Software localisation, the adaptation of software products for different languages, cultures and markets, is an important activity for international software. However, reports suggest that work of developers and translators does not mesh seamlessly, causing disproportionate cost, lack of quality, and delayed product release. Yet there is little research on localisation or its human factors. This research examines the causes of localisation issues by analysing qualitative data about the collaboration between development and translation. Semi-structured interviews with professionals in various roles were analysed towards a grounded theory of interdisciplinary collaboration in software localisation, explaining how collaboration strategies and conflicts reciprocally affect each other and are affected by external influences. Results suggest gaps in knowledge, procedure and motivation between developers and translators, as well as a lack of cross-disciplinary knowledge and coordination.

This research started as a puzzle from my work as localisation team leader in a mid-sized software company. I had observed that most localisation came from seemingly trivial causes nonetheless defying any attempts at proactive prevention. These could usually be phrased in the form of if-then-for example:

• If only software engineers finalised user interface (UI) text a month before product release, there would be no translation-caused release delay.
• If only UI designers remembered to leave at least 30% buffer space for translation- expanded text, there would be fewer instances of cut in the UI.
• If only translators referred to the terminology when translating, we would have fewer retranslations.

Process-related shortcomings in localisation practice have further been acknowledged in the literature:

• Lack of standard processes (Abufardeh and Magel, 2008)
• Incomplete understanding of localisation activities and workflow (Lenker et al., 2011)
• Issues of collaboration between software engineering and localisation (Abufardeh and Magel, 2010; Lewis et al., 2009).

Accordingly, there have been calls to examine the collaboration of software engineering and localisation (O’Sullivan, 2001; Collins, 2001). The conceptual model guiding initial research was based on the project management triangle of cost, time and quality.

What causes localisation issues?

Localisation issues are a result of the hierarchical relationship between developers and localisers. Developers enjoy a privileged position compared to localisers and their relationship with localisers can easily develop into a dysfunctional regime in which processes and tools exclusively cater for development. The more dysfunctional the relationship is, the less localisers request necessary information or warn about potential issues, and instead shift their work and activities towards their unique interests. Eventually, developers and localisers settle into a relationship in which localisation is superseded by alternative interests. Those goals of software localisation which are not among their priorities are compromised and cost, quality or schedule issues occur.

A grounded theory emerged bottom-up from processed, reduced and organised interview data. Below is an overview of the theory of interdisciplinary collaboration and main answers to the research questions. Circles signify categories, boxes signify high-level concepts, lines indicate relationships, and arrows indicate influences.

Acknowledgements

This research was supported by a grant from the Swedish Research Council for Information, Media and Communication (2009-3436). I would like to thank the interviewees and my supervisor, Malte Ressin, for their patience and valuable input. Any errors and omissions are entirely my own.

References


Contact e-mail: malte.ressin@uwl.ac.uk