

Going Agile @ F-Secure

Benefits, Challenges, Lessons Learned

Software Engineering Conference (Russia) 2007



Contents



1. F-Secure in a nutshell
2. Transition to Agile at F-Secure
3. F-LEX, the new agile development framework
4. Benefits, challenges and future steps

F-Secure in a nutshell

The fastest growing AV-company

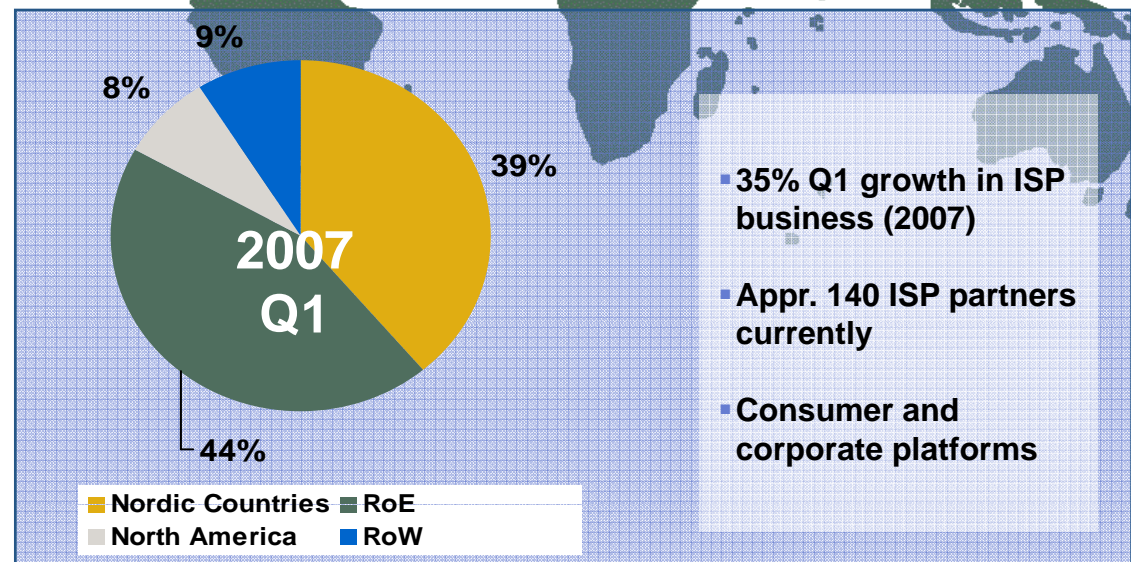
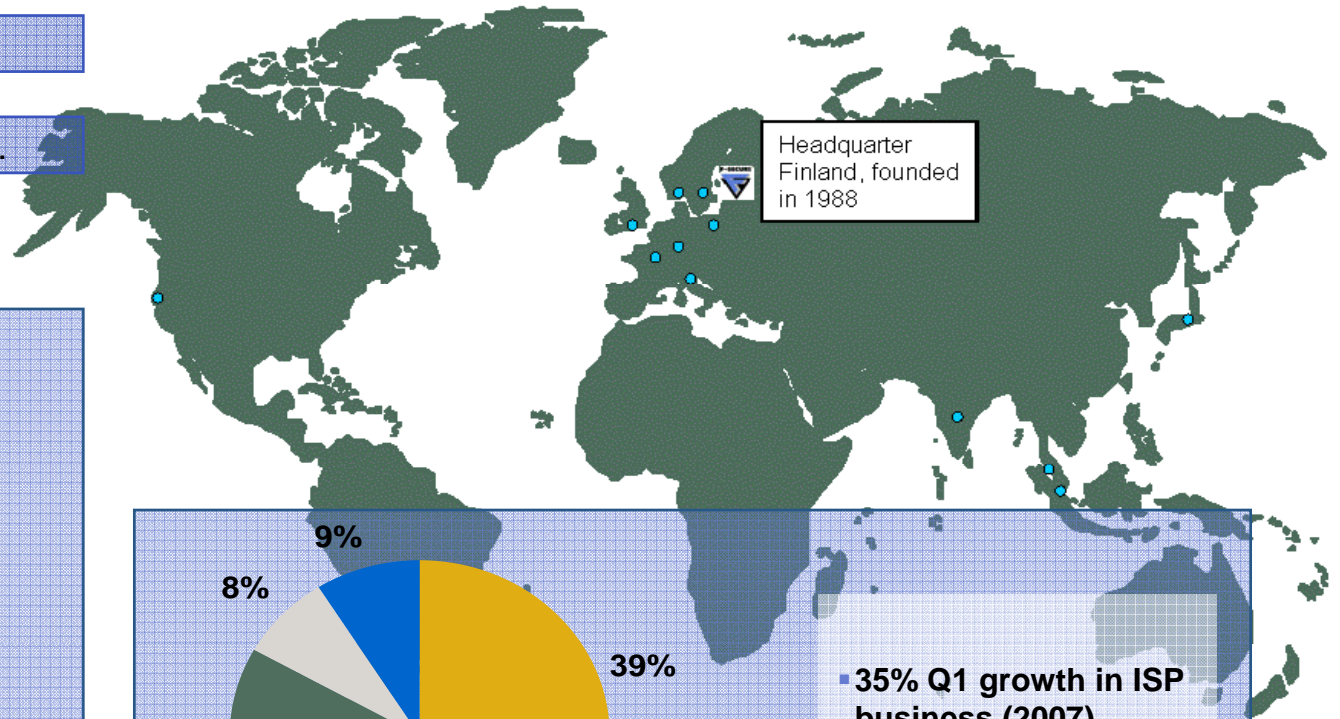


1988 Founded

1999 IPO Helsinki Stock Exch.

2007

- Market Cap.: ~380+ M€
- Revenue 2006: 80.7 M€
- 2006 Growth: 31%
- Staff: 500+ FTE
- Subsidiaries:
 - Helsinki and Oulu
 - Stockholm, Oslo, Copenhagen
 - London
 - Paris
 - Munich
 - Warsaw
 - Milan
 - Singapore
 - Mumbai
 - Kuala Lumpur
 - Tokyo
 - San Jose
 - St.Petersburg (RDC)



- 35% Q1 growth in ISP business (2007)
- Appr. 140 ISP partners currently
- Consumer and corporate platforms

Security as a Service

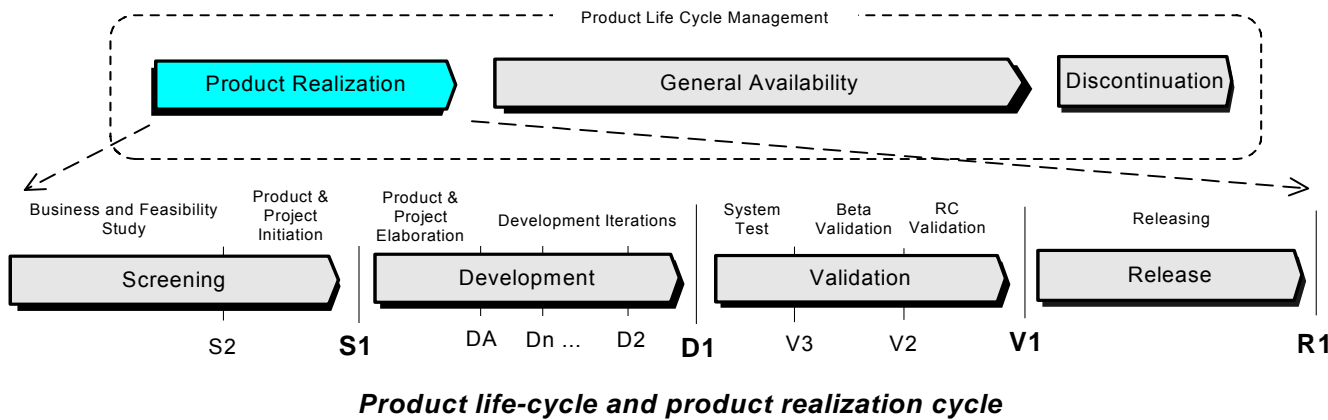


Benefits to subscribers

- Continuity – subscription service
- Safe Internet experience
- Ease of purchase
- Single point of contact
- Trusted partner - Local contact

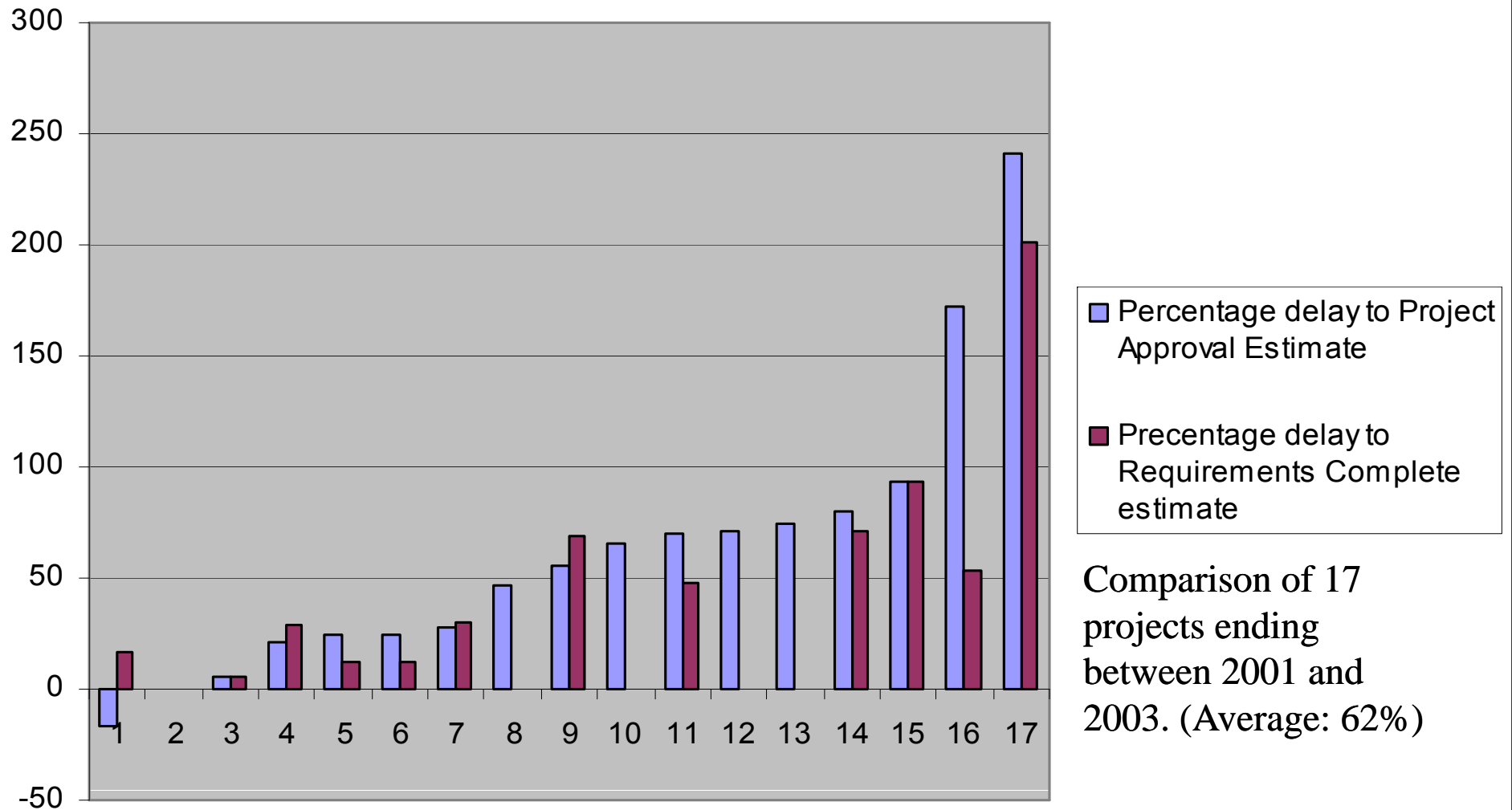


FPRP – F-Secure Product Realization Process (1999 – 2005)



FPRP – F-Secure Product Realization Process

Project schedule accuracy statistics before Agile



Comparison of 17 projects ending between 2001 and 2003. (Average: 62%)

Principal Problems in the Old Model



1. “Frozen requirements” changed continuously
2. Heavy planning and documentation throughout the development
3. Workload for change management grew exponentially towards the end of the project
4. 90% done syndrome
5. Validation phase was always much longer than planned despite of vast automation
6. Culture: relaying only on individuals who are high on rank on organizational hierarchy

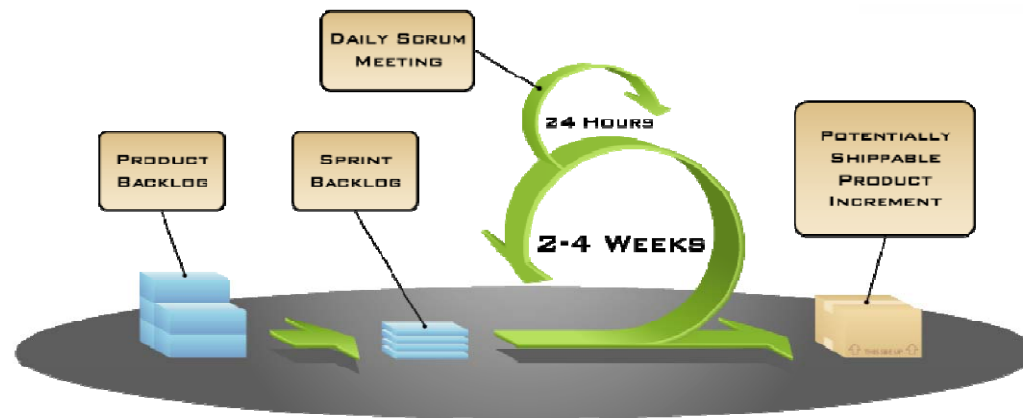
Agile pilots at F-Secure...



Mobile-D in one project

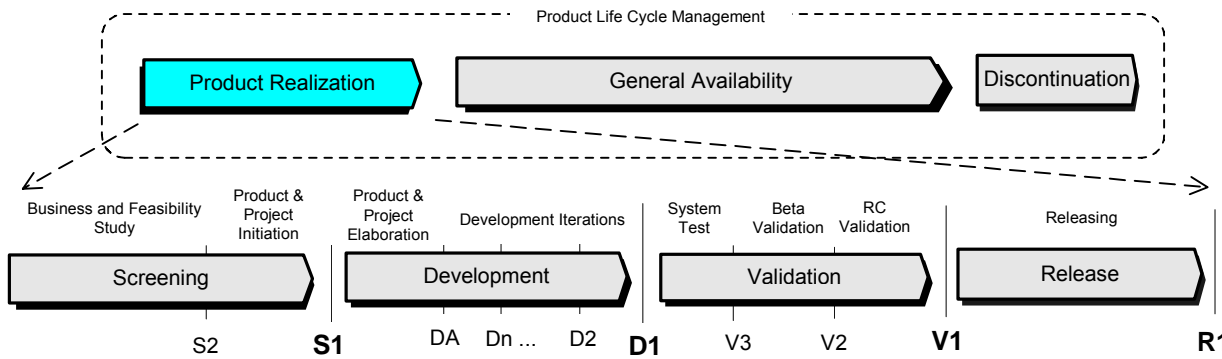


Scrum in two projects



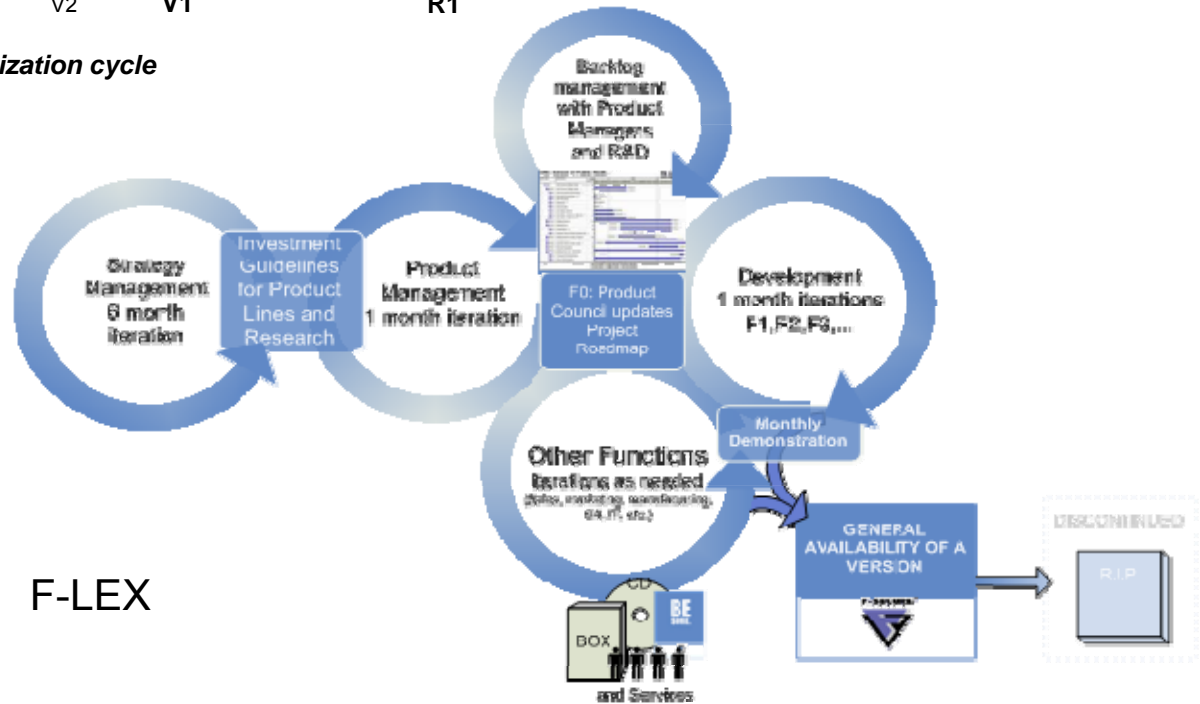
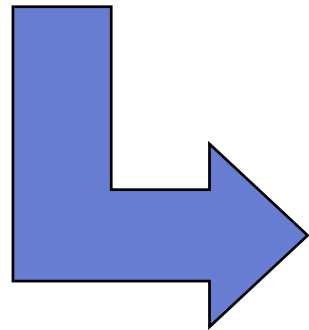
COPYRIGHT © 2005, MOUNTAIN DOG SOFTWARE

F-Secure Development Process Change



FRRP – F-Secure Product Realization Process

Product life-cycle and product realization cycle



F-LEX

F-LEX Purpose



Purpose

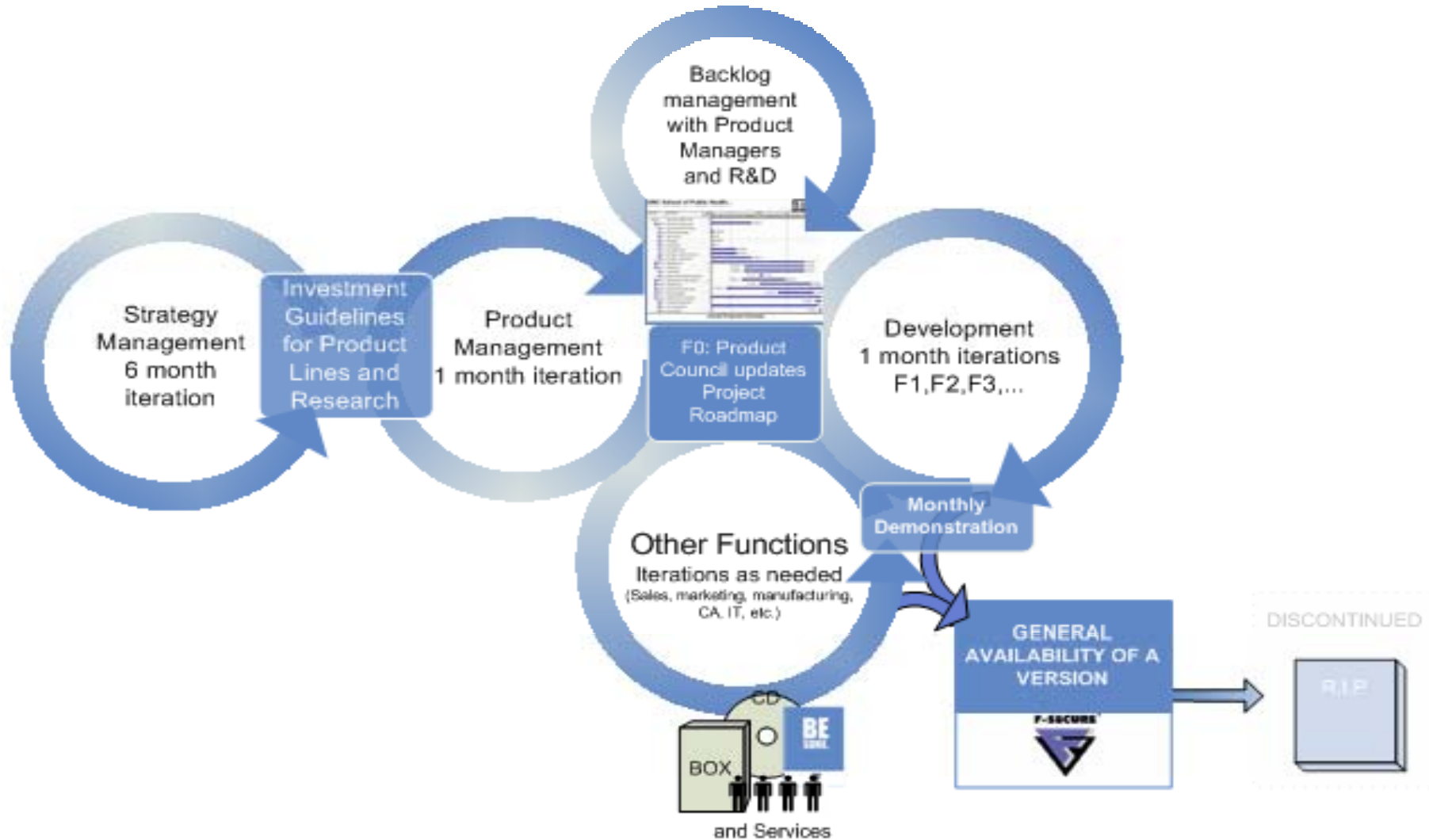
- Describe the **best known** method at F-Secure to make ideas into commercial products efficiently and predictably.
- In other words: ideas to good products really **quickly**

What F-LEX is

- F-LEX is F-Secure product **life cycle** process
- F-LEX is targeted for the **whole corporation**
- Outlines the **processes and practices** which are **commonly agreed** and followed in each organization and project.
- F-LEX is based on **Incremental & Iterative** and Agile models such as Scrum, Evolutionary development, Crystal, XP and Mobile-D



F-LEX, Agile Development Process 2005 - now

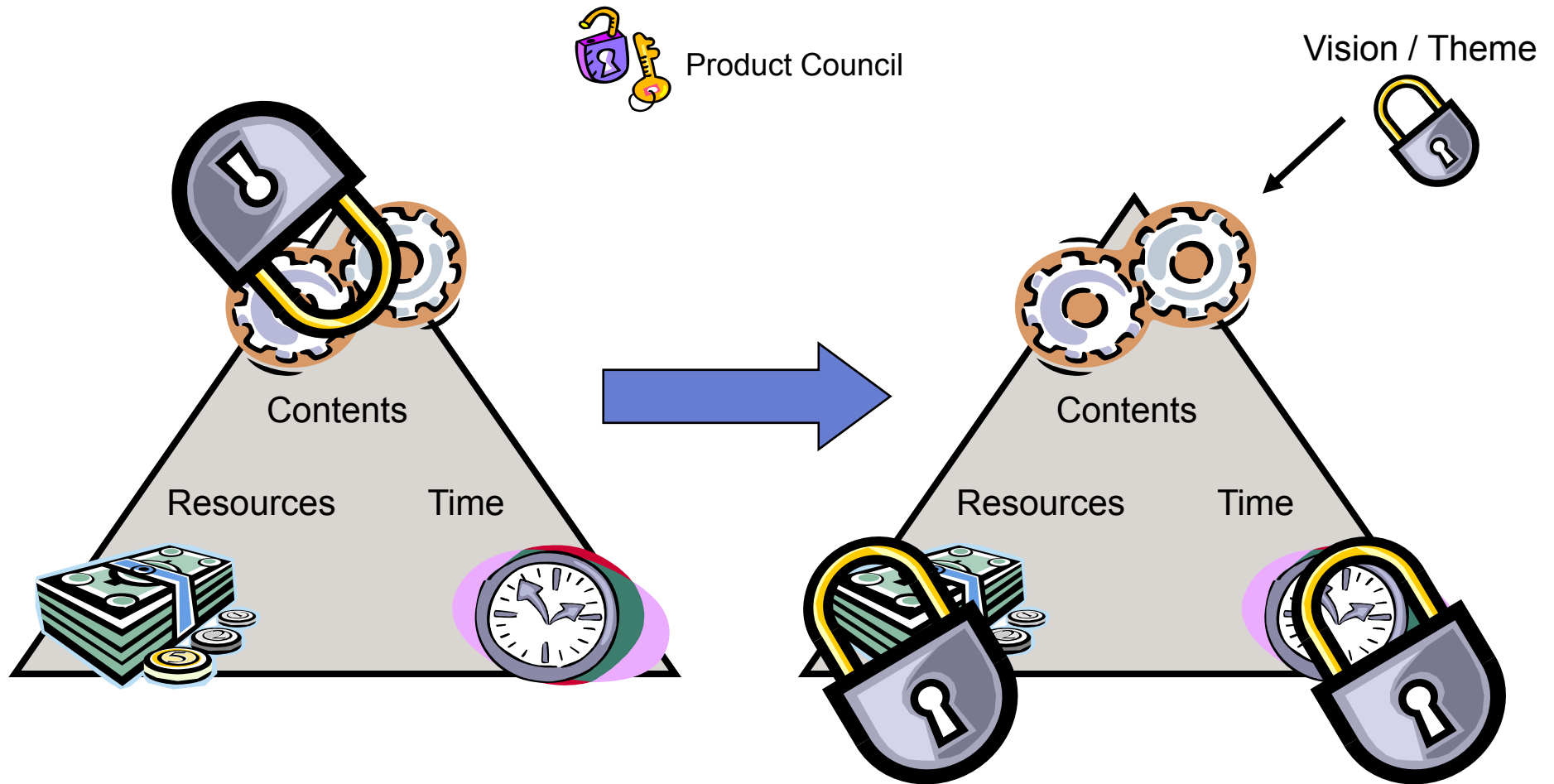


What is new?



- Change of culture. Easier to do when we clearly change to a new process, not just FPRP 3.0
- In R&D
 - A stable software version done once a month
 - Focus on bug fixing all the time, not just at the end
 - SWE's do enhanced unit testing for their own code
 - Clearly iterative delivery, a live demo every month
 - More active Product Management who maintains the backlog
 - Goal: Continuous builds with automated testing
- Unlike FPRP, FLEX describes the **product life cycle management**, not just how projects are run.
- Unlike FPRP, FLEX describes how **decision of phasing out complete products** and earlier product versions are done.
- Unlike FPRP, FLEX clarifies **product management over several projects**
- Unlike FPRP, FLEX describes **project portfolio management**.

Budget & Theme Driven Projects

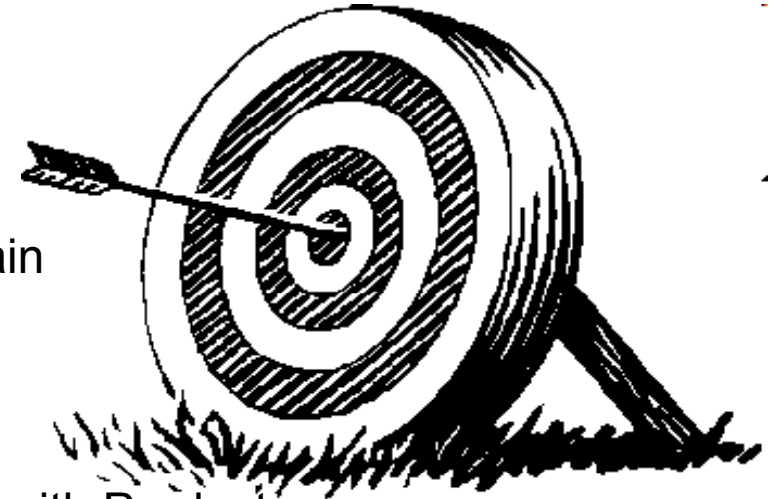


Goal: Define the rhythm of operations and ease planning

Key activities and concepts that we are focusing on



1. **Incremental** development. Ready for at least internal release after each sprint
2. Helping Product Management to make and maintain good **Product Roadmaps**
3. Helping Product Managers to make and maintain excellent **Product Backlogs**
4. Planning **project roadmaps** and discussing them with Product Managers
5. Concentrating on doing the **retrospectives** and **planning days** very well
6. **Visibility** should be a major agile principle
7. **Backlogs** for common components
8. Defining more exactly what **DONE** means for a Product Backlog item



Benefits gained so far



- Working culture is changing
- Clear guidance
- Improved risk management
- Resources and schedule are fixed, the scope is not
- Much better visibility: to strategy, operations and projects
- Our customers are truly involved
- Frequent releases

- Sprint backlogs
- Burn down charts
- Product backlog
- Release backlog
- Weekly report
- Monthly demonstration
- Project meetings and their possible minutes



Customer involvement



Lazy way: Product Manager



Purpose: early release of the evolving product for *review and feedback*

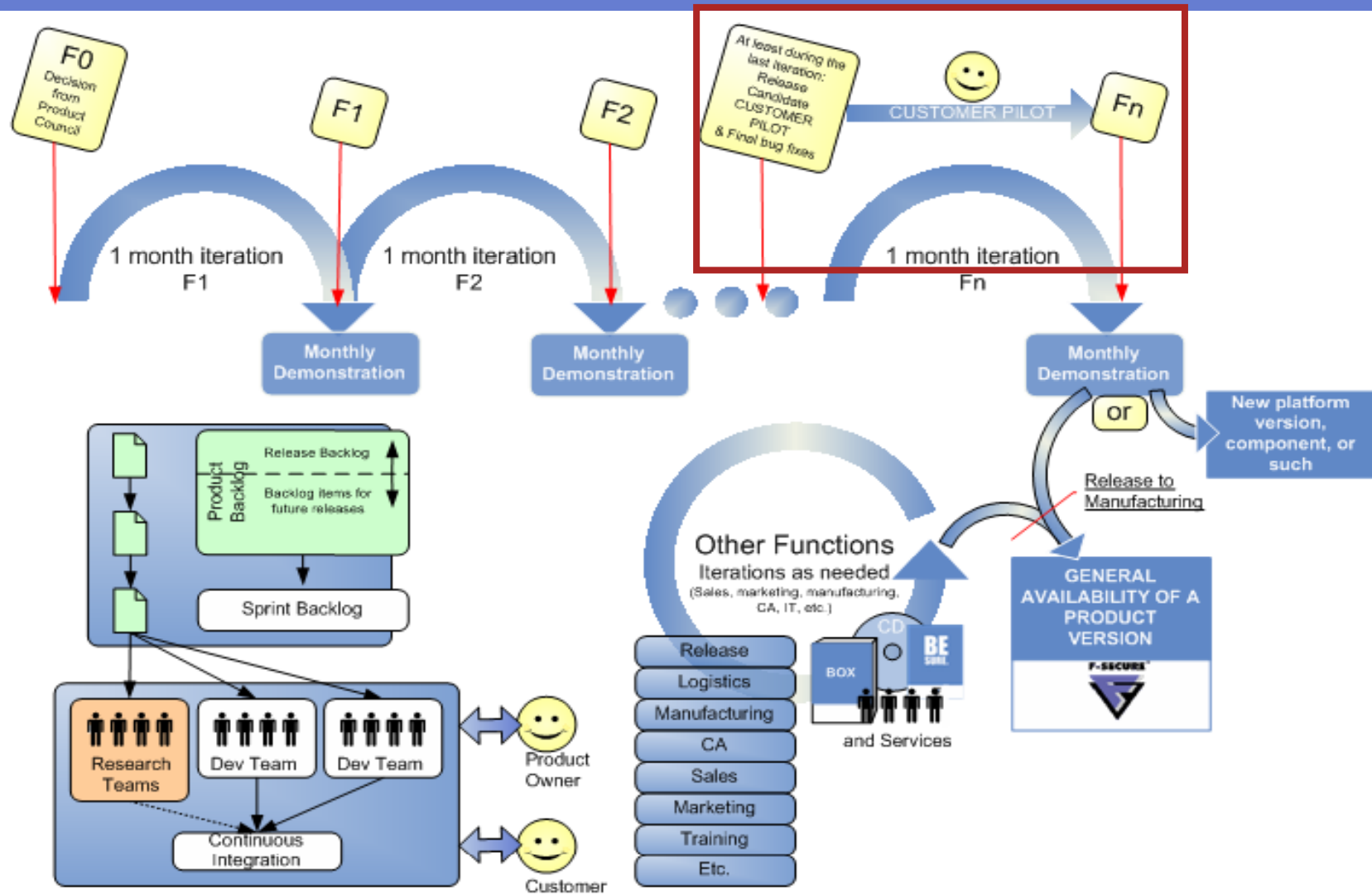
Expensive way, but not as time consuming: outsourcing and research



The best: direct contact with the customer



Customer Betas



Challenges tackled



- Making teams work as teams, not group of individuals
- Having system focus for area to deliver, taking responsibility
- Delivering components for baseline without breaking the system
- Sharing information within the team for one goal
- Close cooperation and collaboration between Software and Quality engineers
- Quick private builds for the whole product
- Contact with customers and relevant stakeholders

Keep items (good practices)



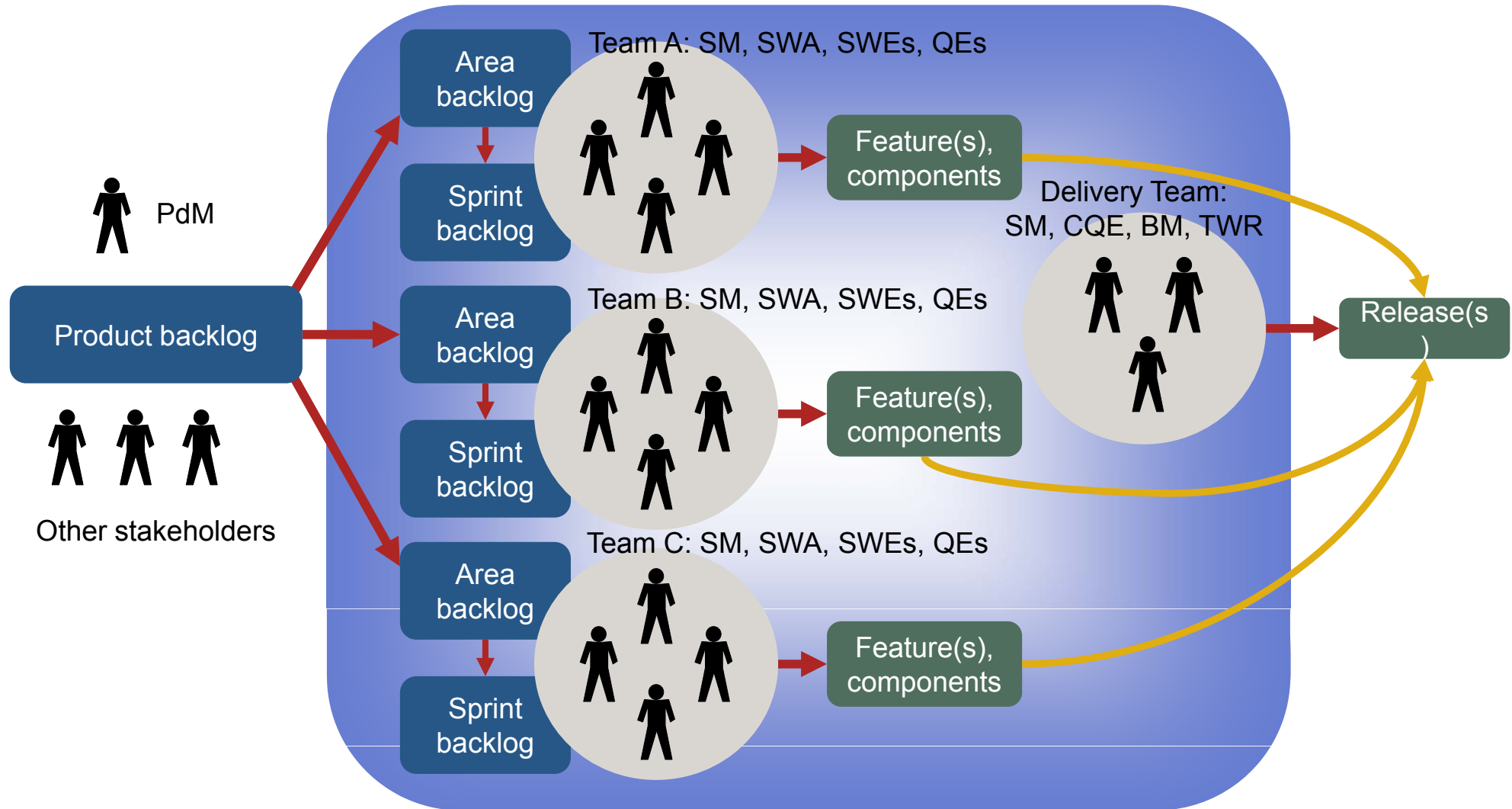
- Daily scrums within the team & weekly meetings with all stakeholders
- Daily/frequent CCBs (Change Control Board)
- Web-based collaboration and information sharing (Wiki, Confluence, XPlanner, etc.)
- Continuous integration and test automation
- Exploratory testing
- Early research and prototyping
- Knowledge transfer sessions for the whole company
- Reserve Brain for meetings 😊

Large projects – Our experiences



1. Don't try to start with multiple parallel teams at once
2. First: develop a feature that cuts through the whole architecture, not just a piece of the whole design.
3. Product backlog must take into account dependencies between teams and backlog items.
4. Divide the system to multiple related applications.
5. Identify *stable* reusable items for the application teams and create a small scrum team around that.
6. MetaScrum (Scrum of Scrums) meeting, once or twice a week

Multi-site/team Development Experience



Challenges to overcome



- Enhancing innovation, innovation management
- Culture change still needs continuous support and coaching (new roles,...)
- Set-up cross functional teams (feature teams)
- Improve continuous integration
 - Having a shippable release after each iteration in all programs and teams
- Multi-site, large projects (multi-projects and multi-products) synchronisation
- Tool support /especially product and project management in large, but agile projects

Current Process Improvement backlog

Top items

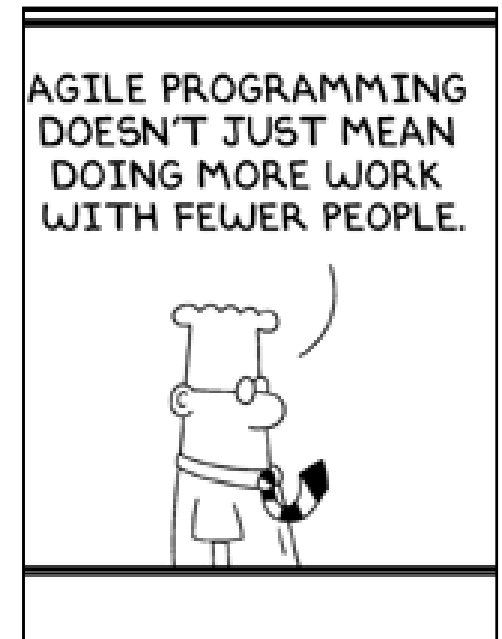


- Describe how to synchronize multi-site, parallel and dependent project teams
- Establish coaching for the method, named mentors for running F-LEX projects
- Define sub processes in F-LEX and create backlog items of what should be modified, deleted or new ones created (compared to FPRP)
- Check list for planning day
- Guidance on how to deal with CTS cases in sprint backlogs
- Clarify and communicate the roles of PjM, PdM, PgM, PSG, ...
- Train and communicate F-FIX (maintenance process)
- Arrange project learning days to collectively process retrospective feedback

Key Learnings



1. Agile is a cultural transformation and revolution, but what is Agile culture?
2. There is no single agile software development process to be taken into use off-the-shelf. Unique product life cycle process F-LEX as an example.
3. There are several different ways to adopt and deploy Agile methods.
4. Agile methods are far from ready: There are lot's of enhancement areas in Agile methods



**BE
SURE.**

