From waterfall to Agile, Case NPM

Petri Taavila, Senior Program Manager

COO/OBS/SM

Nokia Siemens Networks

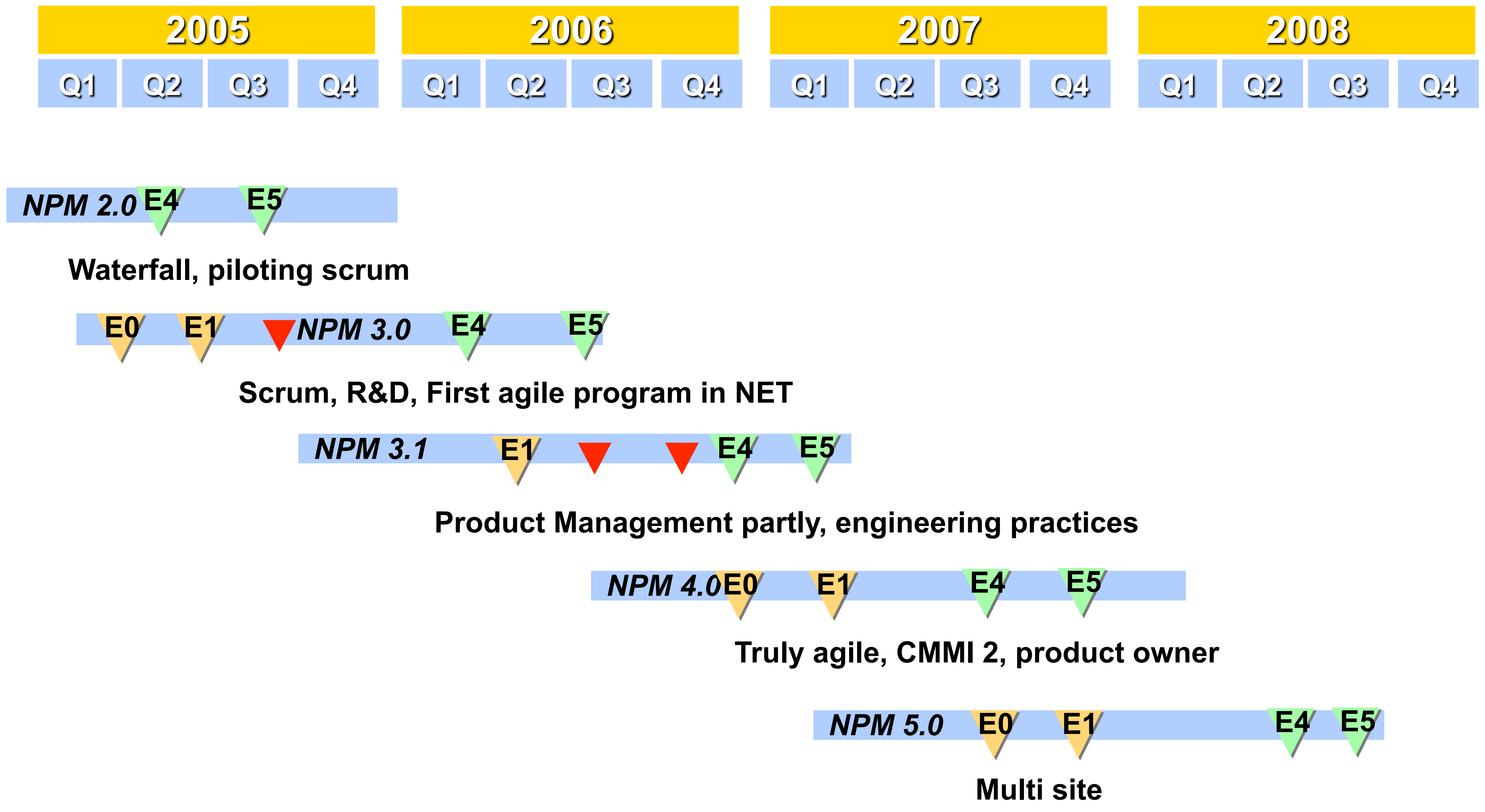
Objective & Agenda

- The objective of this presentation is to go through the change process and motivation from waterfall to agile using NPM program as an example.
- History of NPM programs
  - Timeline
  - NPM2.0, 3.0, 3.1, 4.0, **5.0**
- Conclusions
  - Current situation
  - Factors for success
  - Pain points
  - Agile myths

NPM program's eco-system as today

- Provisioning machine of NSN
- Two sites Finland & India
- Sw-development
- >1M lines of code
- Technologies
  - J2EE
  - Oracle
  - Linux
  - Standard IBM HW
- Scrum as a project Framework

Time line of NPM programs



2005
2006
2007
2008
Q1
Q2
Q3
Q4
NPM 2.0
E4
E5
Waterfall, piloting scrum
E0
E1
NPM 3.0
E4
E5
Scrum, R&D, First agile program in NET
NPM 3.1
E1
E4
E5
Product Management partly, engineering practices
NPM 4.0
E0
E1
E4
E5
Truly agile, CMMI 2, product owner
NPM 5.0
E0
E1
E4
E5
Multi site

NPM2.0 program

- Lasted 3 years
  - E4 was late 1,5 years
- Incremental process using Waterfall inside increment, 3 increments
- Agile seminar by Craig Larman March/2005
- Management commitment was given for change
- one team piloted scrum

•A LOT of Problems, see next slide

Motivation

- Problems that triggered the change (NPM2 program)
  - Huge bureaucracy (handover by documents)
  - Poor visibility to progress (late by 1,5 year)
  - No flexibility of content (fixed at E1)
  - R&D driven (Product management not very active)
  - Work load in peaks (specification, design, coding, test)

NPM3 program

- Which process to choose?
- Pilot Customer challenges
  - Deal Jul/2005
  - First customer committed delivery expected Sep/2005!!!!
- 6 Scrum teams
- Waterfall inside sprints
  - First sprint was disaster!!
- 80% of the content changed after program start
- First “agile” program in NET

NPM3.1

- After NPM3.0 success story some slippages from agility
  - Daily meetings twice a week
  - No proper planning session
  - No retrospectives
  - etc
- R&D quite agile
- Product management involved (only partially)
  - Taking new roles takes time
- Some engineering practices in place

NPM4.0 program

- E5 12/2007
- First truly agile program (In NPM history)
- First agile program in NSN achieved CMMI level 2
- Product management fully involved
- Engineering practices in good shape
  - TDD, continuous integration, test automation, pair programming, Planning poker, coding **dojos**, etc…
- Painful experiences if basic things not used

Nokia Siemens Networks

NPM5.0

- E4 : December,2008
  - Exactly on schedule in spite of major turbulences, with all content planned.
  - Major change in product direction, lots of new functionality.
- Continuous Release Planning
  - Multi-site synchronized sprints.
- Fully Automated Acceptance test suite.
  - Continuous Integration part of daily work.
- Lots of functionality which were never used in “production use” was removed.
  - Code base size was almost halved.
  - Build time for the whole product reduced dramatically by optimizing the build mechanism.
- Multi-site working practices established.
- “Scrum practices” improved even further.

Current situation (original problem)

- greater visibility to progress (poor visibility)
- reduced bureaucracy (huge bureaucracy)
- product management is participating actively (R&D driven)
- welcomes changes to the content (fixed content)
- work load evenly shared along the whole program (peaks in workload)

- **Bonus**
- increased job satisfaction
- delivery capability almost every day

Key factors to success

- Management commitment
- Key people commitment (Program manager, SW-project manager, test PM & Product Owner)
- Reserve enough time
  - Minimum 2 years
  - Introduce new practices gradually
- Use facilitators to introduce new practices
  - Help is available
- Have problem to be solved
- Train people
  - Whole organization
- Tolerate the resistance, be persistence
  - First sprint(s) will fail <- moment of truth
- Have good engineering practices in place
  - Also if waterfall is in use
- Lean thinking (challenge everything)
  - Also if waterfall is in use

Pain points

- Role of the solid line managers
  - There is no role in scrum!
- Self organizations scrum teams
  - People not used to take responsible
- New practices (Reviews, planning sessions, etc)
  - Facilitators needed, some agile experiences needed
- People are feeling that we are cheating by not doing so much
- Loud resistance by 10% of people
  - One by one “missionary work”
- Exposes the organization problems
- Exposes the individual performance
- Product Management role as a product owner
- Role of Program Contract = commitment in the early phase
- Management expectations
- Subcontracting

Common myths about Agile

- No documentation produced
  - Only needed documentation is done
  - Done together with implementation
- No planning
  - Very active planning & re-planning
- Don't scale
  - Successful projects exists
- Not suitable for multi site
  - Needs modular architecture
  - Either is waterfall
- Teams are allow to do what ever they want
  - In some sense yes, but there is the program point of view also
- Silver bullet to almost everything
  - Nope, but for many cases yes

Nokia Siemens Networks