



UWL REPOSITORY

repository.uwl.ac.uk

Of code and context: collaboration between developers and translators

Ressin, Malte ORCID: <https://orcid.org/0000-0002-8411-6793>, Abdelnour-Nocera, Jose ORCID: <https://orcid.org/0000-0001-7935-7368> and Smith, Andy (2011) Of code and context: collaboration between developers and translators. In: 33rd International Conference on Software Engineering (ICSE 2011), 21-28 May 2011, Honolulu, USA. (Unpublished)

This is the Accepted Version of the final output.

UWL repository link: <https://repository.uwl.ac.uk/id/eprint/3574/>

Alternative formats: If you require this document in an alternative format, please contact: open.research@uwl.ac.uk

Copyright:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

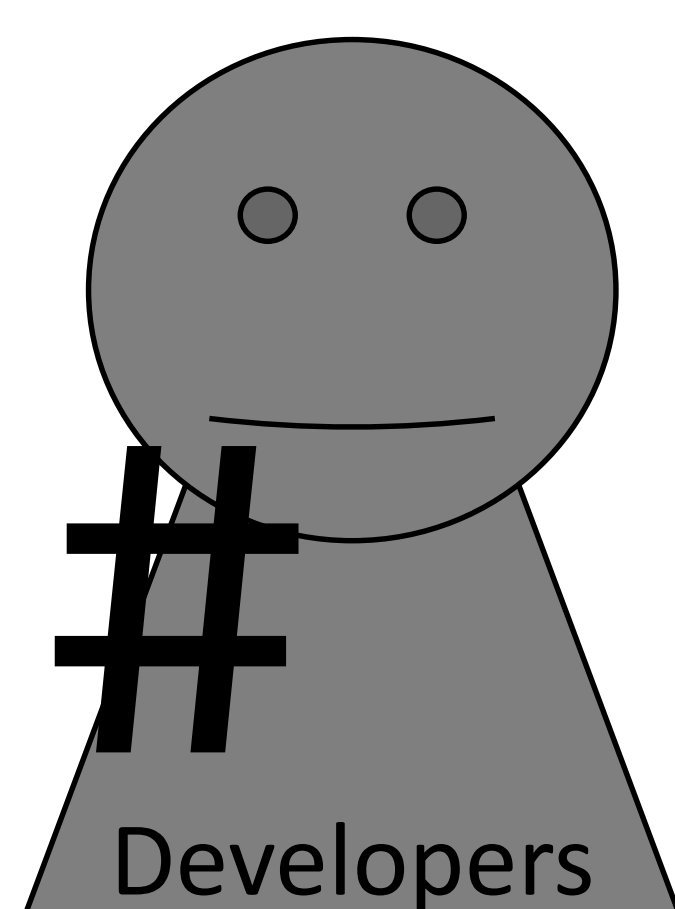
Take down policy: If you believe that this document breaches copyright, please contact us at open.research@uwl.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Of Code and Context: Collaboration between Developers and Translators

Malte Ressin, José Abdelnour-Nocera, Andy Smith

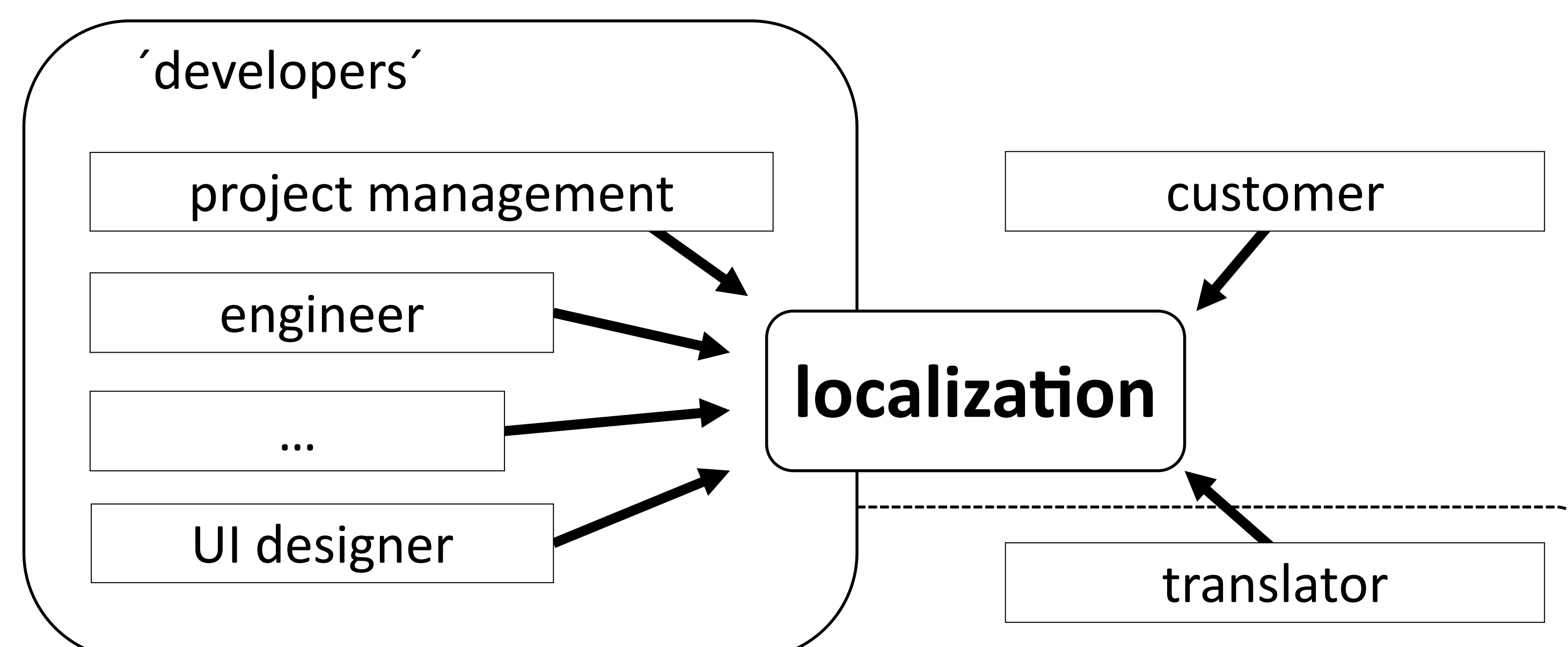
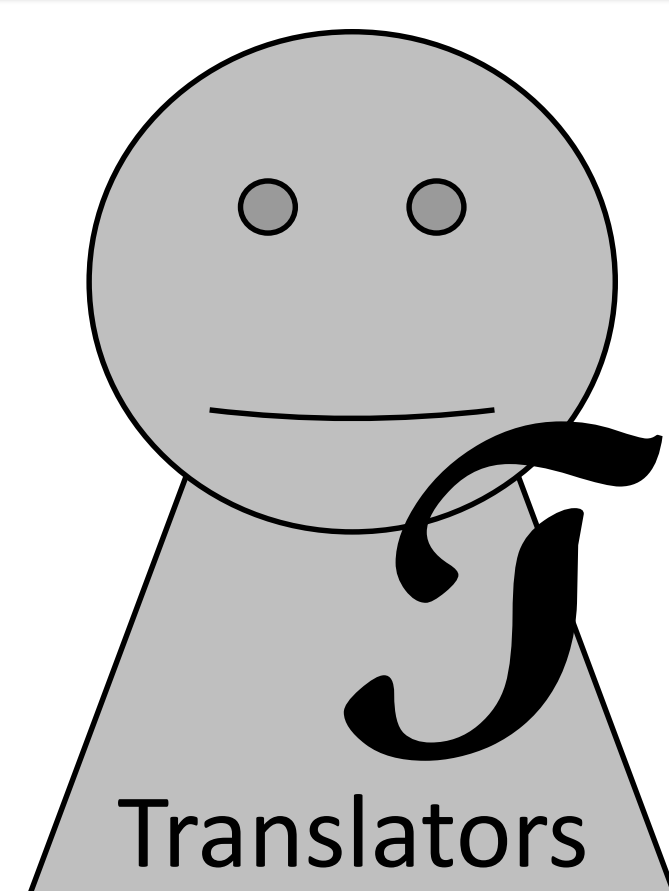
Software for international markets often requires cultural adaption, done by developers and translators internationalizing and localizing the product. This poster contrasts differences in work, education and values, and illustrates other important factors with an impact on collaboration. Results suggest an improvement of collaboration by integrating translators into development, and by understanding each other's work better.

Introduction:



- internationalize
- work iteratively
- develop
- practice teamwork
- are engineers
- utilize colocation

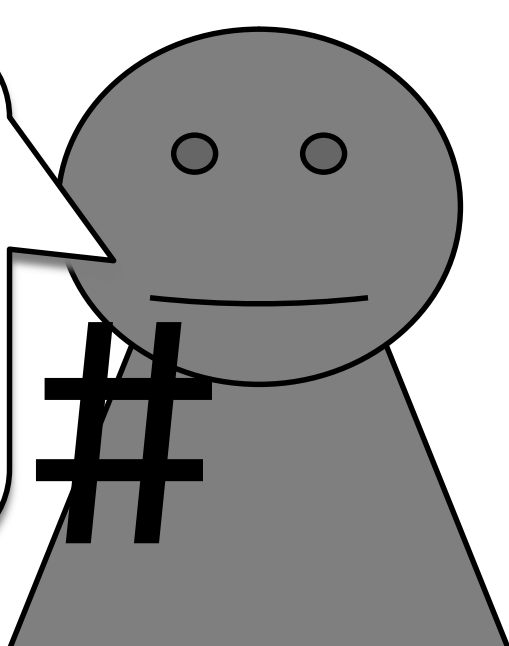
- internationalize and localize
- work linearly
- adapt
- work individually
- are linguists, sociologists etc.
- are often outsourced



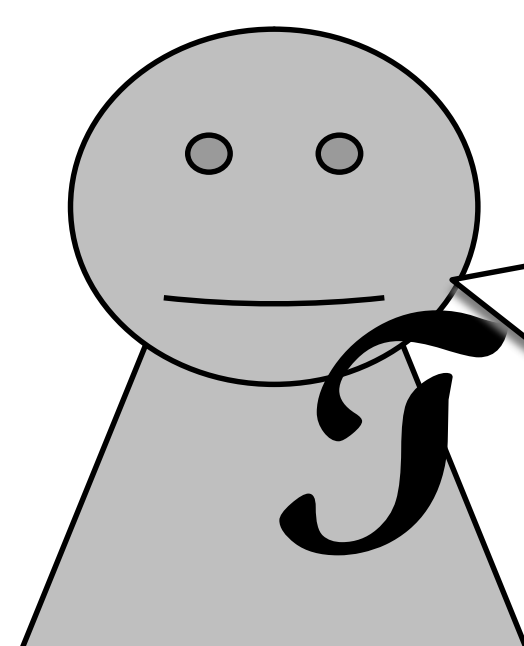
Software developers in this context contribute directly to the product. While this applies also to translators, they are usually outside of the development team [1].

Relationship to the product:

I work on a project, and you are a supplier.



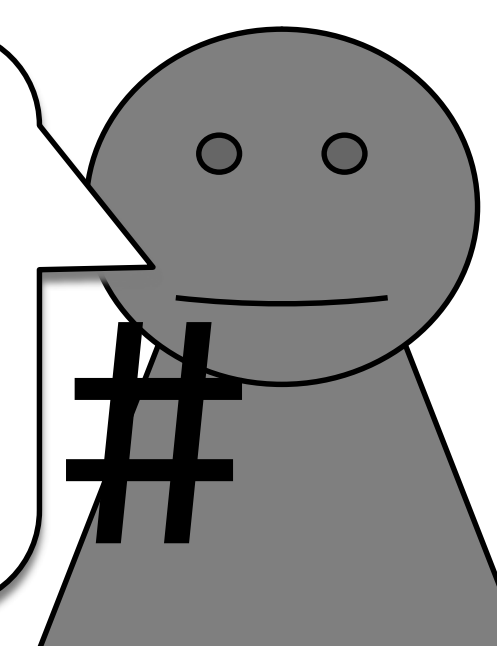
You are my customer.



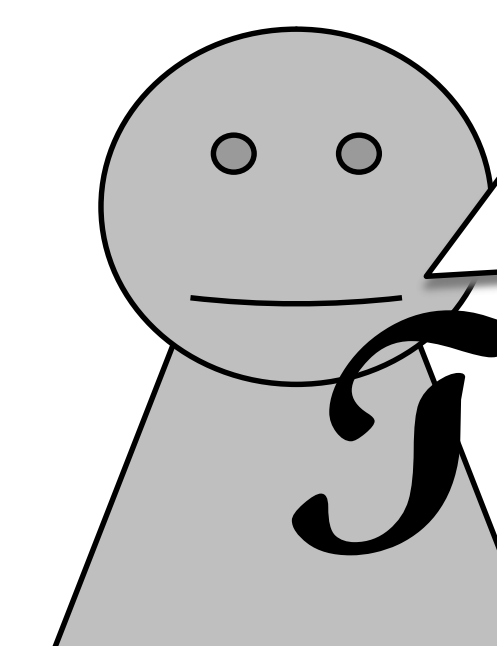
Developers are very close to the product and are paid for time. Translators have work only indirectly on the product, are often outsourced and are paid per word [2].

Concepts of Quality and Achievement:

I need *year* in Spanish, but the 15-element array can't display diacritics.



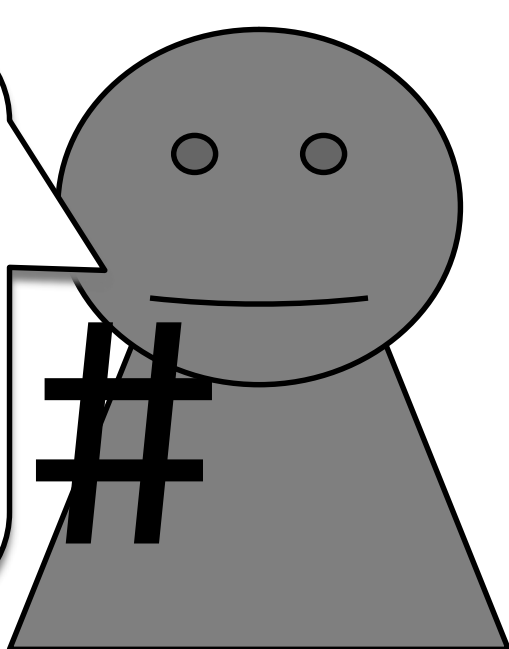
Año is perfect, but don't remove the ~, it would change meaning.



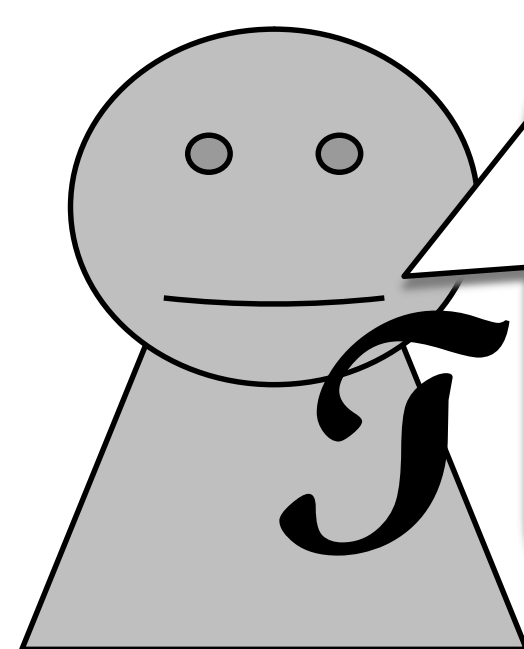
Developers have measurable quality metrics such as speed and completeness. Translation quality is much more subjective and difficult to quantify [3].

Different Work Processes:

This project is agile, we work in sprints.



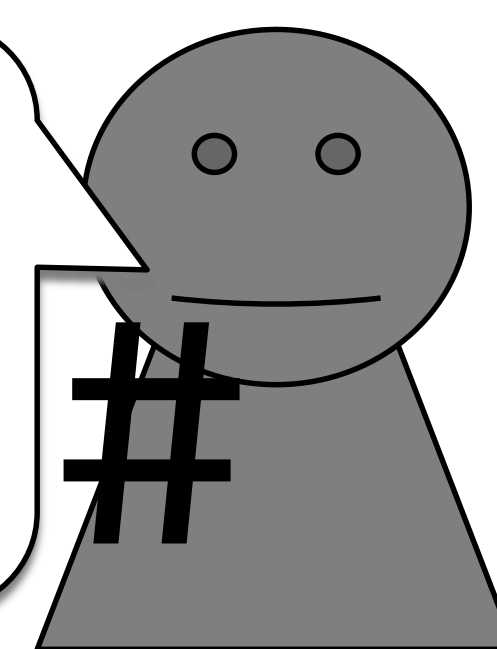
Ok, but I need a wordcount ahead of time. Changes cost extra.



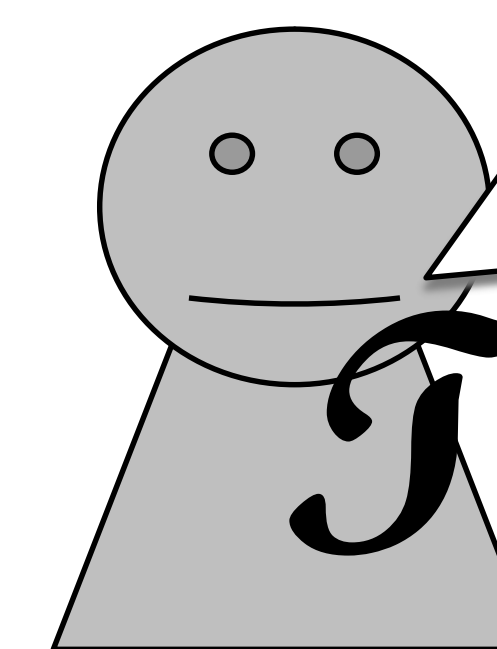
Work processes in software projects are geared highly towards engineers and reusability and extensibility [4]. Translation is linear work step dealing with static text and symbols.

Understanding Each Other's Subject:

I know languages, and translation is like a unique mapping.



I was taught that `%2d` must not be changed as it is a placeholder.



Developers' specialist knowledge is clearly outlined. Translation is done by many people do on a daily basis. As a result, translators' knowledge and skills are easily underappreciated.

Our Ongoing 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 through tools, process guidelines etc.

Preliminary results

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

References:

- [1] Esselink, B. *A practical guide to localization*. John Benjamins Pub. Co, 2000.
- [2] Ciarlone, L. *Evolving global product content practices*, MultiLingual, 20 (4), 50-52.
- [3] Müller, E. *Building Quality into the Localization Process*. MultiLingual Localization Guide, 14-15, 2000.
- [4] Lindvall, M. & Rus, I. *Process Diversity in Software Development*. IEEE Software, 17 (4), 14-18, 2000.