THE KORAP USER INTERFACE. ACCESSING COROLA VIA KORAP

NILS DIEWALD[1], VERGINICA BARBU MITITELU[2], MARC KUPIETZ[1]

Abstract. The user interfaces for corpus analysis platforms must provide a high degree of accessibility for ordinary users and at the same time provide the possibility to answer complex research questions. In this paper, we present the design concepts behind the user interface of KorAP, a corpus analysis platform that has evolved into the main gateway to CoRoLa, the Reference Corpus of Contemporary Romanian Language. Based on established principles of user interface design, we show how KorAP addresses the challenge of providing a user-friendly interface for heterogeneous corpus data to a wide range of users with different research questions.

Keywords: User interface design, corpus analysis, KorAP, CoRoLa, software.

1. INTRODUCTION

Designing a user interface for a corpus analysis platform is a challenging task:

> First, due to the diversity of information that needs to be accessed, the interface must be general enough to cover multiple corpora with heterogeneous annotation and specific enough to enable users to find the information they are looking for. Second, due to the diversity of potential users, the query interface has to be designed to favour high acceptability. (Söhn *et al.* 2008: 27)

The focus on ease of use has therefore often reduced the actual performance of a corpus analysis system (Hardie 2012: 382). On the other hand, the functionality made accessible through the user interface has a big impact on the type of research that can be done with the software (Hardie 2012: 383, Anthony 2013: 141).

KorAP[3] is a corpus analysis platform (Bański *et al.* 2012) developed as a successor to the COSMAS II platform (Bodmer 1996) to serve as the main access point to DeReKo, the German Reference Corpus (Kupietz & Lüngen 2014). As part of the DruKoL Aproject[4], KorAP is now also the main access point to CoRoLa (the Reference Corpus of Contemporary Romanian Language; Tufiş *et al.* 2015, Barbu Mititelu *et al.* 2018), and thus

[1] Leibniz-Institut für Deutsche Sprache, Mannheim, {diewald,kupietz}@ids-mannheim.de
[2] Romanian Academy, Research Institute for Artificial Intelligence, vergi@racai.ro
[3] https://korap.ids-mannheim.de/
[4] http://www1.ids-mannheim.de/kl/projekte/drukola.html; DRuKoLA (2016–2018) was funded by the Alexander von Humboldt-Foundation.

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

a common corpus analysis platform for comparative language research. The appropriateness of KorAP for this task results from its neutrality with respect to corpus data and research questions. The client-server-based system provides access to all corpus search and analysis functions via Web APIs, but most users communicate using the primary web-based user interface Kalamar[5]. Although web-based user interfaces have an "instant advantage in usability" (Hardie 2012: 384), as they do not require installation and are available for almost all operating systems, their design still requires sophisticated interface concepts to meet the challenges.

In this chapter, we introduce the user interface concepts of Kalamar (Section 2), the possibilities to customize and extend the interface as it was done for CoRoLa (Section 3), and we present first feedback from users of the CoRoLa corpus (Section 4).

2. USER INTERFACE CONCEPTS

According to ISO 9126-1 (2000) software quality can be categorized into *internal quality*, *external quality* and *quality in use*. These quality levels are interdependent (see Fig. 2): the internal quality of a software (its implementation) has an influence on its external quality (its behaviour during execution); and the external quality has an influence on the perception of the software by the user and thus on the quality in use (Bevan 1999). The internal and external quality is mostly seen (and monitored) from the developer's point of view, implemented through documentation, code reviews, automated tests, and benchmarks. The developer's perspective also leads to certain design decisions, for example regarding *reliability*, *maintainability*, and *portability* of the software. These are three of the six characteristics of software quality identified by ISO 9126-1 (2000), which also includes *functionality*, *usability* and *efficiency*, characteristics more related to the user's perspective on the software. While all characteristics have an impact on the quality in use, this chapter focuses on our design decisions regarding the *usability* and *efficiency* aspects of KorAP[6], with an emphasis on the user interface Kalamar.

In terms of *functionality*, KorAP tries to achieve functional equality with its predecessor and follows a "wishlist" of features collected at the beginning of the project (Bański *et al.* 2012). This preliminary survey gave a first insight into users' expectations of a new corpus analysis platform – and a better understanding of their needs.

A similar approach was followed by Söhn*et al.* (2008), who in a small questionnaire study (n=20) asked about requirements for a corpus query interface. They categorized the interface requirements into three functional areas: *search*, *visualization* and *export of query results*. Although the scope of the questions exceeded the functions of the user interface (e.g. asking for query requirements), Söhn *et al.* (2008) identified desired features to be adopted for their own corpus platform interface design. In addition, they reviewed three corpus query systems (COSMAS II, TIGER Search and Elan) and identified further features to be adopted. One of their conclusions is that they regard the use of already established design paradigms from existing software tools as beneficial for the user.

[5] https://github.com/KorAP/Kalamar

[6] For design decisions regarding *maintainability* and *portability*, see Diewald *et al.* (2016).

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

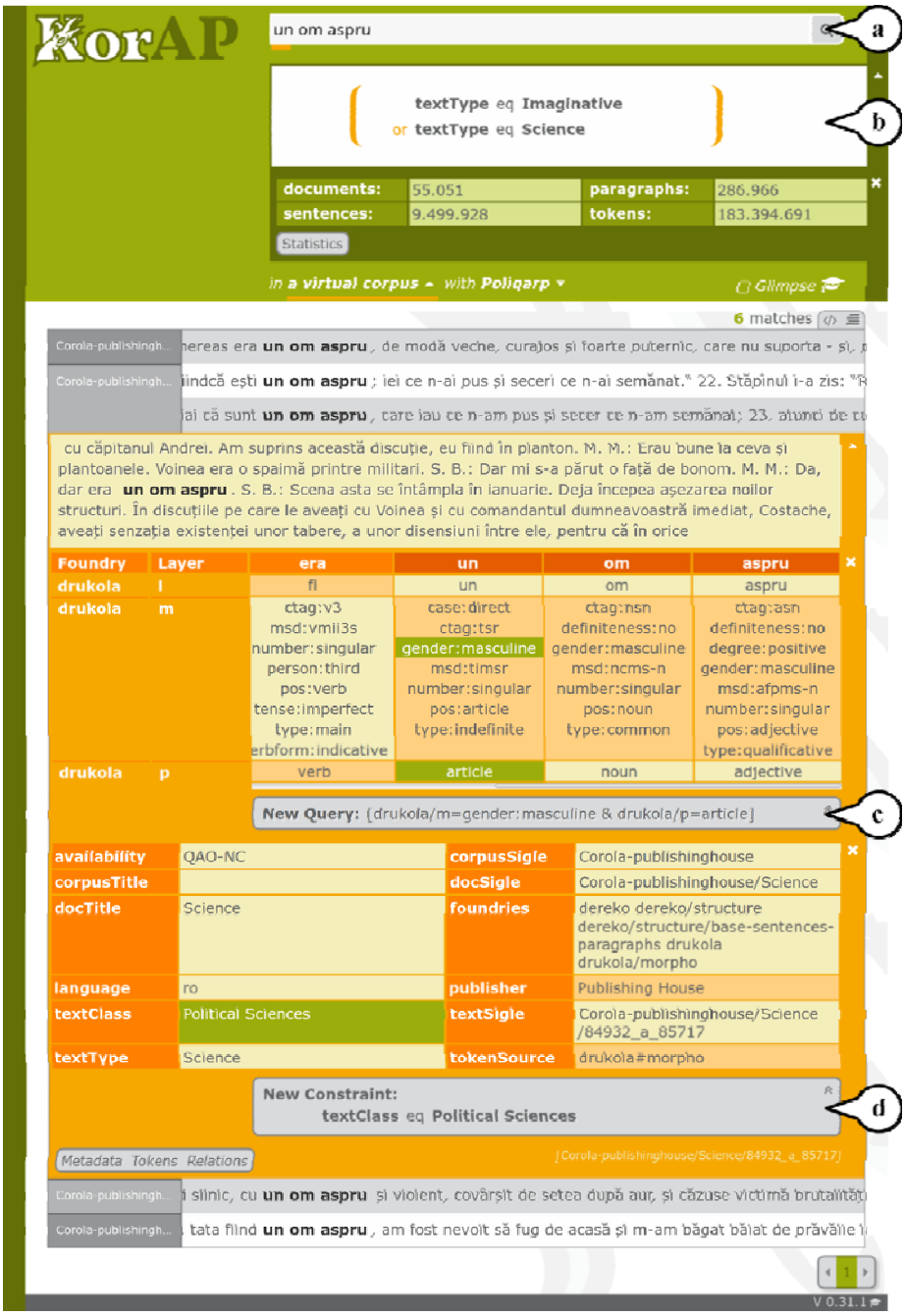


Fig. 1. The expanded user interface with a) the query field, b) the virtual corpus assistant, c) the "query by match" assistant, d) the "corpus by match" assistant.

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

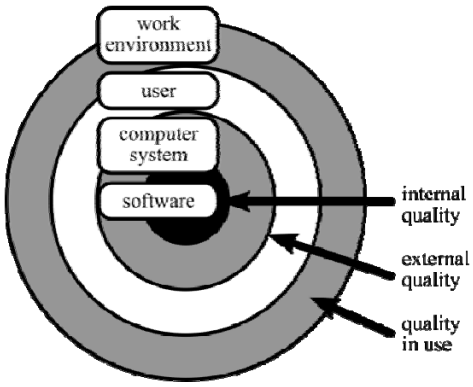


Fig. 2. Internal quality, external quality and quality in use (Bevan 1999: 90).

A larger study (*n*=669, with 401 participants completing the questionnaire in full) was conducted by Fandrych *et al.* (2016) on the use (and users) of oral corpus platforms. In addition to the questionnaire, 15 personal interviews with "power-users" were carried out. Among other things, they discover that the ability to intellectually inspect oral corpus data is more relevant to the user than the ability to query the data, and even encourage a "download first" approach to corpus data (Fandrych *et al.* 2016: 283), although they know that this is discouraged by modern infrastructure approaches such as CLARIN (mostly due to licensing restrictions). Fandrych *et al.* (2016) conclude that the average user does not have a high level of technical knowledge but has high expectations of the usability of the software without being willing or able to spend much time on learning how to use it. On the one hand, this requires that the developers minimize these efforts. On the other hand, they acknowledge that an oral corpus platform must provide a variety of functions to meet all the needs of the diverse user base, concluding: "We are convinced that there is no easy solution to that dilemma" (Fandrych *et al.* 2016: 286).

At the beginning of the KorAP project, an interface design with different user modes for new and experienced users was planned to solve the dilemma ("alternative views", Tidwell 2006: 39). However, this design had some drawbacks: switching from one mode to another, when a user has more experience or needs more functionality, requires in the worst case that the user has to re-learn parts of the interface. In addition, providing two different user interface modes significantly increases maintenance costs. So instead, we decided to create a unified interface that would initially offer the user limited options, hiding most of the complexity of the software and underlying data, but providing *extras on demand* (Tidwell 2006: 45f), and trying "to make the easy jobs easy, without making the hard jobs impossible" (Wall 1996: ix). Starting from a simple KWIC result (keyword in context), the user can embed different views of an interesting match with just a few clicks and thus retrieve all underlying data available (Fig. 3)[7]. The accessibility of corpus data is a driving force for the feature design of KorAP. In this way, users have a convenient way to intellectually inspect all data without having to provide a "download first" option for the corpora.

[7] Hence the title of Diewald *et al.* (2016): the user can figuratively dive in the corpus data.

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)



Fig. 3. With a few clicks, the user can gather all underlying data based on a match.

The user interface of KorAP is divided into several sections (called "panels"), which contain various functional information (see Fig. 4). Currently, panels are available for the following functional areas: a) the creation of the query, b) the display of the search result, c) the creation of the query. In the future, further functional areas are expected to be implemented as panels. These panels allow to provide "actions" (such as switching the text direction from left to right in the result area) and to embed several "views" (i.e. *extras on demand*) into the corresponding panels to provide additional information for the respective area (such as showing statistical information about the virtual corpus, showing annotations about a match, or showing the underlying API response for results). This structure was introduced to ensure a high degree of extensibility to new functions without changing the overall appearance of the interface. It is also the basic structure for plugins (see Section 3). The panel-based layout has been designed to be suitable for different screen resolutions, although we assume that most complex tasks are still performed on desktop or laptop computers (as suggested by an early user tracking study in the KorAP instance of DeReKo).

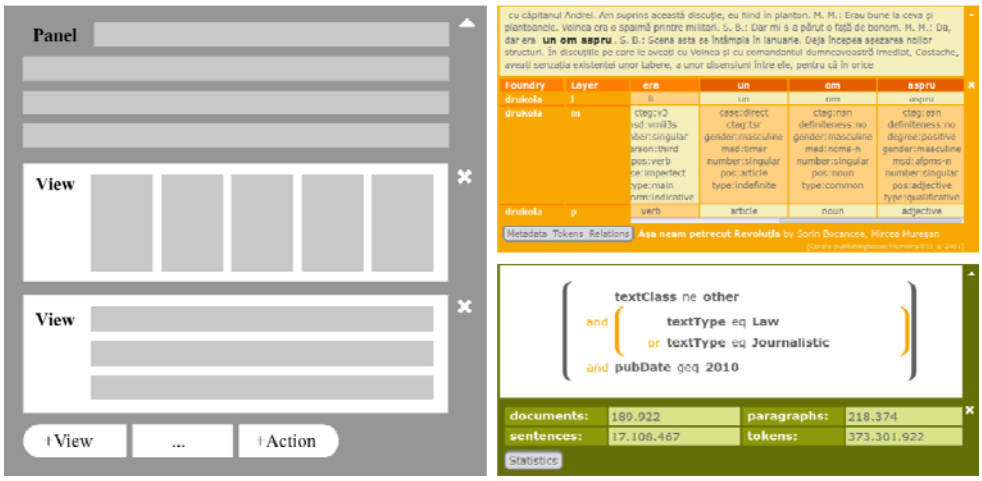


Fig. 4. "Panels", "Views" and "Actions" as elements of the KorAP user interface.

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

In the following, we present various features implemented in Kalamar as examples of our approach to usability. ISO 9241-11 (2006) defines usability as "the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context". Effectiveness refers to the ability to complete tasks accurately, efficiency to the ability to perform these tasks in an acceptable amount of time and effort, and satisfaction to the extent to which the user is satisfied with the system (Bevan 2008). This is accompanied by ISO 9241-110 (2006), which introduces a list of seven so-called "dialogue principles" for the analysis, design and evaluation of the user-friendliness of a user interface: "Suitability for the task", "Self-Descriptiveness", "Controllability", "Conformity with user expectations", "Error tolerance", "Suitability for individualization" and "Suitability for Learning". A dialogue here is the "interaction between a user and an interactive system as a sequence of user actions (inputs) and system responses (outputs) in order to achieve a goal" (ISO 9241-110 2006) and the principles "represent the dynamic aspects of the interface and can be mostly regarded as the *feel* of the interface" (Oppermann 2002: 235). We will associate our examples to these criteria and outline our plans for future improvements.

Suitability for the task requires support for the user to effectively and efficiently complete the task[8]. Our approach of hiding from the user most of the information that is not necessary to complete a task successfully is an important aspect of this principle. We also try to support the user by lowering the entry hurdle for the search task by following the established approach of web search engines and providing only a single search bar with an intuitive query language for at least simple searches. Kalamar's default query language is Poliqarp (Przepiórkowski *et al.* 2004), a variant of CQP (Christ 1994), that is powerful for complex annotation queries (see Cristea *et al.*, in this volume), but supports intuitive word and phrase search, which means that users who are not familiar with the query language can still experiment with searching for keywords as in a web search engine they are familiar with. Kalamar provides some assistants for query formulation, such as the annotation assistant to select the correct annotation tags from several tag sets (see Fig. 7b, embedded by clicking on the orange bar below the query field, see Fig. 1a); the "query by match" assistant (Kupietz *et al.* 2018), which helps to create complex sequential queries based on the annotations of a single sample hit (see Fig. 1c), or the "corpus by match" assistant, which helps to refine a virtual corpus based on metadata information of a single sample hit (see Fig. 1d). We plan to extend the set of assistants, for example to simplify the creation of syntactical queries based on tree annotations.

Self-Descriptiveness requires each step in the dialogue to be immediately comprehensible. Kalamar's query interface describes the query task in a simple, human-readable sentence to make the current query settings visible and understandable at first glance. Figure 5 shows the interface that describes a query in the form of a text template: "[Query] in [Corpus] with [Query Language]". This mechanism is designed to be extensible for future query capabilities as well. Furthermore, we try to visualize information whenever feasible and intuitive to comprehend, for example, the nested view of virtual corpora (see

[8] All short descriptions of the dialogue principles are based on the definitions in ISO 9241-110 (2006).

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

Fig. 1b) or the tree and arc visualization of syntactic annotations. However, initial feedback from users of DeReKo and CoRoLa shows that Kalamar's self-descriptiveness needs to be improved; for example, the integration of some assistants is not yet obvious for users unfamiliar with them.

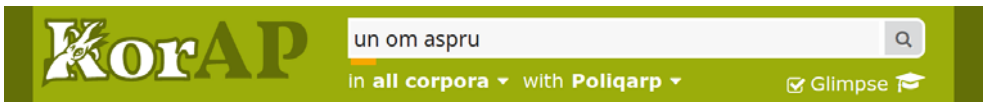


Fig. 5. Query description as sentence *un om aspru* in all corpora with Poliqarp.

Controllability requires that the user be able to initiate and control the direction and pace of interaction. Kalamar does not require any order to create a query or virtual corpus prior to searching. The search is also "stateless", meaning that all query parameters are reflected in the URL of the web application, allowing the user to bookmark or share a query and to navigate through the search history using the browser's "Back" and "Forward" buttons. Unfortunately, the aspect of statelessness has not yet been fully implemented: changes to the virtual corpus are not immediately reflected in the history of the browser, so that "Undo" and "Redo" methods are still missing for the corpus composition.

Conformity with user expectations requires that the user interface is consistent and corresponds to user characteristics such as task knowledge, education, experience, and to commonly accepted conventions. Kalamar's simple design, based on "panels" and "views", attempts to create a consistent mechanism across the interface. The icons of Kalamar are based on the Fontawesome[9] icon set and try to follow the visual language of established graphical user interfaces in order to make semantics clearer to the user (e.g. for actions such as minimizing panels or closing views, Fig. 4). With regard to the specific task of corpus query, KorAP follows established forms of result presentation, such as KWIC lines – mainly in order to facilitate transition of users from the predecessor platform COSMAS II.

Error tolerance requires that the user interface achieves the intended result with either no or minimal action by the user, despite evident errors in the input. If a query contains a syntax error, Kalamar interprets and forwards all errors returned by KorAP's query parsing component[10]. This often results in an error message pointing to the position in the query where the error occurred, with the cursor in the same position for immediate correction (see Fig. 6). When a virtual corpus query is rewritten by the system due to license restrictions (Bański *et al.* 2014), as is necessary in the case of DeReKo, the virtual corpus is flagged with an icon to indicate the change. We are constantly improving error messages, whereby the semantics of query rewriting does not always follow the user's expectations, so we are trying to enhance the semantics and mechanisms of query rewriting in the future.

[9] https://fontawesome.com/, published under SIL Open Font License 1.1, Creative Commons Attribution 4.0, and MIT License.

[10] See https://github.com/KorAP/Koral/ and Bingel & Diewald (2015).

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

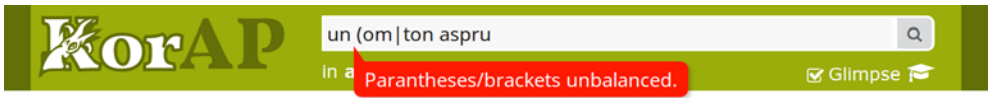


Fig. 6. A syntax error is reported in the query and the exact position at which it occurred is pointed to.

Suitability for individualization requires that the interface is modifiable to suit the task needs, individual preferences, and skills of the user. Kalamar is currently not customizable for the user, especially for the access to CoRoLa, because CoRoLa does not need any user accounts to store preferences or further information about the user which lasts longer than the browser session. While the backend of KorAP already supports user preferences via Web API requests[11] (such as saving defined virtual corpora), these features are not yet available in the frontend.

Suitability for learning requires that the user interface supports and guides the user in learning to use the system. Kalamar's assistant tools not only help to create complex queries and virtual corpora, but also preview the created query fragments to the user prior to searching (see Fig. 1c,d). This helps the user to become more familiar with the query language and the underlying corpus data. In addition, Kalamar offers an embedded tutorial (see Fig. 7c) allowing the user to learn more about the query system and test example queries directly. In addition, we plan to embed an interactive tour that accompanies the tutorial and guides the user through some of the core functionalities by using a simple example task.

3. CUSTOMIZATIONS AND EXTENSIBILITY

As Söhn *et al.* (2008) stated, the diversity of users is a major challenge in the development of user-friendly corpus analysis software. This diversity is not only limited to the dimensions of novice users up to expert users or users with a low level of technical knowledge up to users with a high level. Diversity can also come from different language communities, requiring the user interface to be fully localizable. If the software has to be adopted for different and heterogeneous corpora (heterogeneous in terms of language, annotation layers, metadata, etc.), feature sets must also take this into account. And finally, research questions of users may differ widely. Basic functions of the software can be irrelevant to a particular research question, while advanced functions can be very relevant, requiring improved accessibility for some users while distracting others.

KorAP is primarily developed as the main gateway to DeReKo, but is also the most important query platform of the CoRoLa corpus, is also part of the EuReCo initiative (Kupietz *et al.* 2017), and as an open-source and free software solution[12] usually has to be adaptable to various needs. This is especially true if the software is used for comparative research.

[11] For a documentation about these endpoints, see https://github.com/KorAP/Kustvakt/wiki.
[12] KorAP is licensed under BSD-2.

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

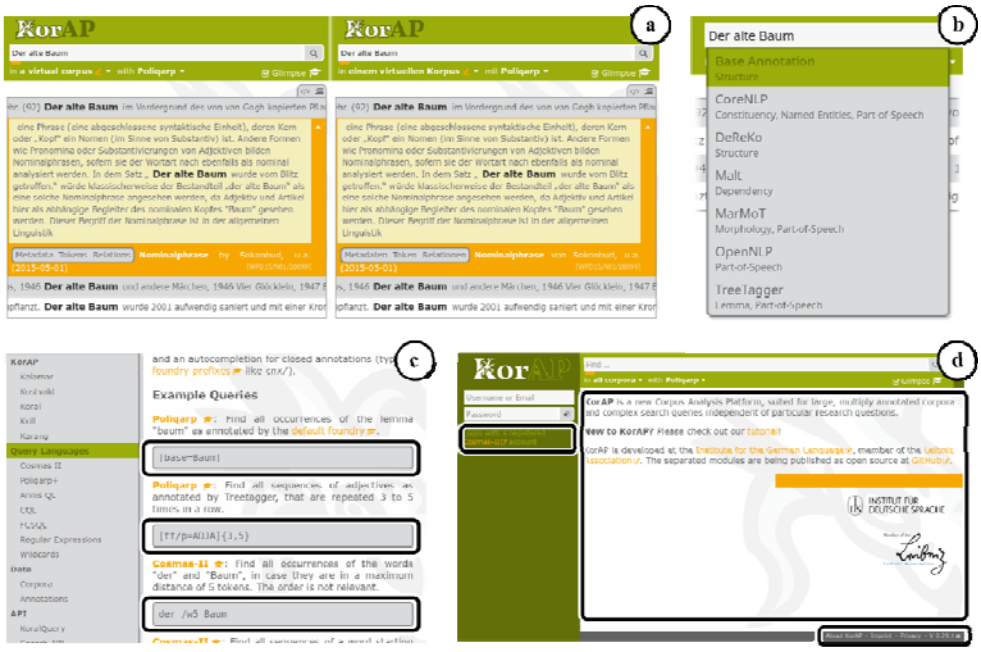


Fig. 7. a) Localization of the user interface, b) customization of the annotation assistant depending on the underlying corpus, c) customization of tutorial examples depending on the underlying corpus, d) customization of user interface templates.

For this reason, Kalamar is customizable in various ways. For example, a multilingual translation dictionary is utilized to provide the user with a localizable interface based on browser preferences (see Fig. 7a). German and English are available by default. For the DRuKoLA project, parts of the dictionary were adopted for Romanian as well.

The CoRoLa corpus also contains annotations other than DeReKo, which require an adaptation of the annotation tools and show only the annotation layers available in the underlying corpora (see Fig. 7b). The different annotations also affect the documentation of KorAP: Kalamar supports the user with an embedded user tutorial including testable query examples (i.e. a search process can be initiated by clicking on example queries). For the DRuKoLA project, these parts of the documentation were rewritten to make them agnostic regarding the query examples[13], since they did not work with the underlying corpus data. The new documentation system now has customizable query examples for testing (see Fig. 7c). Kalamar also supports the integration of additional elements into certain areas of the user interface (see Fig. 7d). Since DeReKo (unlike CoRoLa), for example, requires users to sign a contract for permission to use, technical and legal differences must be documented in the frontend (e.g. integration of a registration form with additional information, imprint).

The diverse research questions of users of corpus analysis tools often demand very specific features that may not be of general interest. These features will probably never

[13] The authors would like to thank Susanne Feix for her work on the documentation.

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

become core functionalities of KorAP. Nonetheless, in order to support very specific use cases, KorAP has made initial attempts to provide interfaces for plugins, so that externally developed features, especially features with closed source components or components that are not compatible with KorAP's BSD-2 license, can be integrated into Kalamar and communicate with the backend via Web APIs. This represents an additional challenge in designing a user interface for corpus research: The requirement to keep the design extensible for new features and new developments in corpus linguistics while maintaining the usability of the system and preventing the user from having to re-learn the interface functionality.

The integration areas reflect the aforementioned extensible "panels": plugins should be added as "actions" or "views" on the query (e.g. word list plugins, inclusion of a visual query creator as in the windows interfaces of COSMAS II, results (e.g. export plugins, visualizations of match distributions), matches (e.g. translations and mappings of annotations, embedding external linguistic resources) or virtual corpora (e.g. visualizations of virtual corpus distribution). The necessary interfaces to connect the KorAP frontend to external plugins are added on demand. In order to enable communication between plugins and the KorAP backend, OAuth 2.0[14] was integrated as an open protocol for (Web) API authorization.

4. USER EXPERIENCE

Since making the decision of creating CoRoLa two types of users have been targeted: language engineers and linguists. Only the latter is considered here because engineers interested in using the corpus for extracting data to be used in further applications do not need an interface for accessing it, even if they need to intellectually inspect the corpus before working with it (see Section 2). However, although we refer here to one type of users, "linguists", one must bear in mind that this is a heterogeneous group: there are linguists who are well accustomed to querying corpora (given their experience with corpora for other languages queryable with various tools), there are others who are not corpora users but are open to learning, to different extents (see above the discussions about the diversity of users). Nevertheless, they all have high expectations from a corpus.

At the beginning of the CoRoLa project, a survey was conducted within the Romanian linguistic community for identifying the degree of interest language specialists show for such a resource, the activities they consider a corpus useful for, the kinds of linguistic information they would query a corpus for and the types of texts that would be relevant for their activities. The survey showed the linguists' great interest in working with such a resource for Romanian, as well as a whole range of activities that need diverse information from all kinds of texts, both written and oral. The results of the survey were taken into consideration by the corpus developers when designing its structure, its processing, metadata fields, annotation levels and when deciding upon a technical solution for its querying. The following discussion will focus only on the written texts, as KorAP does not offer the possibility of indexing and querying an oral corpus, at least for the moment.

Analyzing the users' experience with a corpus offers another perspective on it, providing orientation to the developers and maintainers in their activity.

[14] https://oauth.net/2/

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

Prior to starting using a corpus, the users must become aware both of the corpus characteristics and of the platform that indexes it: the former implies knowledge of the corpus content, structure, annotation levels and formalized labels used for their encoding, metadata fields; the latter implies knowledge of the platform facilities in order to extract as much and diverse information from the corpus as necessary, as well as of the query language. Only if users make these efforts can we speak about effective use of corpora in language studies, which, of course, depends crucially on the corpus characteristics, too.

CoRoLa users have been successful with lexical queries, given the easiness of formulating such queries in a way very familiar to them (see above the Section 2: Suitability for the task). The "query by match" assistant, once introduced to them, opened whole new opportunities for their experience with the corpus. This was the stepping stone to becoming independent users of the platform.

Quantitative and qualitative data about CoRoLa and the metadata content are available in the main papers about the corpus (the most recent one being Barbu Mititelu *et al.* 2018). They also offer information about the annotation levels, but users are encouraged to access the corpus website[15], for becoming familiar with the specifications followed in the morphosyntactic annotation of CoRoLa. Moreover, these (downloadable) files contain the Romanian translation of the names of all the labels, features and values used in the annotation, which is useful for users who are not familiar with the English terminology applying in this domain. On the other hand, for KorAP a tutorial is available on the platform[16], as well as an easy to follow manual (in English and German only, for the time being) for the query language.

The written component of CoRoLa is accessible to users in the KorAP platform for a wide, yet incomplete, experience of the corpus. KorAP copes very well with the corpus size: results of simple queries are returned in no time, while longer queries, using regular expressions, take only a few seconds before displaying the snippets of texts. Users rely on its robustness, which, alongside the capacity of managing huge text quantities, played a decisive role in choosing it for the Romanian corpus.

Reports by users of CoRoLa helped to identify and fix bugs in the system (e.g. regarding wrong behaviour for negation in queries), improve the usability regarding different underlying corpora (see Section 3), and improve the system for previously unexpected workflows (e.g. support of international quotation marks in addition to ASCII quotation marks in regular expressions, to allow for copying queries from MS-Word documents).

5. CONCLUSION

The design of user interfaces for corpus analysis software is a challenging task. The user interface of KorAP, Kalamar, has been designed to make access to corpus data and all research facilities provided by KorAP user-friendly while retaining the ability to answer complex research questions.

We plan to enhance the usability of Kalamar by introducing the features described in Section 2. We also plan to further develop the customization and extension capabilities.

[15] http://corola.racai.ro/; Section Interogare.
[16] http://89.38.230.10:5555/

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

To improve our understanding of user issues, we plan to continue working closely with CoRoLa users and other EuReCo initiative partners. We are also preparing a survey with DeReKo users and have started to monitor the use of queries with the web analysis software Matomo[17], a practice also planned for CoRoLa.

REFERENCES

Anthony, L., 2013, "A critical look at software tools in corpus linguistics", *Linguistic Research*, 30, 141–161.

Bański, P., P. M. Fischer, E. Frick, E. Ketzan, M. Kupietz, C. Schnober, O. Schonefeld, A. Witt, 2012, "The New IDS Corpus Analysis Platform: Challenges and Prospects", in: N. Calzolari, K. Choukri, T. Declerck, M. Uğur Doğan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, S. Piperidis (eds), *Proceedings of the 8th International Conference on Language Resources and Evaluation* (LREC 2012), May 23–25, Istanbul, 2905–2911.

Bański, P., N. Diewald, M. Hanl, M. Kupietz, A. Witt, 2014, "Access Control by Query Rewriting: the Case of KorAP", inN. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, S. Piperidis (eds), *Proceedings of the Ninth International Conference on Language Resources and Evaluation* (LREC 2014), Istanbul / Paris, ELRA, 3817–3822.

Barbu Mititelu, V., D. Tufiș, E. Irimia, 2018, "The Reference Corpus of the Contemporary Romanian Language (CoRoLa)", in: N.Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, T. Tokunaga (eds), *Proceedings of the 11th International Conference on Language Resources and Evaluation* (LREC 2018), Miyazaki / Paris, 1178–1185.

Bevan, N., 1999, "Quality in use: Meeting user needs for quality", *The Journal of Systems and Software*, 49, Elsevier, 89–96.

Bevan, N., 2008, "Classifying and selecting UX and usability measures", in *Proceedings of the International Workshop on Meaningful Measures: Valid Useful User Experience Measurement*, 11, 13–18.

Bingel, J., N. Diewald, 2015, "KoralQuery – a General Corpus Query Protocol", *Proceedings of the Workshop on Innovative Corpus Query and Visualization Tools at NODALIDA 2015*, Vilnius, Lithuania, 1–5.

Bodmer, F., 1996, "Aspekte der Abfragekomponente von COSMAS-II", *LDV-INFO. Informationsschrift der Arbeitsstelle Linguistische Datenverarbeitung*, 8, 112–122.

Christ, O., 1994, "A modular and flexible architecture for an integrated corpus query system", in *Proceedings of the 3rd Conference on Computational Lexicography and Text Research* (COMPLEX 1994), July 7–10, Budapest, Hungary, 23–32.

Cristea, D., N. Diewald, G. Haja, C. Mărănduc, V. Barbu-Mititelu, M. Onofrei, 2019, "How to Find a Shining Needle in the Haystack. Querying CoRoLa: Solutions and Perspectives", in this volume.

Diewald, N., M. Hanl, E. Margaretha, J. Bingel, M. Kupietz, P. Banski, A. Witt, 2016, "KorAP architecture – Diving in the Deep Sea of Corpus Data", in: N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S.Piperidis (eds), *Proceedings of the 10th International Conference on Language Resources and Evaluation* (LREC 2016), May 23–28, Portorož, 3586–3591.

Fandrych, C., E. Frick, H. Hedeland, A. Iliash, D. Jettka, C. Meißner, T. Schmidt, F. Wallner, K. Weigert, S. Westpfahl, 2016, "User, who art thou? User Profiling for Oral Corpus Platforms", in: N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis (eds), *Proceedings of the 10th*

[17] https://matomo.org/

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

International Conference on Language Resources and Evaluation (LREC 2016), May 23–28, Portorož, 280–287.

Hardie, A., 2012, "CQPweb – combining power, flexibility and usability in a corpus analysis tool", *International Journal of Corpus Linguistics*, 17, 3, 380–409.

Heid, U., 2013, "Usability Tests und 'usability design' für die Entwicklung von Internetwörterbüchern", *Guest contribution to the 6th working meeting of the DFG network 'Internetlexikographie'*, 20th November, Mannheim, Germany.

ISO/TC 159/SC 4, 2006, *ISO 9241: Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts & Part 110: Dialogue principles*, International Organization for Standardization.

ISO/IEC JTC 1, 2000, *ISO 9126-1: Information technology – Software product quality – Part I: Quality Model*, International Organization for Standardization and International Electrotechnical Commission.

Kupietz, M., H. Lüngen, 2014, "Recent developments in DeReKo" in: N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, S. Piperidis (eds), *Proceedings of the 9th International Conference on Language Resources and Evaluation* (LREC 2014), Reykjavik / Paris, Iceland, 2378–2385.

Kupietz, M., N. Diewald, M. Hanl, E. Margaretha, 2016, "Möglichkeiten der Erforschung grammatischer Variation mithilfe von KorAP, der neuen Korpusanalyseplattform des IDS", in M. Konopka, A. Wöllstein (eds), *Grammatische Variation. Empirische Zugänge und theoretische Modellierung*, Berlin/Boston, De Gruyter, 319–329.

Kupietz, M., A. Witt, P. Bański, D. Tufiş, D. Cristea, T. Váradi, 2017, "EuReCo – Joining Forces for a European Reference Corpus as a sustainable base for cross-linguistic research", *Proceedings of the Workshop on Challenges in the Management of Large Corpora and Big Data and Natural Language Processing*, Mannheim, Germany, 15–19.

Kupietz, M., N. Diewald, E. Margaretha, F. Bodmer, H. Stallkamp, P. Harders, 2019, "Neues von KorAP", in: L. M. Eichinger, A. Plewnia (eds), *Neues vom Heutigen Deutsch. Empirisch – Methodisch – Theoretisch*, Berlin/Boston, De Gruyter, 345–349.

Oppermann, R., 2002, "User-Interface Design", in: H. H. Adelsberger, B. Collos, J. M. Pawlowski (eds), *Handbook on Information Technologies for Education and Training*, Springer, 233–248.

Przepiórkowski, A., Z. Krynicki, Ł. Dębowski, M. Woliński, D. Janus, P. Bański, 2004, "A search tool for corpora with positional tagsets and ambiguities", *Proceedings of the 4th International Conference on Language Resources and Evaluation* (LREC 2004), Lisbon, 1235–1238.

Söhn, J.-P., H. Zinsmeister, G. Rehm, 2008, "Requirements of a user-friendly, general-purpose corpus query interface", in: *Proceedings of the LREC 2008 Workshop Sustainability of Language Resources and Tools for Natural Language Processing*, May 31, Marrakech / Paris, 27–32.

Tidwell, J., 2006, *Designing Interfaces. Patterns for Interaction Design*, O'Reilly & Associates, Inc.

Tufiş, D., V. Barbu Mititelu, E. Irimia, Ş. D. Dumitrescu, T. Boroş, H. N. Teodorescu, D. Cristea, A. Scutelnicu, C. Bolea, A. Moruz, L. Pistol, 2015, "CoRoLa Starts Blooming – An Update on the Reference Corpus of Contemporary Romanian Language", *Proceedings of the 3rd Workshop on Challenges in the Management of Large Corpora* (CMLC-3), Mannheim Germany, 5–10.

Wall, L., T. Christiansen, R. L. Schwartz, S. Potter, 1996, *Programming Perl*, 2nd Edition, O'Reilly & Associates, Inc.

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)

BDD-A30403 © 2019 Editura Academiei
Provided by Diacronia.ro for IP 216.73.217.142 (2026-11-14 12:59:08 UTC)