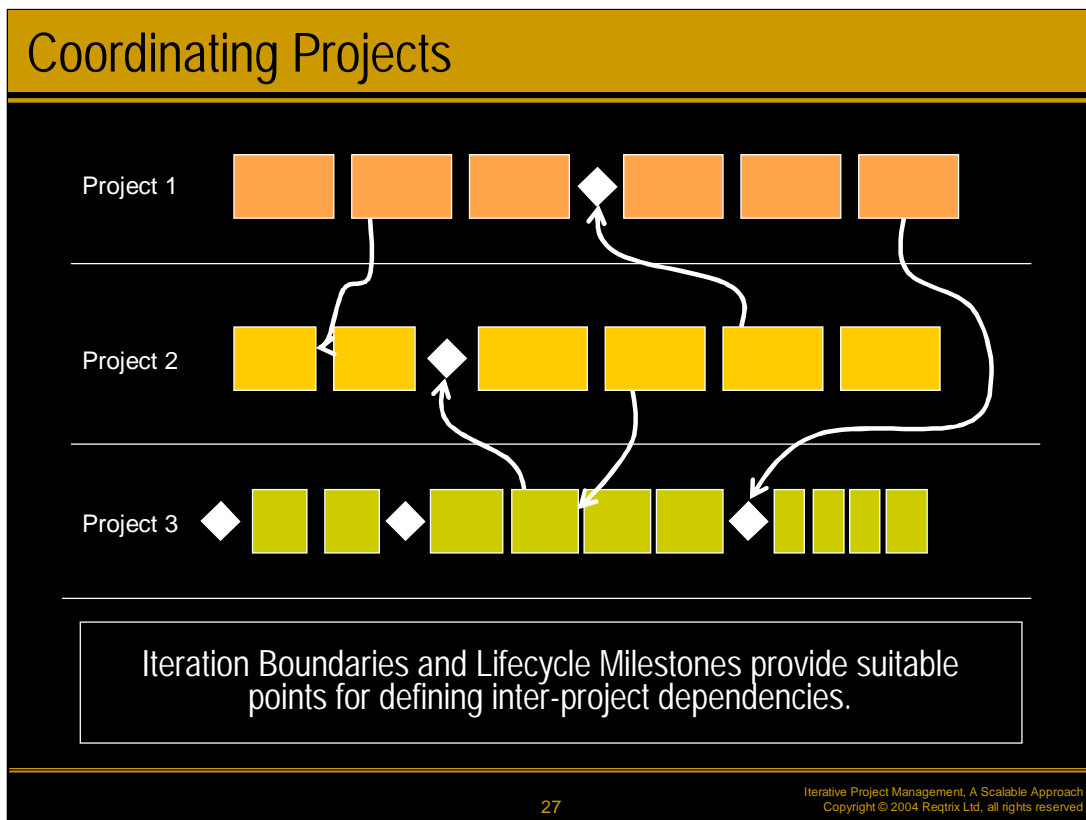
Note: in this model there is only one plan, one set of iterations and one management team.

Scaling Up & Scaling Down



We start as though we had a small project. As the architecture and the requirements start to stabilise additional sub-projects are spun off as required.

Scaling Up & Scaling Down



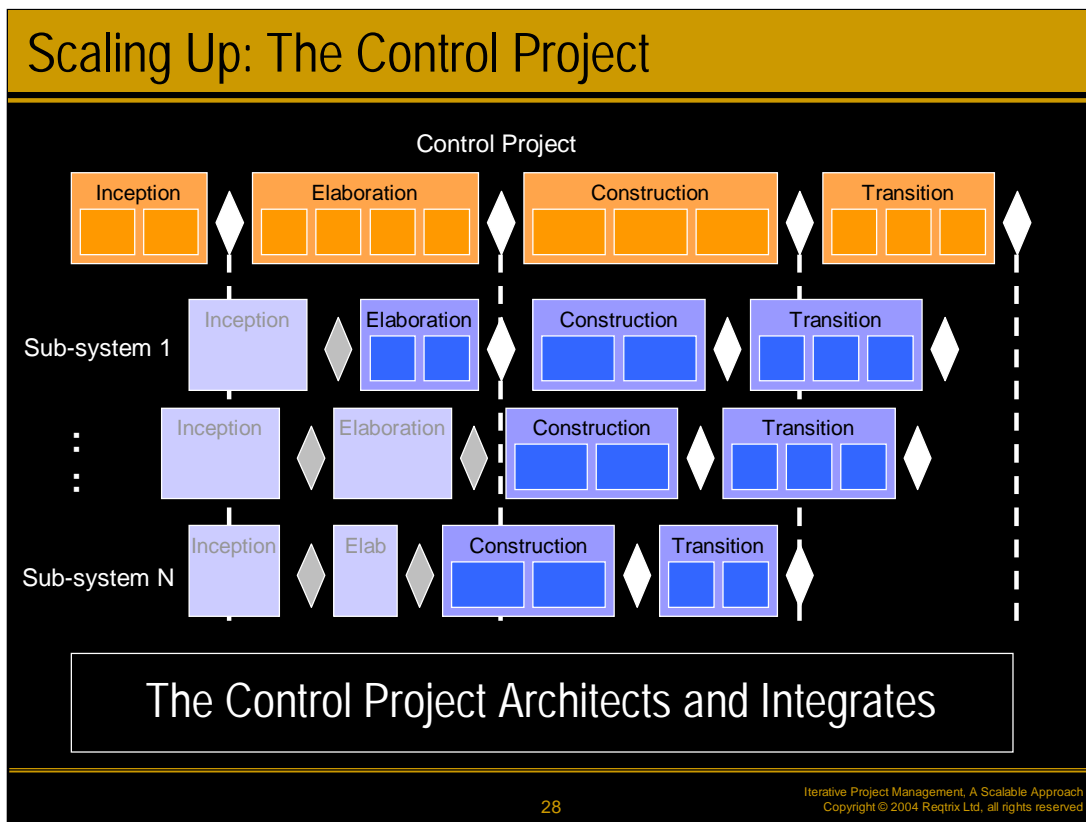
The most stable points in an iterative project's lifecycle are the iteration boundaries and the lifecycle milestones. These are points when the project is in a steady state and artefacts and releases are most likely to be in a fit state for re-use.

If a project manager knows that another project depends on the results of an iteration she can make sure that the elements depended upon are the highest priority for the team and do not get moved to a later iteration if things are not going so well.

The fact that a project has passed one of the lifecycle milestones provides insight into the stability and state of the products it is producing.

We exploit these relationships when we start to overlap our the evolutions producing our series of major releases.

Scaling Up & Scaling Down



The control project starts by capturing the requirements for the project as a whole and creating the overall architecture. As the subsystems within the architecture become stable, and their roles and responsibilities emerge, sub-projects are created to develop the sub-systems. Typically the Inception Phase, and sometimes the Elaboration Phase, for the subsystem project is performed by the control project. This leads to the sub-projects appearing to go straight into their Elaboration or Construction phases. The greyed phases in the slide above are intended to represent these 'virtual' phases.

As the sub-projects iterate and produce releases of their sub-systems these are delivered to the control project, which then integrates them into a single system and system tests it.

The Control Project will integrate the results of all of the sub-system's last completed iterations.

Scaling Up & Scaling Down

Projects, Programmes and Portfolios

w Project

§ A temporary endeavour undertaken to create a unique product, service or result.

w Programme

§ A group of related projects managed in a coordinated way. Programmes usually include an element of ongoing work.

w Portfolio

§ A set of projects grouped together for financial control purposes.

Programmes Leverage Commonality

29

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

The first two definitions are from the PMBOK, the third the authors own.

The key thing about a true programme is that the benefit accrued by executing the projects in a coordinated way is greater than that which would accrue if the set of projects were all done individually. This is the real difference between programme management and portfolio management.

Programme can be:

Strategic Programme – shared vision and objectives

Business Cycle – shared budget or resources

Infrastructure Programme – shared standards

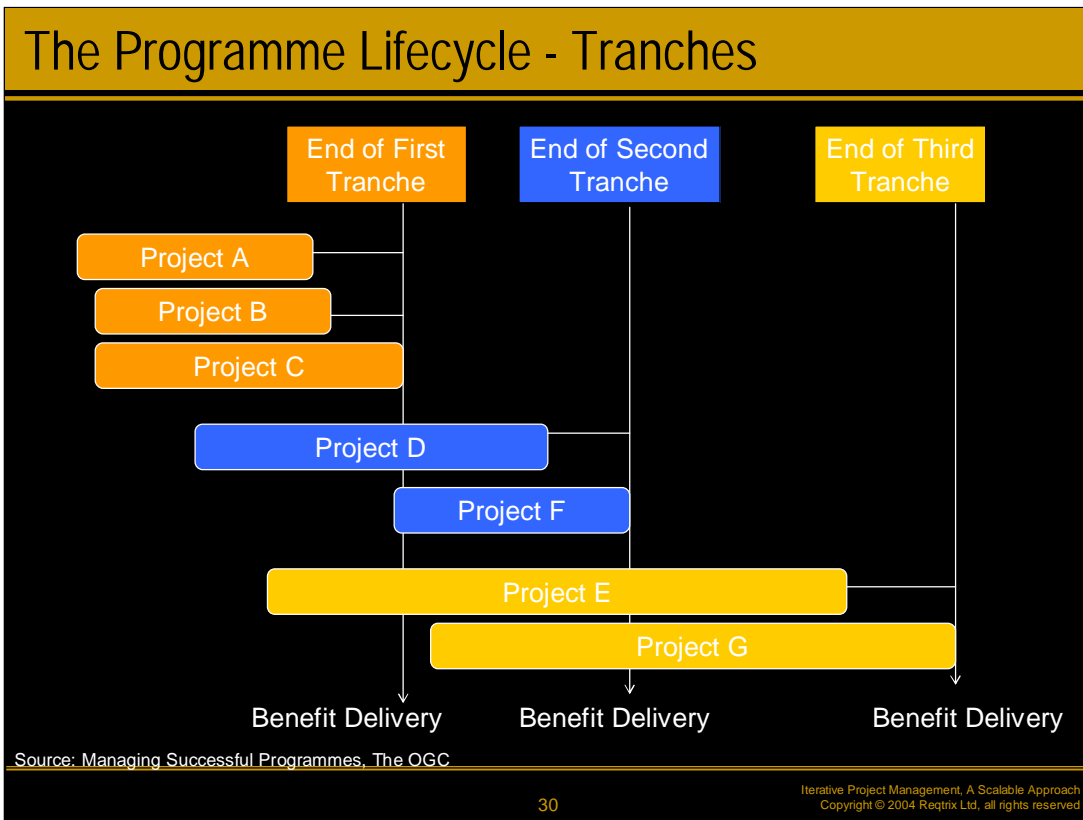
Research and Development – shared assessment / focus

Partnership – collaborating organizations

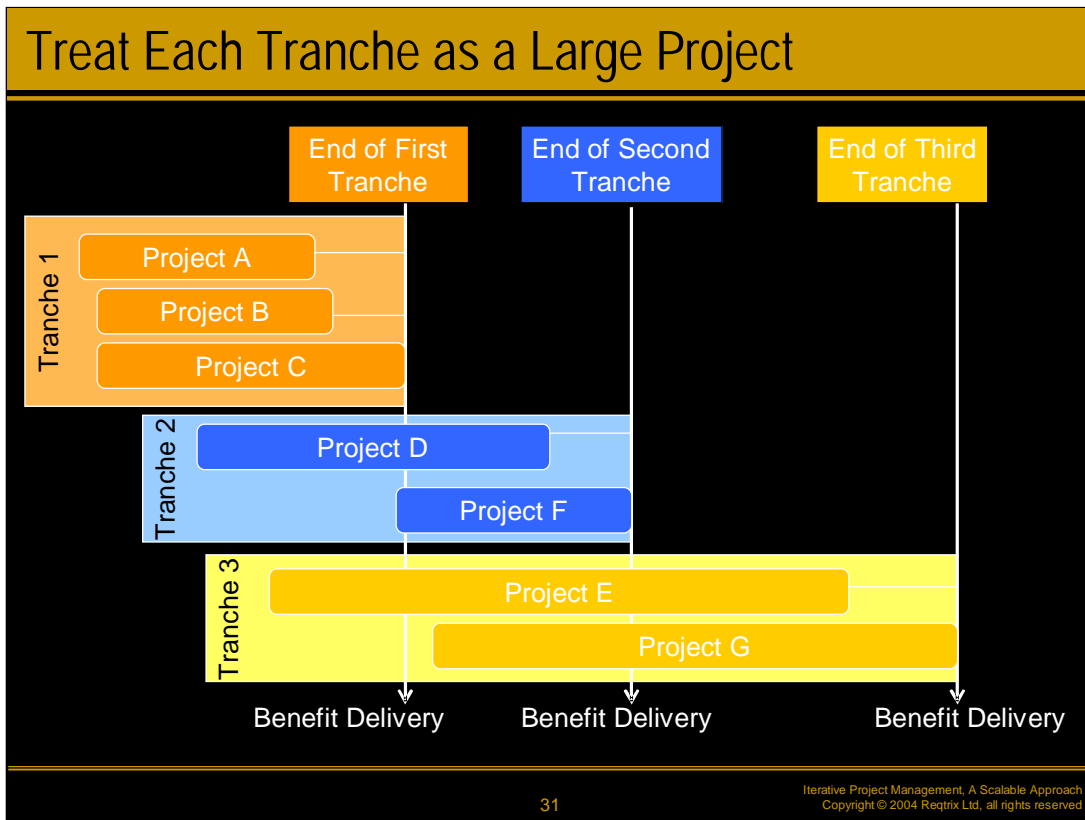
or a mixture of these.

Within a Programme IT Projects can share objectives, requirements, architecture, process, Assets, components and responsibilities.

Scaling Up & Scaling Down

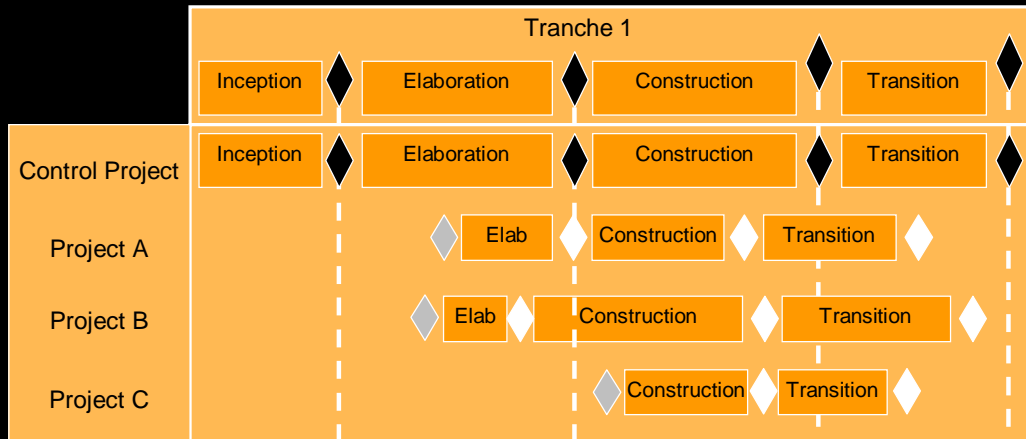


Scaling Up & Scaling Down



Scaling Up & Scaling Down

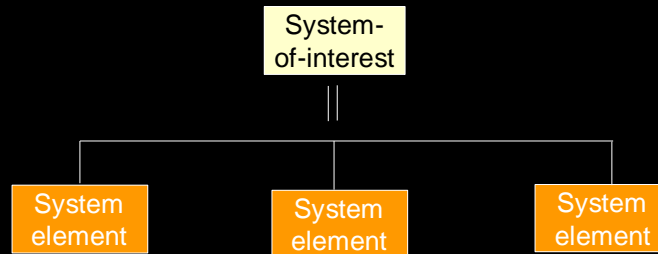
Give Each Tranche A Control Project



The Tranche and the Control Project Have Concurrent Lifecycles

Scaling Up & Scaling Down

Scaling Up: Systems and System Elements



A system is completely composed of a set of interacting system elements, our components.

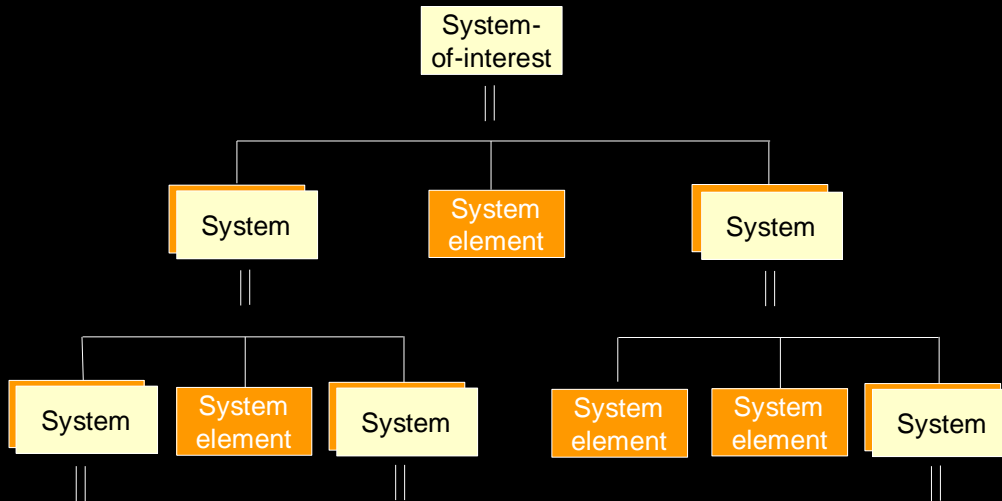
Source – ISO/IEC 15288:2002

33

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Scaling Up & Scaling Down

Scaling Up: Reifying Elements To Be Systems



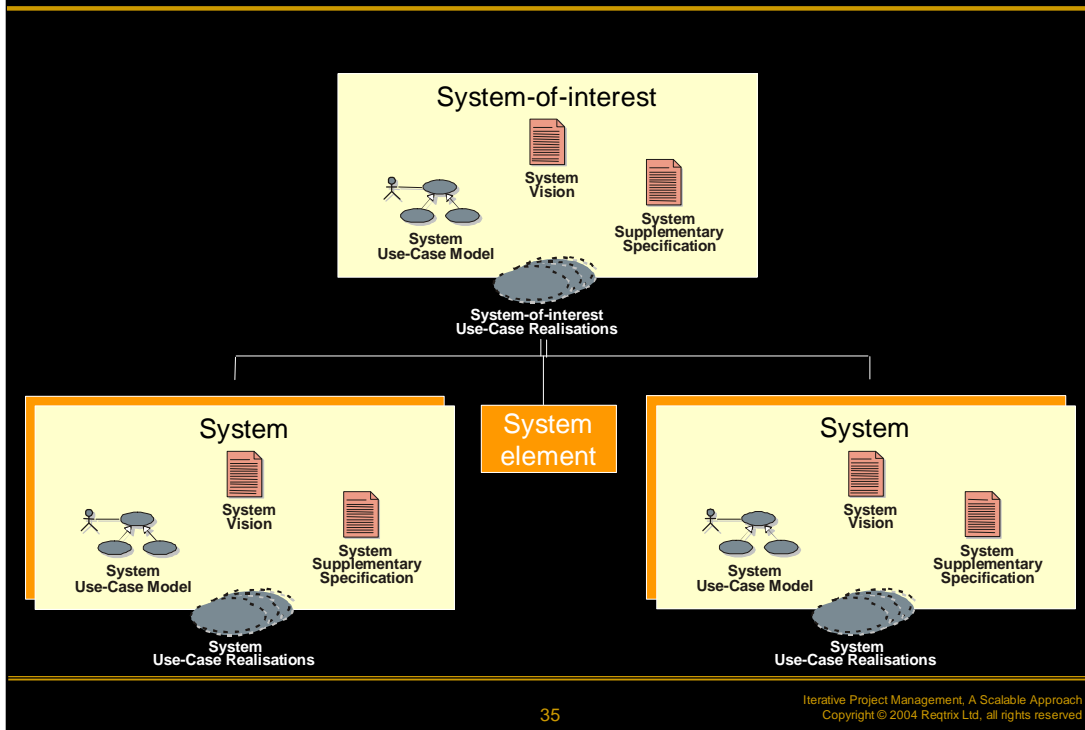
Source – ISO/IEC 15288:2002

34

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Scaling Up & Scaling Down

Use Cases – A Recursive Requirements Solution



The whole cannot be defined without understanding the technicalities of the parts, and the parts cannot be defined in detail without understanding the whole.

However the risk is that we will forget to consider how the solutions for each of the smaller problems impact the overall solution and get bogged down in building parts that don't fit together.

If you are interested in learning more about using use cases for this kind of 'systems engineering' then we would recommend the Reqtrix Advanced Use-Case Modelling course, which covers this and other patterns for large scale requirements capture.

Scaling Up & Scaling Down

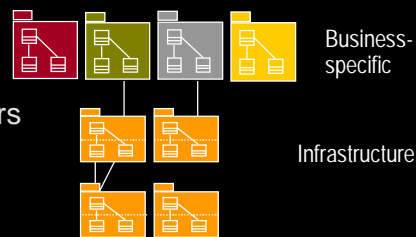
Case Study: Outsourcing Subsystem Development

w A Programme Building a Large System

- § Each sub-ordinate system to be built independently
- § Some sub-ordinate systems already exist within the organisation
- § Some sub-ordinate systems to be built in-house
- § Some sub-ordinate systems to be out-sourced as insufficient skills and resources in-house
- § Architecture and infrastructure defined for system as a whole

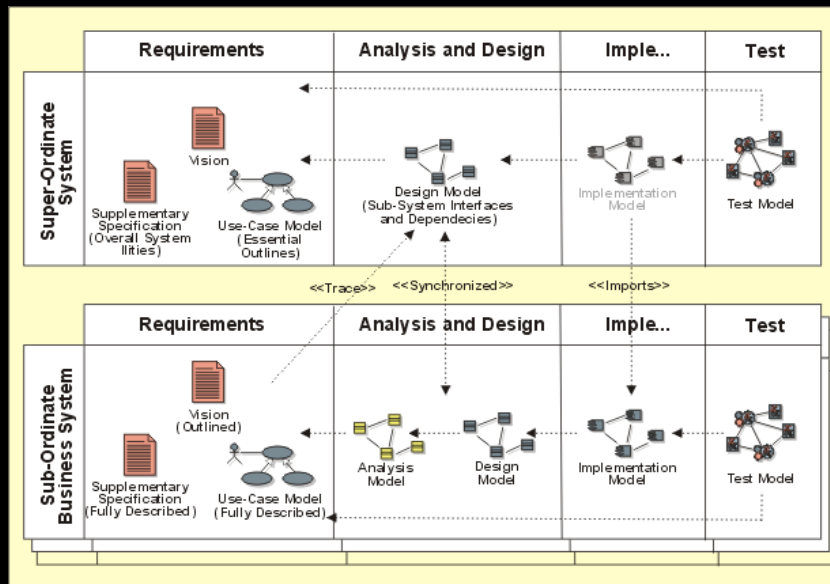
w Architecture:

- § Four collaborating business sub-systems to be built
 - Each with its own stakeholders and business purpose
- § 4 major infrastructural components to be acquired



Scaling Up & Scaling Down

Case Study: Artefacts and Their Relationships



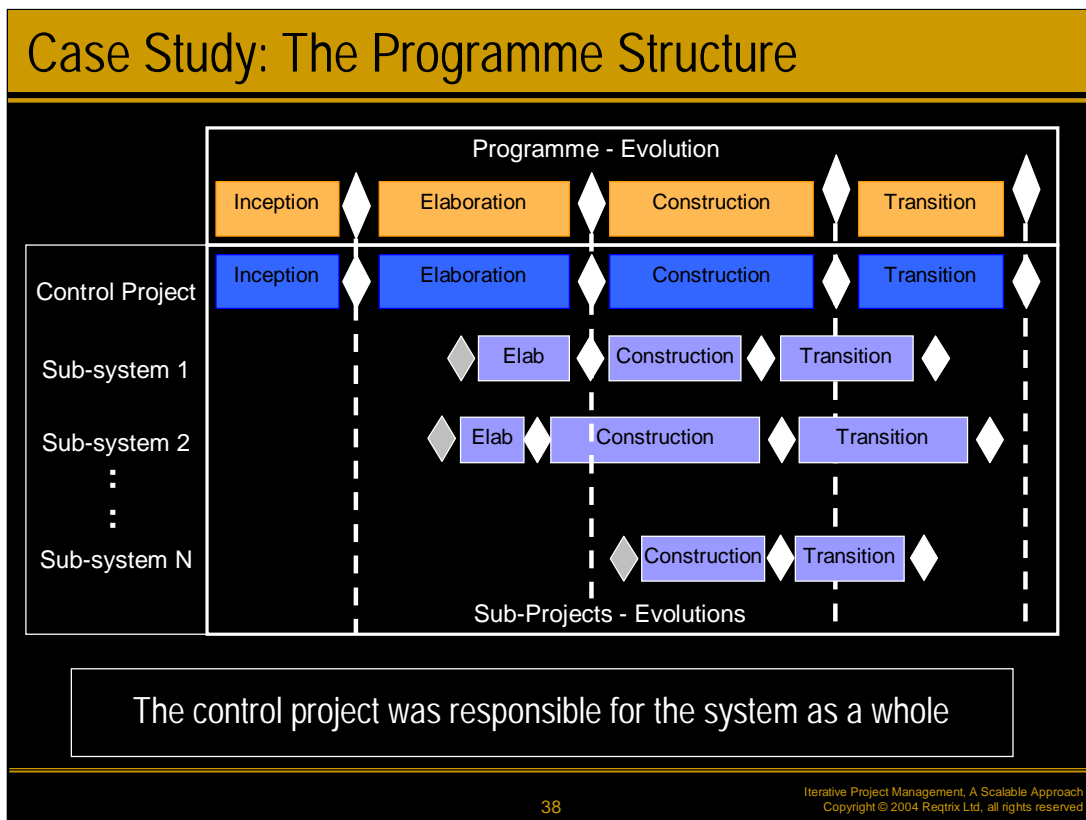
Adapted from Modelling For Enterprise Initiatives with the IBM Rational Unified Process, Eeles and Ericsson, Rational Edge 2003

37

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

As you can see life is starting to become more complicated.

Scaling Up & Scaling Down



The control project starts by capturing the requirements for the project as a whole and creating the overall architecture. As the subsystems within the architecture become stable and their roles and responsibilities stabilise sub-projects are kicked off. Typically the Inception Phase, and sometimes the Elaboration Phase, for the subsystem project is performed by the control project. This leads to the sub-projects appearing to start straight into Elaboration or Construction. The greyed phases in the slide above are intended to represent these 'virtual' phases.

As the sub-projects iterate and produce releases of their sub-systems these are delivered to the control project, which then integrates them into a single system and system tests it.

The Control Project will integrate the results of all of the sub-system's last completed iterations.

Scaling Up & Scaling Down

Scaling Up: Lessons Learned

- w The pace of the overall project is set by the weakest link
- w Decomposition can cause adversarial and competitive relationships to appear between teams
- w Think 'evolutions and release planning' before 'system of systems'
- w A stable architecture is required to support parallel working
- w The techniques can be applied recursively – the higher level, integrating projects will iterate more slowly

Scaling Up & Scaling Down

Agenda

- w Background
- w A Question of Scale
- w A Typical RUP Project
- w Scaling Up - Managing Large Projects
- w Scaling Down - Managing Small Projects
- w Conclusions

Scaling Up & Scaling Down

A Small Project

- w Number of software products: 1
- w Single Project Team
- w Duration: < 6 months
- w Number of people: < 5
- w Iteration Length: < 4 weeks
- w Formality: Low
- w Team Distribution: Co-located

Scaling Up & Scaling Down

Scaling Down: Reducing Overheads

- w Its not the things you do or produce that vary with project size it's the extent of their contents.
- w Many products can be combined to reduce the artefact set
- w Typically smaller projects have better communication and require less documentation
- w Typically small projects are expected to be fast and so have less formality and ceremony

42

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Experience of the Unified Process indicates that small projects still requires all of the key artefacts, it is just that the artefacts themselves are a lot less complex and therefore smaller. Where artefacts are not required it is typically not because the project is small but because disciplines or particular artefacts are not applicable to the sort of problem the project is solving or the environment within which the project is working.

Scaling Up & Scaling Down

Scaling Down: How Low Can You Go?

w Time – A single iteration?

§ Is this still an iterative project

§ Typically 3 iterations is considered small

w Time boxes – Less than a week?

§ 1 week iterations are possible but need to be very informal and very disciplined

§ Typically 2 weeks is as short as iterations get

For any project under 4 weeks consideration should be given on whether to iterate or not.

Note: two 2 week time boxes is still better than one 4 week one.

Scaling Up & Scaling Down

Scaling Down: Lessons Learnt

- w Short iterations are very fragile
- w Small teams benefit from light-weight informal processes
- w To move quickly small projects require dedicated resources
- w Small projects are typically more focussed, more productive and more effective
- w Small projects still need to consider all the same things as a large project, just on a smaller scale

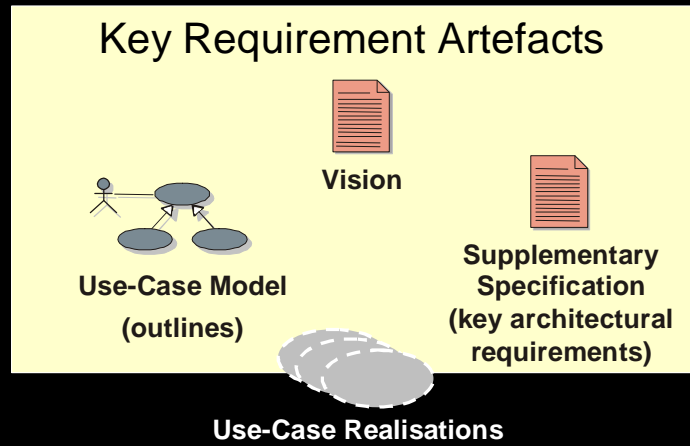
Scaling Up & Scaling Down

Agenda

- w Background
- w A Question of Scale
- w A Typical RUP Project
- w Scaling Up - Managing Large Projects
- w Scaling Down - Managing Small Projects
 - § Applying the lifecycle to small endeavours
- w Conclusions

Scaling Up & Scaling Down

The Good News Is.....



... Everything Starts The Same

46

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Similar thing applies to the outlined project management approach and the architectural approach.....

Scaling Up & Scaling Down

Getting Started

- w Start with a small team focussed on the business and technical risks
- w Start with 4 week iterations
- w Plan to deliver benefit every 6-9 months
- w Do release planning as well as iteration planning
- w Start with a co-located team
- w Address the architectural risks before scaling up

Scaling Up & Scaling Down

Getting Started: Don't Assume the Project is Large

- w Start with the assumption that it is small**
 - § If you start big it will stay big
 - § If you don't fight to keep things small they will naturally tend to get large
- w Do everything within your power to keep the project small**
 - § The larger the project the more likely it is to fail
- w Exploit the layering to keep things small at each layer**
 - § To be comprehensible, plans must be small

Scaling Up & Scaling Down

Getting Started: Your First Iterative Project

- w Needs a team that wants to iterate
- w Needs to be big enough to have at least four iterations
 - § It will take a few iterations for the benefits to be accepted by the team
 - § The team will need time to get used to the new ways of working
- w Requires a full development team to be in place
 - § Testers are needed early
- w Will probably be over planned
- w Is unlikely to be more productive than a non-iterative project
 - § But it will be less risky

Scaling Up & Scaling Down

Conclusion

- w Iterative project management provides a scalable, adaptable approach to managing software development
- w Project size has many dimensions, many of which are unknown at the start of the project
- w Start small and scale up where necessary
- w The mechanisms are the same regardless of the size of the endeavour being undertaken

Scaling Up & Scaling Down

QUESTIONS

51

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Scaling Up & Scaling Down

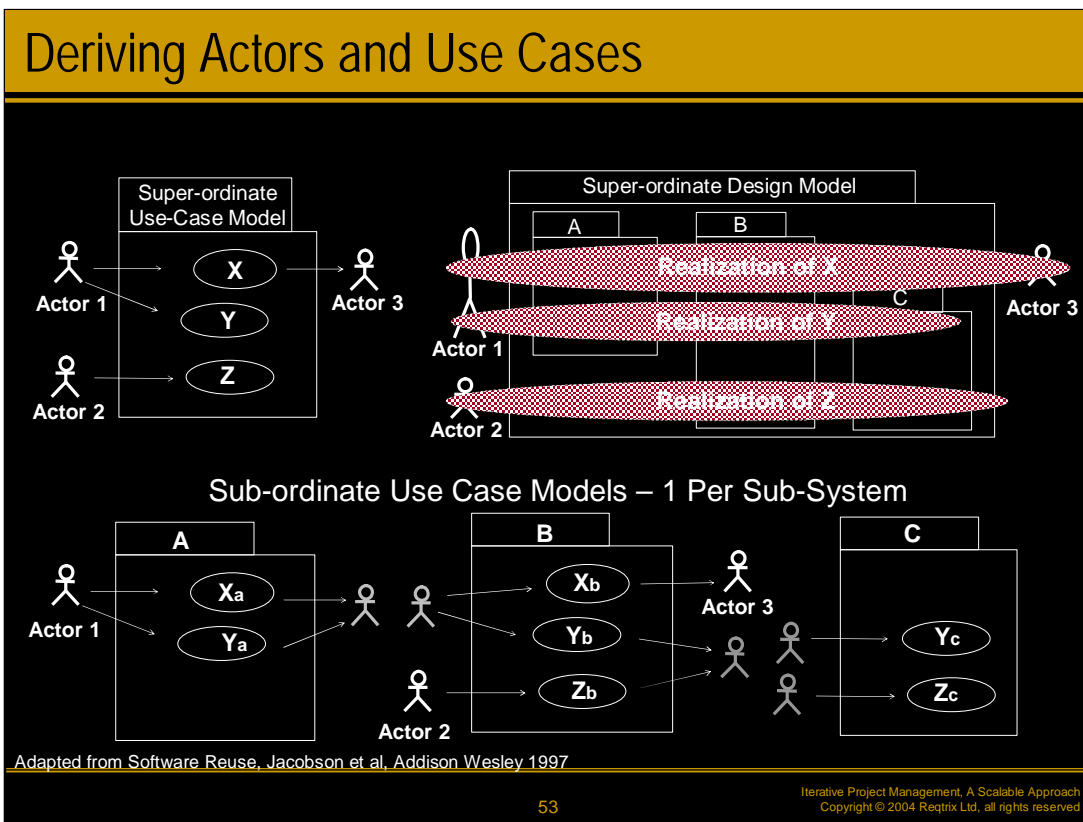
Thank You

*Ian Spence
Reqtrix Ltd
ian@reqtrix.com*

52

Iterative Project Management, A Scalable Approach
Copyright © 2004 Reqtrix Ltd, all rights reserved

Scaling Up & Scaling Down



For more information on deriving actors and use cases inside a system of systems see *Software Reuse* by Ivar Jacobson, Martin Griss & Patrik Jonsson

Addison Wesley, 1997 ISBN 0-201-92476-5