Let the EAGLES Fly into New Standards: Adapting our CAT Tool Evaluation Methodology to the ISO 25000 Series.

Starlander, Marianne
University of Geneva, Translation and Interpretation
Faculty, Translation technology Department (TIM)
Marianne.starlander@unige.ch

Abstract

This paper is a follow up to our teaching case study described in ASLIB 2013. The subject of the present paper is how do we integrate the new ISO 25000 series (ISO/IEC 2014) to update the EAGLES 7-steps recipe, which is one of the deliverables of the Evaluation of Natural Language Processing Systems project (EAGLES I and II) based on the ISO 9216 software evaluation series. The present poster paper will focus on the methodology proposed to the students and give some preliminary results in order to give a flavor of the achieved work within only several weeks of our MA course. The main aim of this paper is thus to provide a ready-made methodology to evaluate CAT tools, that can be reused not only in the academic field by contributing to include such knowledge into "basic" translator's training but also by freelancers willing to evaluate several tools before making their choice.

1 Introduction

The subject of the present paper is the integration of the new ISO 25000 series (ISO/IEC 2011, ISO/IEC 2014) to update the EAGLES 7-steps recipe[1], which is one of the deliverables of the Evaluation of Natural Language Processing Systems project (EAGLES I and II) dating back to the 90's, based on the ISO 9126 software evaluation series.

As mentioned in Starlander and Morado Vazquez (2013), the main objective of the methodology taught within the Computer Assisted Translation (CAT) MA course at the Faculty of Translation and Interpreting of the University of Geneva, is to provide our students with a functional evaluation methodology and the necessary knowledge to fulfil a task that they often have to face at the start of their carrier as a translator, or hence as freshly baked CAT tool "experts".

The given assignment did not change radically from what was described in previous work. What is new in the present case study is that the students need to move away from the "classic EAGLES 7-steps" through the integration of the new quality characteristics contained in the ISO 25000 standards series. The main changes in the latter compared to the ISO/IEC 9126 series is the clarification of terminology used (Abran et al, 2005) and the set of quality characteristics (Abran et al, 2007).

It must be noted that although the "ability to evaluate the suitability of a tool in relation to technical needs and price" was identified by Pym (2012) as one of the necessary skills that translation students should acquire, this skill is not yet usually included into classical translation's training, not even during CAT tool classes.

The proposed methodology is based on a yet another simplification of the EAGLES methodology while integrating a quality model based on the new ISO 25000 series (ISO/IEC 2014), in order to make it accessible to MA students but also to freelance translators or more generally language professionals using CAT tools.

[1] EAGLES Evaluation Working Group (1999): The EAGLES 7-step recipe, available at http://www.issco.unige.ch/en/research/projects/eagles/ewg99/7steps.html

Proceedings of the 37th Conference Translating and the Computer, pages 96–101,
London, UK, November 26-27, 2015.

The advantage of the ISO standards is that they offer a general framework to software evaluation which needs to be adapted and interpreted according to each evaluators needs. The aim is to standardise the evaluation practices. So far these are rather of an ad hoc nature, not generalizable or replicable to other evaluations which forces the evaluators to start all other again for each new evaluation. In the following part of the introduction we will briefly explain the EAGLES 7-steps recipe and compare the ISO/IESC 25000 series to the well-known 9216 series originally used in EAGLES. Then, in section 2, we will describe in more details the methodology we invite our students to use for their assignments and in their future work. In section 3, we will give some preliminary results of how the students applied the method. In section 4 we will conclude and discuss this experiment.

1.1 EAGLES

The aim of the Expert Advisory Group on Language Engineering Standards (EAGLES)[2] was to adapt the relevant ISO standards (ISO/IEC 9126-1 1991 and ISO 14598 1998) to the translation environment and to create a flexible and modifiable evaluation framework using a hierarchical classification of features and attributes (Quah 2006: 142). Their work has resulted in concrete examples for spell-checkers (appendix D of (EAGLES 1996)) but also for CAT tools as terminology extractors and Translation Memory Systems (TMS) (appendix E of (EAGLES 1996)). This work has also widely influenced the ISO/IEC 9126 (ISO/IEC 2001) standards and has resulted in a shortened and simplified *seven steps recipe*[3]. This recipe focuses on the importance of the context of use and gives seven clear steps to achieve an objective evaluation. The aim is to guide the evaluator in the jungle of the quality characteristics in order to determine which are important for the specific context of use. The original EAGLES recipe integrates mainly the external quality characteristics of the ISO/IEC 9126 series.

Since 2007-2014 a new set of series has appeared that is to replace the 9126 series, this is why we decided to adapt EAGLES to this new set and also to add a focus on quality in use we therefore concentrate on these new characteristics that we will now describe in the following section.

1.2 ISO/IEC 25000 Series

The new ISO 25000 series Software Product Quality Requirements and Evaluation (SQuaRE) (ISO/IEC 25000:2014) are equivalent to the ISO/IEC 9126 series and ISO/IEC 14589 series. The object of the new series is the evaluation of software defined as follows "systematic examination of the extent to which a software product is capable of satisfying stated and implied needs[4]".

As in the series represented in the original EAGLES series 9126 1-4, SQuaRE is divided into several norms: the Quality Model Division (ISO/IEC 2501n) "presents detailed quality models for computer systems and software products, **quality in use**, and data[5]".

ISO/IEC 2502n – **Quality Measurement Division** includes "a software product quality measurement reference model, mathematical definitions of quality measures, and practical guidance for their application[6]", which is equivalent to ISO/IEC 9126-2:2003. It also provides examples of internal and external measures for software quality (cf. ISO/IEC 9126-2, appendix A-C), and measures for quality in use, which is equivalent to 9126-4. The new series is based on the concept of "Quality Measure Elements" (QME) that form the foundations for these

[2] EAGLES Group Site, http://www.issco.unige.ch/en/research/projects/eagles/, accessed on the 30.07.2015.
[3] EAGLES Final Report Site, presenting the seven steps recipe TAL, http://www.issco.unige.ch/en/research/projects/eagles/ewg99/7steps.html, accessed on the 02.08.2015.
[4] Source: ISO/IEC 25000:2014, p.6.
[5] Source: ISO/IEC 25000:2014, p.8.
[6] Source: ISO/IEC 25000:2014, p.8.

measures. Furthermore, what was divided into internal and external quality models (ISO/IEC 9126-1 and ISO/IEC 9126-2) has been combined into a single product quality model[7].

From this very short overview it comes clear that the scope of the quality models have "been extended to include computer systems, and quality in use from a system perspective"[8]. This implies a more comprehensive point of view. Apart from this major change, the set of characteristics and sub-characteristics has changed (cf. Table in Appendix 1), two of the main characteristics remain unchanged: effectiveness and satisfaction, while as the latter has now four sub-characteristics. What used to be called *productivity* is now labeled *efficiency* and finally the fourth main sub-characteristic *safety* has been changed to *freedom from risk*, divided into six sub-characteristics that have been given more accurate names A fifth characteristic has been added: *context coverage* decomposed into *context completeness* and *flexibility* (cf. Table in Appendix 1).

The major change compared to Starlander and Morado Vazquez (2013) is that we moved entirely to the **quality in use** characteristics. The definition of quality in use is the "degree to which a product or system can be used by specific users to meet their needs to achieve specific goals with effectiveness, efficiency, freedom from risk and satisfaction in specific contexts of use" (ISO/IEC 2500:2014). This thus differs from the original EAGLES recipe, since the characteristics included there (EAGLES, 1999) are drawn from the ISO/IEC 9126-2 (2003), and therefore based on the set of the six following main characteristics (functionality, reliability, usability, efficiency, maintainability and portability).

We will not describe each characteristic further for space restriction but rather concentrate on how we integrated the five main characteristics (effectiveness, efficiency, satisfaction, freedom from risk and context coverage) and sub-characteristics into EAGLES 25000.

2 The EAGLES 25000 Methodology: the 7-Steps Revisited

Our approach is based on the context of use defined as follows in (ISO 25010:2011): "context of use: users, tasks, equipment (hardware, software and materials), and the physical and social environments in which a product is used" which is identical to the definition given in ISO 9241- 1. This implies live tests in real environment use. Although we are in an academic context, we try to recreate possible professional scenarios. We therefore ask our students to choose between a range of contexts of use (similar to Starlander and Vazquez (2013)):

1. Novice freelance-translator
2. Experienced freelance- translator with a lot of previous translations to be recycled into a TMS.
3. Experienced in-house translator, working in a company were so far a particular TMS has been used but the decision has been taken to potentially move to another TMS.
4. Head of translation support unit of an international organisation, you need to introduce a TMS that suits best the given work environment.

Once they have chosen their scenario, the students have to follow the 7-steps recipe, where they will determine a set of quality in use characteristics during step 3 and 4.

[7] Source: ISO/IEC 25000:2011, p.v.
[8] Source: ISO/IEC 25000:2011, p.1.

Step #	Description of the seven steps (EAGLES 25000)
1	**Define the aim of the evaluation**: What exactly is being evaluated? Is it a system or a system component? In which a specific context of use (Scenario1-4)?
2	**Elaborate a task model**: What is the system going to be used for? Who will use it? What will the users do with it? What is the user profile?
3	**Define top level quality characteristics:** What characteristics (effectiveness, efficiency, satisfaction, freedom of risk and context coverage) of the system need to be evaluated? Are they all equally important according to the context of use?
4	**Produce detailed requirements for the system under evaluation:** Choose the appropriate characteristics and sub-characteristics (Cf. Table in Appendix 1). The quality model should end-up with measurable features.
5	**Devise the metrics to be applied to the system according to quality model chosen:** How will the chosen characteristics be measured. Define the applied measure but also for each measurable attribute, define the interpretation scale.
6	**Design the execution of the evaluation:** Develop test materials to support the testing of the object. Find the participants to the tests. What form will the end result take? Design a clear test protocol.
7	**Execute the evaluation:** Run tests and make measurements. Compare with the previously determined satisfaction ratings. Summarize the results in a concise evaluation report

Table 1: Description of the seven steps according to EAGLES 25000

As you can see from Table 1, we have adapted the original EAGLES 7-steps to the new ISO 25010:2011 quality in use characteristics and sub-characteristics for the students. This methodology is accompanied by a brief general introduction on software evaluation. Guidance is provided during the three weeks available for the assignment. The final product is both a concise written report and a 5-minute oral presentation.

3 Preliminary Results

Our students widely chose the first context to which they can better identify themselves with. Out of the 48 enrolled students this year (2015-16), a majority chose the first scenario, which was also the case in the previous years. What is new is the wider range of evaluated TM systems, with a consequently higher amount of cloud systems represented. During the explanation of the task and the description of the methodology students understood what we expected from them and from what we can observe from the preliminary working plans, the 7-steps recipe was well applied by the majority of them.

We are unfortunately not able at the time of writing the paper to provide the results of the current academic year since the students work is due for December 2015, but in the poster presented we will be able to give more details because the students will have handed in their detailed working plan.

4 Conclusion

We have presented in this poster paper a straightforward methodology adapted to our students' capacity and time available for the class that allows them to construct their comparative evaluation according to the latest ISO standards but leaving a certain space to freedom and personal thinking. The methodology implies indeed determining a tailor-made evaluation

according to the chosen scenario but also the functionalities of a system each group decided to focus on.

This methodology could be extended to a wider professional context. In fact, most alumni from previous CAT tool classes continue to use this methodology in their future career as recommended and also adapt it for their MA thesis (Gray, 20014, Walpen, 2011).

In future work, it would be interesting to study the feasibility of applying this methodology in a professional or industrial context. Is there enough time to adapt this methodology, or should a readymade version for each type of system be proposed to accelerate the process? This was also the aim of Celia Rico (2001), but so far the general evaluation practice in our field has not yet adopted such an evaluation readymade library of evaluation models. The question that arises here is: would it be possible to propose a large enough range of tailored evaluations, and would the impact of such a standardization only be positive?

References

Abran, A., Al-Qutaish, R. E., & Desharnais, J. M. (2005). Harmonization issues in the updating of the ISO standards on software product quality. *Metrics News*, *10*(2), 35-44.

Abran, Alain, et al. "ISO-based Models to Measure Software Product Quality." *Institute of Chartered Financial Analysts of India (ICFAI)-ICFAI Books* (2007).

EAGLES Evaluation Working Group (1999): The EAGLES 7-step recipe, available at http://www.issco.unige.ch/en/research/projects/eagles/ewg99/7steps.html

Gray, C. (2014). A comparative evaluation of localisation tools: Reverso localize and SYSTANLinks.

ISO/IEC (2001). ISO/IEC 9126-1:2001 Software engineering — Product quality — Part 1: Quality model

ISO/IEC (2003). ISO/IEC 9126-2:2003 (en) Software engineering - Product quality - Part 2: External metrics. Geneva, International Organization for Standardization / International Electrotechnical Commission: 86.

ISO/IEC (2011). ISO/IEC 25010:211(en) Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models

ISO/IEC (2014). ISO/IEC 25000:2014(en) Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE. Geneva, International Organization for Standardization / International Electrotechnical Commission.

Pym, A. (2012) 'Translation skill-sets in a machine-translation age', [online], available: http://usuaris.tinet.cat/apym/on-line/training/2012_competence_pym.pdf [accessed 6 Nov 2013].

Quah, C. K. (2006) *Translation and Technology*, Hampshire/New York: Palgrave. Macmillan.

Rico, C. (2001) 'Reproducible models for CAT tools evaluation: A user-oriented perspective', *Proceedings of the Twenty-third International Conference on Translating and the Computer*, London. Aslib.

Starlander, M. and L. Morado Vazquez (2013). Training translation students to evaluate CAT tools using Eagles: a case study. Aslib: Translating and the Computer 35. Londres, Aslib.

Walpen, N. (2011). Translation technology for the federal chancellery - the usefulness of a translation memory system for the German section of the central language services.

Appendix 1: Quality in use characteristics, sub characteristics and definitions (ISO 25010:2011)

Characteristics	Sub-Characteristics
Effectiveness: Accuracy and completeness with which users achieve specified goals	
Efficiency: Resources expended in relation to the accuracy and completeness with which users achieve (time to complete the task, materials, or the financial cost of usage.)	
Satisfaction: Degree to which user needs are satisfied when a product or system is used in a specified context of use	
	Usefulness: Degree to which a user is satisfied with their perceived achievement of pragmatic goals, including the results of use and the consequences of use
	Trust: Degree to which a user or other stakeholder has confidence that a product or system will behave as intended
	Pleasure: Degree to which a user obtains pleasure from fulfilling their personal needs
	Comfort: Degree to which the user is satisfied with physical comfort
Freedom from risk: Degree to which a product or system mitigates the potential risk to economic status, human life, health, or the environment	
	Economic risk mitigation: Degree to which a product or system mitigates the potential risk to financial status, efficient operation, commercial property, reputation or other resources in the intended contexts of use
	Health and safety risk mitigation Degree to which a product or system mitigates the potential risk to people in the intended contexts of use
	Environmental risk mitigation: Degree to which a product or system mitigates the potential risk to property or the environment in the intended contexts of use
Context coverage: Degree to which a product or system can be used with effectiveness, efficiency, freedom from risk and satisfaction in both specified contexts of use and in contexts beyond those initially explicitly identified	
	Context completeness: Degree to which a product or system can be used with effectiveness, efficiency, freedom from risk and satisfaction in all the specified contexts of use
	Flexibility: Degree to which a product or system can be used with effectiveness, efficiency, freedom from risk and satisfaction in contexts beyond those initially specified in the requirements (Flexibility can be achieved by adapting a product for additional user groups, tasks and cultures).