Kill-1: process refactoring in the PyPy project

Europython 2006, CERN, Switzerland
Beatrice Düring (bea@changemaker.nu)

http://pypy.org/
http://codespeak.net/pypy

PyPy: project facts 1/2

- A F/OSS community within the Python community (350 subscribers,150 000 LOC)
- A consortium of 12 partners managing a fixed prize contract with the European Union
- 3 objectives – mainly to produce a fast and flexible Python implementation written in Python

PyPy: project facts 2/2

- Work divided into 14 work packages and 58 deliverables (code, reports, tools, “work”)
- Work divided into 3 phases over 2 years (1/12 2004-30/11 2006)
- Sprint driven development is used: the project meets and sprints ca every 6th week

Organizational structure year 1

- Formal structure of consortium work/coordination was centered around the management board and technical board
- Regular consortium meetings (monthly) mostly on IRC – coordinating consortium level work between partners
- Regular development meetings “sync-meetings” (IRC) – coordinating development work
- Tracking of time and costs among partners

Organizational structure year 2

- Still a technical board coordinating work (more people added though), still “sync-meetings”
- Refactored the management board into a more agile structure: agile management team
- “Identify issue, create suitable team, prepare the issue, recommend to the consortium, decide, implement”
- Regular consortium meetings replaced by singular decision meetings

Why? Our reasoning then...

- Although the work in the later phases seemed more segmented there was no need for centralized, ongoing coordination on project level
- Year 1 structure started to feel “artificial” - not fitting the reality of the project in some senses
- The project process needed to be more be “quicker” in answering the “right” questions – a Just-in-Time approach

But really, why?

- My own personal reflections looking back
- Three driving factors besides the operative justifications:
  - The group factor/FIRO
  - The situational leadership factor/SLM
  - The agile vs plan-driven factor/Boehm

1. The group factor

- FIRO model shows the development stages of a group:
  - First phase: inclusion
    - Short “honeymoon” period
  - Second phase: control
    - Short “honeymoon” period
  - Third phase: affection
- By end year one the group gone through the phases and needed another style of structure

2. The situational leadership factor

- The leadership within a group should balanced - suiting the situational needs of the group
- 4 cathegories of development of a group performing a result
  - Directing (low competence/low commitment)
  - Coaching (some competence/low commitment)
  - Supporting (high competence/variable commitment)
  - Delegating (high competence/high commitment)

3. The agile vs plan-driven factor

“ Agile development requires agile organizations” **(**Barry Boehm, keynote XP 2006 Finland)

"Agile teams are characterized by self-organization and intense collaboration, within and across organizational boundaries" (Cockburn, Highsmith, 2001)

Summary of driving factors

- The group had oriented itself towards the project environment, had “conquered it” - thus creating a change to minimalize the “scaffolding”
- The leadership needed to change to fit the nature of the group – enforcing more of the self-organizing workstyle, decentralized, Just in Time management
- The change was *created* by the group, the need was visible, the developers had shown the way

Conclusion

- Do not overdesign the process, as with coding – requirements do change
- Create change based on the nature of work, as opposed to patching onto something that does not match the needs of reality anymore
- Be clear about the difference between project process and development process
- Involve your team in refactoring – the are key stakeholders (needs, wants?)

Conclusion