

# Evaluating Language Technologies: The MULTIDOC Approach to Taming the Knowledge Soup

Jörg Schütz and Rita Nübel

IAI  
Martin-Luther-Straße 14  
D-66111 Saarbrücken  
GERMANY  
email: {joerg,rita}@iai.uni-sb.de

**Abstract.** In this paper we report on ongoing verification and validation work within the MULTIDOC project. This project is situated in the field of multilingual automotive product documentation. One central task is the evaluation of existing off-the-shelf and research based language technology (LT) products and components for the purpose of supporting or even reorganising the documentation production chain along three diagnostic dimensions: the process proper, the documentation quality and the translatability of the process output. In this application scenario, LT components shall control and ensure that predefined quality criteria are applicable and measurable to the documentation end-product as well as to the information objects that form the basic building blocks of the end-product. In this scenario, multilinguality is of crucial importance. It shall be introduced or prepared, and maintained as early as possible in the documentation workflow to ensure a better and faster translation process. A prerequisite for the evaluation process is the thorough definition of these dimensions in terms of user quality requirements and LT developer quality requirements. In our approach, we define the output quality of the whole documentation process as the pivot where user requirements and developer requirements shall meet. For this, it turned out that a so-called "braided" diagnostic evaluation is very well suited to cover both views. Since no generally approved standards or even valid specifications for standards exist for the evaluation of LT products, we have adjusted existing standards for the evaluation of software products, in particular ISO 9001, ISO 9000-3, ISO/IEC 12119, ISO 9004 and ISO 9126. This is feasible because an LT product consists of a software part and a lingware part. The adaptation had to be accomplished for the latter part.

## 1 Introduction

MULTIDOC is a European project of the Fourth Framework Programme of the European Commission within the Language Engineering Sector. It is founded on the specific needs and requirements of product documentation expressed by several

representatives of the European automotive industry, among them are Bertone, BMW, Jaguar, Renault, Rolls-Royce Motor Cars, Rover, Volvo and others. The focus of the project is particularly on the multilingual aspects of product documentation. Therefore, the general goal is to define and specify methods, tools and workflows supporting stronger demands on quality, consistency and clarity in the technical information, and shorter lead times and reduced costs in the whole production value cycle of documentation including the translation into multiple languages. The results of the project, however, are applicable to any other component or system manufacturing business; thus, they are not restricted to the automotive industry.

The project is divided into two phases: an inception and elaboration phase, the so-called MULTIDOC Concerted Action, and a construction or development phase, the so-called MULTIDOC Project. The first phase has been finished by the end of 1997, and the second phase has started in January 1998.

Evaluation is a task or rather a continuous process that is maintained throughout all project phases, so that a strict user-orientedness is ensured. In the inception and elaboration phase, we assessed several language technology (LT) products and components for their deployment in supporting and enhancing the quality of technical documentation, and to control and streamline the translation process which in most cases is contracted out to translation agencies or translation companies. For this assessment, we defined three so-called diagnostic quality dimensions which form the basis of all verification and validation activities:

1. Process (workflow) quality requirements.
2. Product (documentation) quality requirements.
3. Multilinguality (translatability) quality requirements.

These diagnostic dimensions are iteratively further elaborated and refined during the construction phase. On the one hand, this ensures that we will have quantitatively and qualitatively measurable improvements of the documentation value chain based on the initially stated needs and their associated quality features and characteristics. And on the other hand, this will guide the further development and the adaptation of LT products and components for the task-specific application.

In the remainder of this paper we will describe the prerequisites and the different steps of our evaluation approach. After a brief overview of the main user quality requirements in MULTIDOC which we have identified within the specific domain of technical documentation of Service and Repair Methods (SRM), we elaborate our evaluation methodology and the adopted method and principles. The user requirements have primarily guided the choice of the functionalities of the LT components which will be described subsequently. In the MULTIDOC project, the purpose of evaluating LT components is not to ultimately decide which specific component should win over another. Rather, the evaluation shall result in quantitatively and qualitatively measurable improvements of the whole documentation value chain, and shall also guide the introduction of possible extensions, amendments and improvements to the LT products and components according to user needs and demands. The subsequent sections are dedicated to the discussion of the MULTIDOC evaluation principles (metrics and metric value scales)

and the design of the evaluation process. In the last section, we will summarise our findings and draw some further conclusions.

## 2 Quality Requirements Analysis

Within the MULTIDOC application scenario, we distinguish two types of users:

1. Technical writers as the producers of technical information for automotive service and repair.
2. Technicians and mechanics in automotive workshops as the consumers of technical information in their day-to-day operations.

Both groups have different quality requirements on the technical documentation, and in particular on the different information objects which form the basic building blocks of technical documentation and which are associated to the appropriate car function groups and car components (see below). Technical writers have to produce high-quality documents which have to adhere to the general principles of

- Consistency,
- Comprehensibility,
- Non-ambiguity, and
- Process-oriented preciseness

which all feed into translatability. Technicians and mechanics, on the other hand, are the consumers of this information. Their work is demand-driven; therefore they need:

- Fast and easy access to the right information at the right time (electronic delivery, retrieval software and update mechanism).
- Simple but precise descriptions of, for example, repair procedures.
- Technical information which bridges the chasm between technically correct descriptions and their own perhaps more economic but sloppy workshop jargon.

In the following, we will introduce in more detail the three different quality requirement domains, which we have identified as diagnostic quality dimensions.

### 2.1 Workflow Quality Requirements

Today, technical documentation is a sequential process performed over several stages with very restricted communication channels between the different stages. The main stage in this process is authoring which is concerned with the actual composing and writing of service information and repair instructions (here, we will only deal with these types of technical documentation because this is the main application area of MULTIDOC). Authoring is preceded by an information gathering and documentation design stage. In this stage, the information from the design and construction departments (product data and service data) is converted into a form suitable for the consumers of technical documentation. In our case, these are workshop technicians and mechanics; in other cases it could be the people of the marketing department, the high level management, or even the car owners. This conversion procedure mainly affects the wording used to describe a certain technical fact, and therefore it is

massively terminology related. The terminology concerns not only the naming of car components and car function groups, denoted by nominal terms such as nouns and multiword units, but also the naming of service and repair activities, denoted by verbal terms.

Globally operating car manufacturers are obliged to deliver their technical documentation in SGML (ISO 8879) format in accordance with the Clean Air Act Amendment (CAAA) 1992 as specified in the SAE J2008 norm. The employment of SGML in the documentation process not only has opened the way to view documents in a content-oriented way (see below) as opposed to the predominant layout orientation in Desktop Publishing (DTP) systems but also to deploy the power of SGML to better guide and control the authoring process and the whole documentation value cycle, including maintenance, update and versioning.

With the completion of the authoring stage technical documentation is stored in a document management system (DMS) for the further processing in the so-called acceptance stage, where technical checks and legal checks are performed, and in the editing/formatting stage, where the documentation is prepared for different types of delivery (paper, CD, Web, and so forth). The very last stage in the documentation process is translation which in most cases is done by external translation agencies or translation companies. The translated documents are also stored in the central DMS but there are no sufficient control mechanisms to control the translation process and follow-up translation activities during the documentation maintenance phase.

The most obvious quality requirements for the documentation process are thus derived from the following business problem areas:

- Combination of product data and documentation data.
- Reuse of information.
- Linkage of source language information and target language information.

All three areas benefit from the definition of so-called information objects. Thus, information objects are also one part of the product. They are represented either as a geometric representation (product data diagrams) or as an SGML representation in form of an SGML tagged text unit, and combined through an abstract representation. This view permits the effective, timely and accurate description of the product components and associated service and repair processes, and the ability to manage product documentation as a product.

## **2.2 Documentation Quality Requirements**

During the quality requirements analysis for technical writers, a number of application areas for the employment of LT functionality have been identified; among them the most important are:

- Terminology and abbreviation consistency.
- Spell checking and grammar checking.
- Style consistency, including corporate writing guidelines, i.e. controlled language.
- Intelligent information object search and retrieval.

- Foreign language support in different forms such as bilingual and multilingual glossaries, summarisation, information retrieval and indicative translation.

These areas also contribute to the reusability of the information objects in terms of information structuring (form, not layout, see above), and information content, which aims at conceptually precise descriptions of service and repair operations. For example, if in a repair operation the mechanic has to put away a specific part component of a car before executing a certain repair step, this has to be reflected in the repair information with the right wording and the right sequencing as exemplified in Listing 1.

```
<op_stepgrp id="v114" size="s1">
  <op_step><note> ensure extreme cleanliness</note>
    <op_substep type="disconnect"> ... </op_substep>
    <op_substep type="release"> ... </op_substep>
    <op_substep type="remove"> ... </op_substep>
    ...
  </op_step>
</op_stepgrp>
```

#### **Listing 1. Step Group Information Object**

This SGML excerpt of a repair information object shows how this is achieved. The parameters of the step group pattern (op\_stepgrp tag) determine the characteristics of a certain repair operation, which the technical writer has to describe, and which the workshop mechanic has to follow when executing the repair operation. Additional conceptual information specified in the type parameter associated to the op\_substep SGML tag triggers the selection of the right wording (terminology and corporate style guidelines) of the repair operation. This then will also control the appropriate and correct translation of this repair operation in a foreign language even if there are cultural differences in service and repair behaviours.

Besides the above introduced principles, the employment of LT in these areas has also an impact on the time and costs. As an example, we will demonstrate that the effective control of terminology helps to reduce costs at a very early stage of the documentation workflow. This is motivated by the costs that are needed to detect and repair a terminology error. Let us assume that a unit cost of one is assigned to the effort required to detect and repair an error during the authoring stage, then the cost to detect and repair an error during the data gathering, harmonisation (synchronisation between product data and product documentation) and documentation design stages (which are similar to the requirements stages in software engineering) is between five to ten times less. On the other hand, the cost to detect and repair an error during the maintenance stage of documentation is twenty times more. The reasons for this large difference is that many of these errors are not detected until well after they have been made. This delay in error discovery means that the cost to repair includes the cost to correct the offending error and to correct subsequent investments in the error. These investments include rework (perhaps redesign) of documentation, rewrite of related documentation, and the cost to rework or replace documentation in the field.

This shows that errors made at early stages in the documentation workflow are extremely expensive to repair. If such error occurred infrequently, then the contribution to the overall documentation cost would not be significant. However, terminology errors are indeed a large class of errors typically found in complex technical documentation. These errors could be between 30 % and 70 % of the errors discovered in technical documentation. It seems reasonable to assume that a 20 % or more reduction in terminology errors can be accomplished at various levels of organisational maturity, in particular with the employment of LT functionality. Because of the multiplying effect, any such reduction can have a dramatic overall effect to our project's bottom line (time and costs, future revenues and increased competitiveness), and thus contributes to the overall documentation quality and the user's satisfaction.

Similar calculations were obtained for abbreviation errors, spell and grammar errors, and style errors, although their correction can only be accomplished during the authoring process, i.e. the writing and composing of the information objects. These examples profile that we are able to define the central and measurable metrics cost and time for the employment of LT components which can be further classified by their contribution to the overall increase of the so-called "hit rate". The "hit rate" is concerned with the measuring of the effectiveness and efficiency of information object search and information object reusability, including the reuse of already translated information objects. This is important because today inefficient search and retrieval facilities contribute to the redundancy of information object storage, which then has an impact on unnecessary follow-up translations causing additional costs.

The information consumers in the automotive workshops need precise information in terms of structure and content at the right time to assure efficient and effective service and repair measures. Here, the LT employment will contribute to certain search and retrieval operations in hotline information applications (see, for example, [8]), including a "translation-on-demand" option in cases where a specific hotline information is not available in a certain language. In the latter application, the maintenance of a terminology repository that also supports domain-specific action and event readings for verbal terms contributes to a successful and terminologically correct "shallow translation" (indicative or informative translation) of the hotline information.

### **2.3 Multilinguality Quality Requirements**

Multilinguality plays a very important role in automotive technical documentation. Today, the automotive industry is faced with the following serious bottlenecks in addition to the above discussed translation related aspects:

- More and more languages in which product documentation has to be published; there is a tremendous increase in Asian and East-European markets.
- Increasing costs of translation.
- Inappropriate lead time of the translation process.

- Poor or no possibility to measure and control the translation process, also in terms of reusing already translated information objects.

The long-term goal within the MULTIDOC project is the definition of a Translation Engineering (TE) methodology and a TE process (method and procedures) which gives up the present way of viewing the documentation process as strictly chronological or linear, not linked with product data environments, and of translation being a separate step at the end of the processing chain. The most important investigation areas to reach this goal are:

- Graphics and other multimedia incarnations, such as video, animation and virtual reality applications, may enrich or even replace text in certain information objects and facilitate new approaches to information production such as symbolic authoring.
- Translation-on-demand policy to allow for an efficient and effective control of the actual translation needs because not all information objects need to be stored in every language that is supported by the business.
- Compilation of documentation from multilingual information objects, either already stored in a foreign language, translated on demand, or generated from an abstract representation; this allows for the simultaneous delivery of multilingual documentation.

Listing 2 below exemplifies that multilinguality can be achieved with the already introduced SGML authoring approach.

```
<op_stepgrp id="v114" size="s1">
  <op_step><note> &note_clean </note>
    <op_substep type="disconnect"> ... </op_substep>
    <op_substep type="release"> ... </op_substep>
    <op_substep type="remove">      <en> ... remove ... </en>
                                   <de> ... abbauen ... </de>
                                   <se> ... ta bort ... </se>
    ...
  </op_substep>
  ...
</op_step>
</op_stepgrp>
```

**Listing 2.** Multilingual Step Group Information Object

### 3 MULTIDOC LT Components and LT Quality

#### 3.1 Language Technology Components

An LT component normally consists of a software part and a lingware part to which different evaluation patterns can be assigned. Whereas for the software part developers and users mostly apply the software standards within the ISO 9001 and

SEI/CMM framework, especially the evaluation process is most often carried out in accordance with the ISO 9126 "Software Quality" standard with commercially available source code control products, such as the Logiscope system of Verilog, there is no consensus on "Lingware Quality" evaluation patterns today. The EAGLES initiative has proposed to apply ISO 9126 to Natural Language Processing (NLP) systems ([1]); however, they did not explicitly distinguish between the two parts, and therefore we still do not have measurable metrics for lingware.

Before going into the details of our lingware evaluation patterns, we will list the LT components that we considered in our MULTIDOC evaluation work, and how the evaluation work triggered the further development of these components.

On the one hand, our goal is to support the authoring process along the above mentioned principles, and on the other hand, to foster the process of defining the form and content of information objects and to maintain them through their whole life cycle. For both goals the employment of the following LT products and components is our focus:

- Basic LT components such as morphological, syntactic and semantic analysers and generators for a number of languages including German, English, French, Spanish, Italian, Swedish and some Asian languages, with corresponding dictionaries, including bilingual dictionaries.
- Checking utilities for orthography, grammar, style and consistency derived from (or based on) the basic LT components.
- Translation utilities either derived from the basic LT components or complete MT systems and translation memory (TM) systems.

This selection imposes a distinctive quality degree on the LT products (see below) which cannot be achieved by a monolithic and proprietary system design. Within MULTIDOC we are aiming at distributed system solutions that fit with the targeted distributed documentation environments.

### **3.2 Language Technology Quality**

In our evaluation approach, lingware is defined as the intellectual creation comprising formal natural language descriptions, rules and any data, information and knowledge pertaining to the operation of a natural language processing system. This definition is in accordance with ISO 9000-3 which provides the guidelines for the application of the ISO 9001 standards that are concerned with the development, delivery and maintenance of software. A lingware product then is a language enabled system, i.e. a language technology product or component. It is the complete set of computer programs, procedures and associated documentation and data including the lingware designated for delivery to a user. Now, we are also in the position to define "Lingware Quality" in accordance with ISO 9126, which is the totality of features and characteristics of a lingware product (LT product) that bear on its ability to satisfy stated or implied needs. As with the definition of "Software Quality" this is a very broad and flexible definition which needs further refinement for its actual applicability to an LT product, i.e. a quality model. Instead of evaluating the above

listed LT components as they are, i.e. with their built-in general language coverage (vocabulary and grammar), the language resources are continuously enriched with terminology, syntactic, semantic, and translation memory data according to our particular application scenario.

This approach of a cyclic evaluation gave us the possibility to even apply ISO 9126 derived metrics to the lingware part of the components (besides the source code control of the software part), in particular for the ISO quality factors functionality, reliability, usability, efficiency, maintainability and portability, as well as the EAGLES extensions to ISO 9126 customisability and scalability. The results of the cyclic evaluations constantly feed into further refinement and improvement steps. This work also gave us new insights for the future developments of the components, especially for their deployment in networked applications as proposed in [9].

Systems that can be evaluated in this way must be open, extensible and integratable on the software level through the specification of appropriate APIs, and on the lingware level through the specification of suitable "LT APIs" that permit the communication with the existing lingware resources, or through already existing system utilities that allow users to customise the lingware resources or to define their own lingware resources (lingware development environment). An LT API is also provided if the system permits the use of resources according to existing and emerging exchange format standards such as MARTIF (ISO/FDIS 12620) and OLIF ([11], [12]) for terminology and general lexicon resources, the TMX format of the OSCAR working group of LISA for the exchange between TM systems ([13]), and the OpenTag format for text data ([6]). Today, these standards only allow for an ASCII (ISO 8859) based encoding of the data, except TMX which already permits Unicode data encoding (ISO/IEC 10646).

To allow for a strict user-centred evaluation process, we have also included a so-called verification step. In this step the users contribute to the finalisation of the adapted evaluation method and to the definition of the evaluation metrics. The verification process is performed on a theoretical level taking, however, into account the user's genuine working environment as described above.

## **4 MULTIDOC Evaluation Methodology**

### **4.1 Conceptual Framework**

In our evaluation scenario we distinguish three categories for the evaluation:

1. Task which determines the specific requirements in terms of application (domain), system (operating aspects such as resource allocation) and process (workflow integration).
2. Domain which identifies the applicable norms and standards.
3. Safety which defines possible risks and safety levels of a specific workflow.

The basic ingredients that have to be defined and acknowledged by all project partners are:

- Approved terminology for all verification and validation tasks.

- Identified product with its sub-components including an appropriate documentation.
- Process according to the task (verification or validation).
- Testing environment (simulated working environment test, restricted field test, production test).

An additional contribution to our evaluation framework is derived from the ISO 9004 standard for "After Sales Servicing" for the particular areas risks, cost and benefits which play an important role in our industrial task-specific application scenario.

## 4.2 Diagnostic Evaluation

The evaluation methodology we have adopted within the MULTIDOC project is a diagnostic evaluation. Our definition of this type of evaluation differs from the EAGLES definition ([1]) in so far as we include the user requirements of a task-specific application in our evaluation methodology. This view does not only extend the EAGLES definition, it also permits the application of the ISO 9126 quality model for software systems to lingware systems including lingware developments in a balanced way. We call this a "braided" diagnostic evaluation. It means the systematic and regular application of predefined evaluation principles during the customising phases and scaling phases of a multi-purpose LT product or component. These principles are the central features of continuous quality control, progress monitoring and quality assurance during the evaluation process, and during the further development of the LT component. In this context, the meaning of the term development is twofold. On the one hand, it concerns the software solutions of the system, and on the other hand, the lingware resources such as grammars, lexicons, thesauri, corpora, style rules, test suites, translation modules, and so forth, and the language enabling technologies such as analysers, translators, and generators that implement the language enabled applications. The "braided" diagnostic evaluation methodology is defined in terms of:

- User and developer requirements which define the aimed at or needed functionality and the existing functionality of an LT product or component.
- Evaluation factors with their associated characteristics, metrics and value scales for multi-purpose and task-specific applications in terms of usability (deployment potential), reliability (stability in different application scenarios, see above), efficiency (throughput capabilities according to time and space considerations), maintainability with respect to future customisability and scalability of the LT component. This is accomplished by a quality model that associates to each of the before mentioned quality factors a quality criterion such as readability, testability, integratability, and so forth. If a particular quality criterion is satisfied then the associated quality factor is also satisfied.
- Process steps according to the task-specific evaluation principles consistency, comprehensibility, non-ambiguity, and operation oriented preciseness, which all contribute to the more general principle of translatability.

The actual diagnosis is then similar to a fault tracing procedure along the specifications of a symptom tree or graph. The edges of the symptom tree specify a certain phenomenon and the nodes trigger appropriate actions or they are linked to predefined requirement classes. This resembles very much the idea of the quality tree concept in software evaluation. A phenomenon can be derived from the predefined principles and evaluation patterns (factor, criterion and metrics) in terms of an error classification, and an action defines a certain measure for the error repair.

For example, if a defined style checking rule does not apply according to a pre-selected set of input structures (evaluation test suites), then a possible repair operation has to further identify possible error locations as well as associated steps for finally fixing the cause of the error. Such a repair operation has to obey subsequent tests to ensure that the error fixing is monotonic. The evaluation test suites are derived from the profile of a certain information object, for example, language (style) use and associated quality factors.

## **5 MULTIDOC Evaluation Process**

The described evaluation methodology, and in particular the actual performance of the evaluation method, turned out to be very well suited for the MULTIDOC application since the evaluation process fostered in addition the communication between users and developers, and therefore, a common understanding of the different procedures could be maintained at each stage of the project. This also minimised the risk potential of the LT developments, so that the users were not surprised about the results and possible side-effects of the LT component's behaviour.

### **5.1 Quality Requirement Definition**

According to our quality dimensions (workflow, documentation, translatability) and the ISO 9126 quality factors appropriate quality criteria and possible subcriteria are defined. For this the LT product is decomposed into its major components because requirements derived from the overall product may differ for the different components.

It turned out that it is not an easy to accomplish task to evaluate LT products (neither on the software level nor on the lingware level). This is due to the fact that currently LT vendors are not really open to indepth customer evaluations because they fear the loss of their intellectual assets. Another strong aspect is the monolithic design of many LT products which mostly aim at personal use but not at enterprise use in a distributed information technology (IT) environment. However, this situation is changing now because LT vendors are looking for alliances in the field of sharing lingware resources (see above) and for cooperations with LT VAR/OEM partners for LT add-ons and even certification processes.

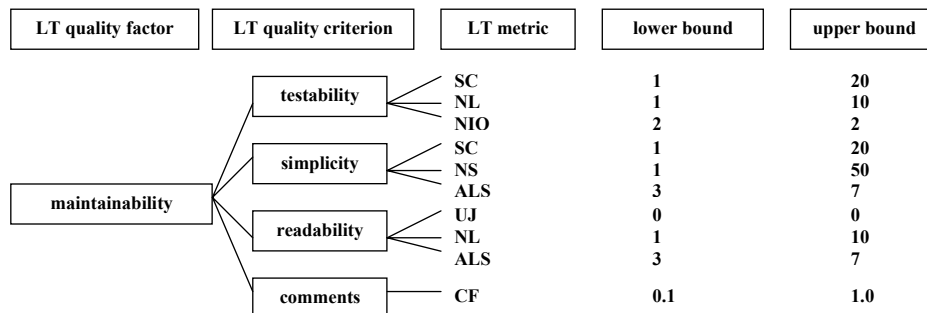
Given this situation, we evaluated the software part of an LT product according to its integration potential (existing command language, APIs, and so forth), its runtime

behaviour and its reliability in the IT environment of an enterprise, and its cost/benefit ratio for the whole documentation process. The evaluation of the lingware part was restricted to existing lingware development environments for adding and enhancing lingware resources and possible existing LT APIs. In addition, the delivery throughput of the LT vendor's service was also evaluated. For the research based systems we could apply the whole evaluation cycle according to our evaluation methodology, i.e. software source code control with Logiscope and lingware validation based on our LT metrics (see below).

## 5.2 Evaluation Preparation

Since we tried to adapt the chosen software metrics to lingware metrics, we selected the following quality metrics for both parts: number of statements (NS), structural complexity (SC), nesting levels (NL), unconditional jumps (UJ), comment frequency (CF), average length of a statement (ALS), number of inputs and outputs (NIO), and structural similarity in one function (SSF).

Each quality factor defined for the different quality requirement dimensions is associated with a set of quality criteria, and each criterion correlates with a set of metrics which were assigned to different rating levels (upper and lower bounds). An example of such a quality model for the quality factor maintainability is given in Figure 1.



**Figure 1.** Example of a Quality Model

The quality criterion is satisfied if all measured metrics are within the defined boundaries. In addition, we can also assign a specific weighting for each metric which then contributes to the satisfiability ranking of the quality criterion. If all quality metrics are within the defined boundaries, then the software or lingware function/module gets the classification simple, small deviations account for the classification normal, large deviations imply the classification complex or critical if almost all metrics are above the boundaries. The classification undefined is assigned in those cases where a certain metric cannot be assigned. The assignment of these

classes gives hints for further indepth code inspections. In particular, it also helps the developers in streamlining their code (software and lingware).

### **5.3 Evaluation Procedure**

The last step of the evaluation process is the actual measurement of the selected metrics, which is then followed by a rating step. In this assessment step the quality of the LT product is summarised and the specifications for the adaptation process of the existing LT component are defined, i.e. the system "as is" including its language resources, resulting from the evaluation steps performed on a general level, especially metrics such as maintainability, customisability and scalability (see the discussion of APIs above). Based on the results of the evaluation of the current component, specifications can be developed which yield at the optimisation of the system's performance with respect to the predefined evaluation metrics. These specifications relate to concrete requirements resulting from the specific application domain (see above), for example, the treatment of a certain information type, typical linguistic phenomena (controlled language), use of domain-specific terminology, and so forth.

Aspects related to the given information technology infrastructure, for example, a network-based deployment including specific evaluation strategies that could be fulfilled by "intelligent" software agents (see [9]) are also taken into account, as well as time and cost aspects of the product's actual deployment in a corporate environment.

As already outlined above, these evaluation steps are performed in iterative cycles. The continuous communication between the users and the developers ensures that the different evaluation patterns are applied in an optimal way, and that feedback is given on a regular basis. In addition, this processing strategy permits the adaptation or even the redefinition of the evaluation patterns, thus introducing a certain dynamic, i.e. the "braid", into the otherwise static (and perhaps inflexible) evaluation procedure.

## **6 Conclusion and Perspectives**

In this paper we have introduced the MULTIDOC evaluation methodology based on diagnostic dimensions and performed through a cyclic processing technique (method). The utilised methodology is entirely user-centred with additional support through developer oriented requirements to sanction a "braided" evaluation. This approach allows for a clear distinction between the software level and the lingware level in the evaluation process, and the applicability of the ISO 9126 quality model to both levels on a thorough foundation.

The users of the MULTIDOC project agree on the fact that this approach should also be the standard approach to be adopted by LT vendors to support the integration of an LT component into an existing industrial workflow. Today, neither LT vendors nor LT OEM/VAR service providers operate in this way. In this context, the

definition of an independent certification procedure seems to be most interesting for LT users.

One of the future next steps is the investigation into automatisable processes to permit the development of source code control facilities for LT components, which are similar to the existing software source code control tools.

## Acknowledgements

The MULTIDOC project is partly funded by the European Commission under contracts LE3-4230 and LE4-8323. The content of this paper does not reflect any official statement of the European Commission or the MULTIDOC project partners. The responsibility for the content is solely with the authors of the paper.

## References

1. EAGLES: Evaluation of Natural Language Processing System. Final Report, EAGLES Document EAG-EWG-PR.2, Geneva, Switzerland (1995)
2. Haller, J.: MULTILINT - Multilingual Documentation with Linguistic Intelligence. In: Proceedings of 'Translating and the Computer', ASLIB, London, Great Britain (1996)
3. Haller, J. and Schütz, J.: Integration linguistischer Intelligenz in die multilinguale technische Dokumentation. In Proceedings of EUROMAP Forum 'Sprache ohne Grenzen', München, Germany (1997)
4. Maas, H.D.: Multilinguale Textverarbeitung mit MPRO. In: Lobin, G., Lohse, H. Piotrowski, S and Poláková, E. (Eds.): Europäische Kommunikationskybernetik heute und morgen, KoPäd, München, Germany (1998) 167-173
5. Nübel, R.: End-to-End Evaluation in Verbmobil I. In: Proceedings of Machine Translation Summit VI, San Diego, California, USA (1997) 232-239
6. OpenTag - Formal Specifications. Version 1.1b April-22-1998, Last edit: May-01-1998. Available on the Web at <http://www.opentag.org/otspecs.htm> (1998)
7. Schütz, J.: Language Engineering – Fixing Positions. IAI Memo 0695, Saarbrücken, Germany. Available on the Web at <http://www.iai.uni-sb.de/memos.html> (1995)
8. Schütz, J.: Combining Language Technology and Web Technology to Streamline an Automotive Hotline Support Service. In: Proceedings of AMTA 96, Montreal, Canada (1996) 180-189
9. Schütz, J.: Utilizing Evaluation in Networked Machine Translation. In: Proceedings of the 7th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI) 1997, Santa Fe, New Mexico, USA (1997) 208-215
10. Schütz, J. and Nübel, R.: Multi-purpose vs. Specific Application: Diagnostic Evaluation of Multilingual Language Technologies. In Proceedings of the 1st International Conference on Language Resources and Evaluation, Granada, Spain (1998) 1251-1254
11. Thurmair, G.: Exchange Interfaces for Translation Tools. In Proceedings of MT Summit VI, San Diego, California, USA (1997) 74-92
12. Thurmair, G., Ritzke, J. and McCormik, S.: The Open Lexicon Interchange Format - OLIF. OTELO Report available on the Web at <http://www.otelo.lu> (1998)
13. TMX Format Specifications. Version 1.0 November-25-1997. Available on the Web at <http://www.lisa.org/tmx/tmx.htm> (1997)