Defects and agility: localization issues in agile development projects

Defects And Agility: Localization Issues in Agile Development Projects

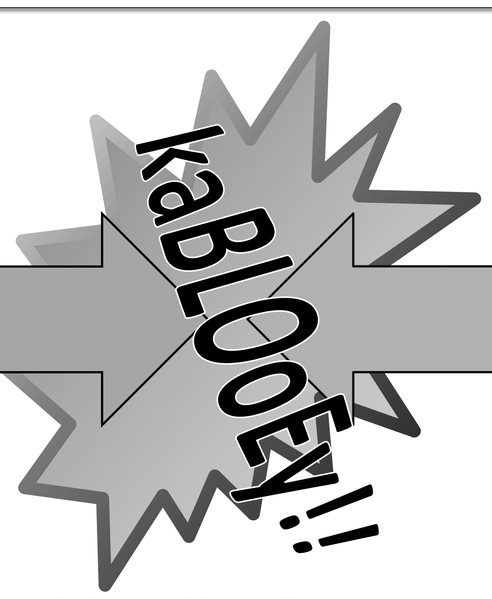
Malte Ressin, José Abdelnour-Nocera, Andy Smith

UNIVERSITY OF WEST LONDON

Software localization does not always fit well into agile software development. In this poster, we illustrate their relationship by examining how problems may occur. A list of common localization issues is presented, and their potential connections to the agile methodology are explored.

Agile Development

- Iterative
- Development
- Geared towards engineers
- Self-organized teams
- Emphasis on short-term planning
- Utilizing colocation



Localisation

- Linear
- Adaption
- Done by linguists, sociologists etc.
- Self-organized individuals
- Fixed duration
- Often outsourced

Friction between agile methodologies and localization:

Few/Lightweight processes
"Individuals and interactions over processes and tools" [1]

- No communication with localizers (if outsourced) [2, 3].
- No processes for localization needs and handovers [3].

Minimal documentation
"Working software over comprehensive documentation" [1]

- No documentation of localization requirements [2].
- Missing context information [2].

Iterative development
"Responding to change over following a plan" [1]

- Re-localization of previously localized, changed elements [3].
- Change of previously localized elements [3].

Bad localization quality
Text translations are misleading. Usability and user experience suffer from misleading adaptions.

Missing localizations
User interface elements are not adapted for certain languages, features don't work for some cultures, etc.

High localization efforts
Localization work was longer or more expensive than necessary.

Our research:

Research questions

- What factors have an impact on localization effort and quality?
- How do concepts, expectations regarding localization differ between translators and stakeholders?

Data sources

- Interviews: Gather stakeholder experiences.
- Surveys: Gain process information.
- Focus groups: Insights into issues.
- Case studies: Observe execution *in situ.*

Goals

- Create a comprehensive scientific model of localization in agile software development.
- Enable guidelines for facilitated localization though tools, process guidelines etc.

Preliminary results

- Communication between developers and translators.
- Automated content pipeline.
- Context information for translations.

References:

1. Beck, K. et al. (2001) *Manifesto for Agile Software Development.* [Online]. Available at: http://www.agilemanifesto.org/.
2. Carey, J. M. (1998) Creating global software: a conspectus and review. *Interacting with Computers,* 9, p. 449-465.
3. Turk, D. et al. (2002) Limitations of Agile Software Processes. *3rd Int. Conference on XP and Agile Processes in Software Engineering.*