SoberIT
Software Business and Engineering Institute

SEMS

Combining Agile Methods

Kristian Rautiainen
SoberIT
http://www.soberit.hut.fi/sems/

SoberIT
Software Business and Engineering Institute

SEMS

Presentation Outline

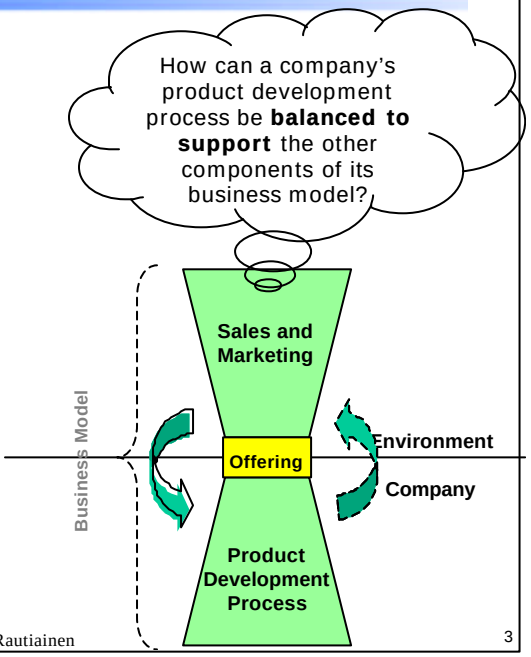
- Introduction
- SEMS *Cycles of Control* Framework
- Using the Cycles of Control in Choosing and Combining Agile Methods

SoberIT
Software Business and Engineering Institute

SEMS

Background and Context

- SEMS research project
  - Partners are small software product companies
    - Striving for standard products with minimum customer tailoring
  - The companies operate in turbulent environments
    - New technologies
    - Immature markets
  - There is a need to better manage and coordinate the different parts of the business model
    - Linking business and product development



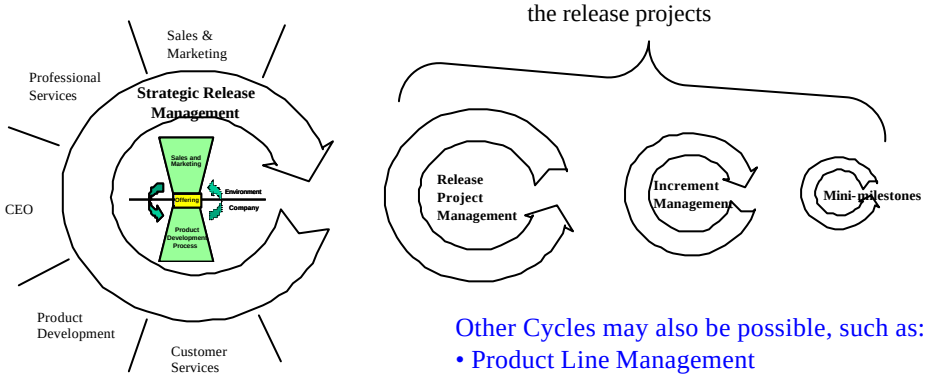
SoberIT
Software Business and Engineering Institute

SEMS

Characteristics of Software Product Business

- Based on releases of different types
  - Major and minor product versions
  - Maintenance and bug fix releases
- Sales and Marketing have to be well coordinated with Product Development
  - Risk of over or under selling
    - Sales sells something Product Development cannot deliver
    - Sales starts after product is ready
- Product quality must be good (enough)
  - The few resources may be tied up in maintenance work
- In immature markets the company must be able to react to and utilize changes
  - The product development process needs to be flexible and controlled

SEMS *Cycles of Control*



Other Cycles may also be possible, such as:
- Product Line Management
- Product Architecture Management
- Multi-project management
- …

Strategic Release Management Decisions

- Product mix
  - The product offering portfolio of the company
- Release types
  - Major, minor, maintenance, …
- Release content
  - Product vision to guide releases
  - Requirements for releases
- Release timing/pacing
  - Planned release dates/rhythm for the different types of releases
- Development model
  - How the releases are built
  - Pacing and phasing of the development work
- Organisation, roles and responsibilities
  - Deciding on decisions
  - Outsourcing vs. internal work
  - Competences and resource availability
    - Training and recruiting
- Technology and architecture
  - Technology used and investigated
  - Product architecture
  - Development infrastructure
- Quality assurance
  - Testing strategy
  - Release criteria
- Risk management

SoberIT
Software Business and Engineering Institute

SEMS

Release Project Management

- Input
  - Product vision for release
  - Product requirements for release
  - Release criteria
  - Release date
- Release planning
  - Requirements refinement
  - Effort estimation
  - Technical risks
  - Possible renegotiation of release requirements
  - Phasing the development into increments
- Output to customers
  - Ready product version
  - Acceptance or release criteria met
  - All accompanying product documentation ready
  - Other product release tasks done
- Output to Increment Mgmt
  - Release goal
  - Increment plan
    - Order and priority of requirements
    - Increment themes
      - Architecture
      - Testing
      - ...

SoberIT
Software Business and Engineering Institute

SEMS

Increment Mgmt and Mini-milestones

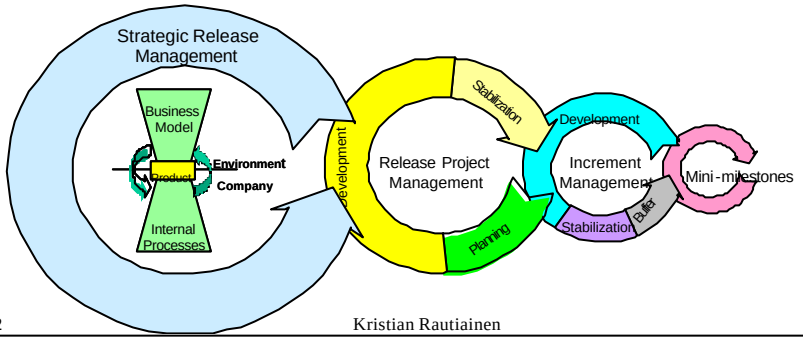
- Increment Mgmt
  - Increment goal set
  - Requirements broken down to tasks
  - Tasks allocated to people
- Output to Release Project
  - Ready product increment that is fully integrated to the product
  - At least the highest priority system tests passed
  - Some or all product documentation up-to-date
- Mini-milestones
  - Synchronising the individual efforts
    - For example the daily build, daily test cycle at Microsoft
- Early warning signs of trouble
  - Mini-milestones show progress of work
    - If work is not progressing it can be spotted right away
    - Corrective actions and evaluation of increment goal volatility
  - After each increment the same is applied to release goals

SoberIT
Software Business and Engineering Institute

SEMS

Product Development Management System

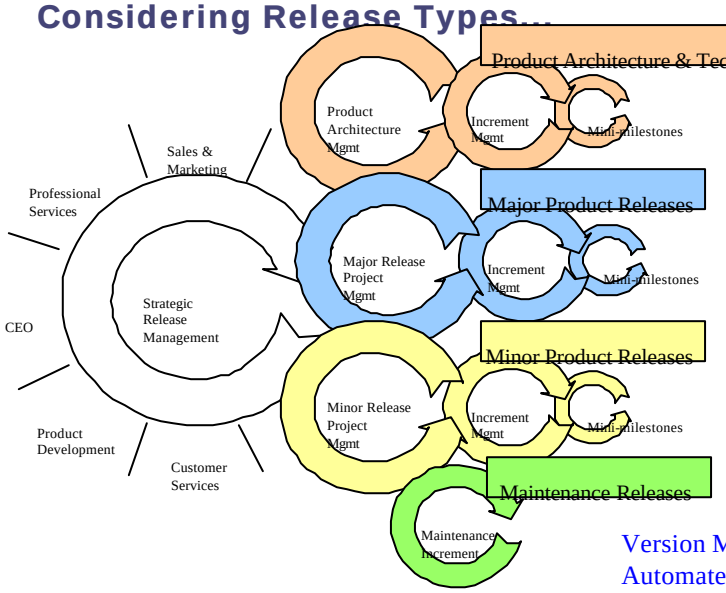
- Is created by combining and specifying the cycles
  - The length and number of the cycles
  - The software engineering activities and practices within and between the cycles
    - Here ideas and principles from Agile Methods can be used
  - For example: Microsoft's Synch-and-Stabilize (one release)



SoberIT
Software Business and Engineering Institute

SEMS

Considering Release Types...



Implementation details vary between different types of projects

- Roles and resourcing
- # and duration of cycles
- Communication patterns
- Decision-making rights
- ...

Version Mgmt challenging, Automated testing important, Concurrency issues, ...

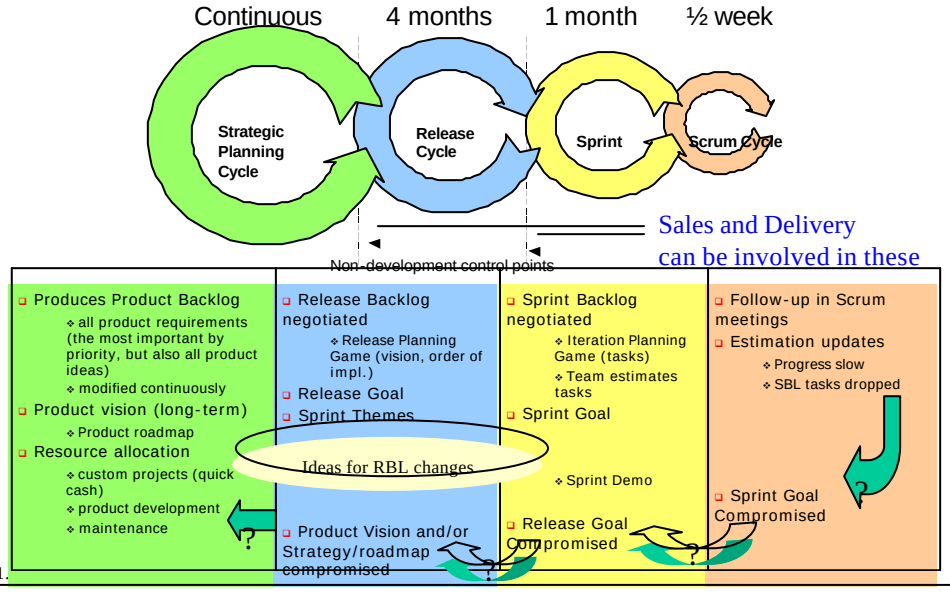
SoberIT
Software Business and Engineering Institute

Choosing and Combining Agile Methods

- Pekka told in his presentation about the deployment and suitability of different Agile methods
- Combining different Agile methods is possible, but some risks are involved
  - The practices in each method are carefully planned to support and utilise each other's strengths and weaknesses
  - To combine a different set of practices requires good understanding of the underlying implications
- Worst case:
  - An organisation throws away all practices that seem "boring" and only chooses practices to fit a "hacking" approach
- Good news
  - Agile methods address different aspects of software engineering
    - Project management
    - Development work
    - ...
  - For example Scrum and XP have been proposed to be combined, so that project management would use Scrum practices and development would use XP practices

SoberIT
Software Business and Engineering Institute

Example: (From Requirements Perspective)



SoberIT
Software Business and Engineering Institute

SEMS

Development Work and Quality Assurance

- Frequent synchronisation of development work
  - Early and frequent integration of the code base
    - Configuration mgmt!
    - Automated unit tests to catch bugs in regression
    - Examples:
      - XP practices
      - MS Synch-and-Stabilize Daily Build cycle
- Early defect detection by other means than testing
  - Peer reviews
    - Pair Programming
    - Review by Chief Architect
- Testing should start as early as possible
  - Key point of several Agile methods
    - Not too much detailed guidance given, though
  - Applies to all levels of testing
    - Unit, integration, system, acceptance

SoberIT
Software Business and Engineering Institute

SEMS

Summary

- Agile methods and/or practices can be combined
  - If the underlying implications are understood
  - Hint: start by piloting the one that seems to be the best fit, learn and improve/modify
- Understand your business and the needs and constraints it imposes upon your product development and vice versa
  - A balance should be found
  - Identify and utilise the strengths of your organisation
- The SEMS Cycles of Control can be used as a framework to ease communication about how product development could be organised
  - It is still tentative and needs refinement and detail
  - Initial experiences are encouraging