DOI: https://doi.org/10.14232/wpcl.2022.7.2

Generating preview word maps in the DMW project

Kai-Uwe Carstensen

1. Introduction

The DMW project – funded by the Academy of Sciences and Arts of Northrhine-Westfalia, Germany, and involving the universities of Bonn, Münster, Paderborn and Siegen – is a long-term project (2016-2032) that aims at digitally collecting, analyzing, storing, presenting and preserving dialect data in the western part of Germany (mainly Northrhine-Westfalia).[7] To make this possible, we digitally record spoken data in ca. 800 different places, with four informants per place, each two of them aged '30-45' and '70+', respectively. We use a questionnaire of more than 600 succinct tasks, according to which the informants mostly have to answer open and yes/no-questions, to describe pictures or video scenes with a term, or to read sentences in their non-standard manner of speaking. These data are processed (cutting/segmentation and analysis of the sound files), and presented on dynamically generated maps (so-called *preview maps*) of a digital dialect atlas of the region.

We distinguish between two versions of the digital, dynamic DMW atlas according to the user types addressed: first, a standard, public version (aka "Speaking DMW"), which allows to view the distribution of dialectal variants in space, and to listen to the corresponding recordings (these are either the answers to some question on a "word map", or the Wenker sentences read out by the informants on "Wenker sentence maps"); second, an extended expert version with restricted Shibboleth access, which will also portray the distribution of ca. 1200 dialect phenomena variants (similar to other dialect atlases). Both versions are based on visually presenting the variants in categorized form ("taxates" in Goebl 2010's terminology), yet the types/taxates are determined *automatically* in the former according to the scheme described below, *intellectually/manually* in the latter. In the final phase of the project, some of the phenomena-related data will be evaluated and published as classic, annotated dialect maps of the region.

Preview maps[8] systematically depart from classic dialect maps in at least the following respects. Preview maps are inherently digital and dynamic, i.e., they are instantaneous projections of actual analyzed data generated automatically on-line as a result of a user query. By offering various kinds of selection and presentation options, they are interactive and can be used for visual exploration of *all available* dialect data even by lay people from early on. Yet they lack classic, evaluation-based dialectal annotations ((core) dialect areas, isoglosses etc.) or fully theory-driven data clusters and variant types. In contrast to that, *classic dialect maps* are static (print-oriented), non-interactive, non-explorative maps constructed manually by experts for experts. They are based on time-consuming *evaluation* of – mostly selected – data and portray them as filtered by intellectual considerations and with corresponding

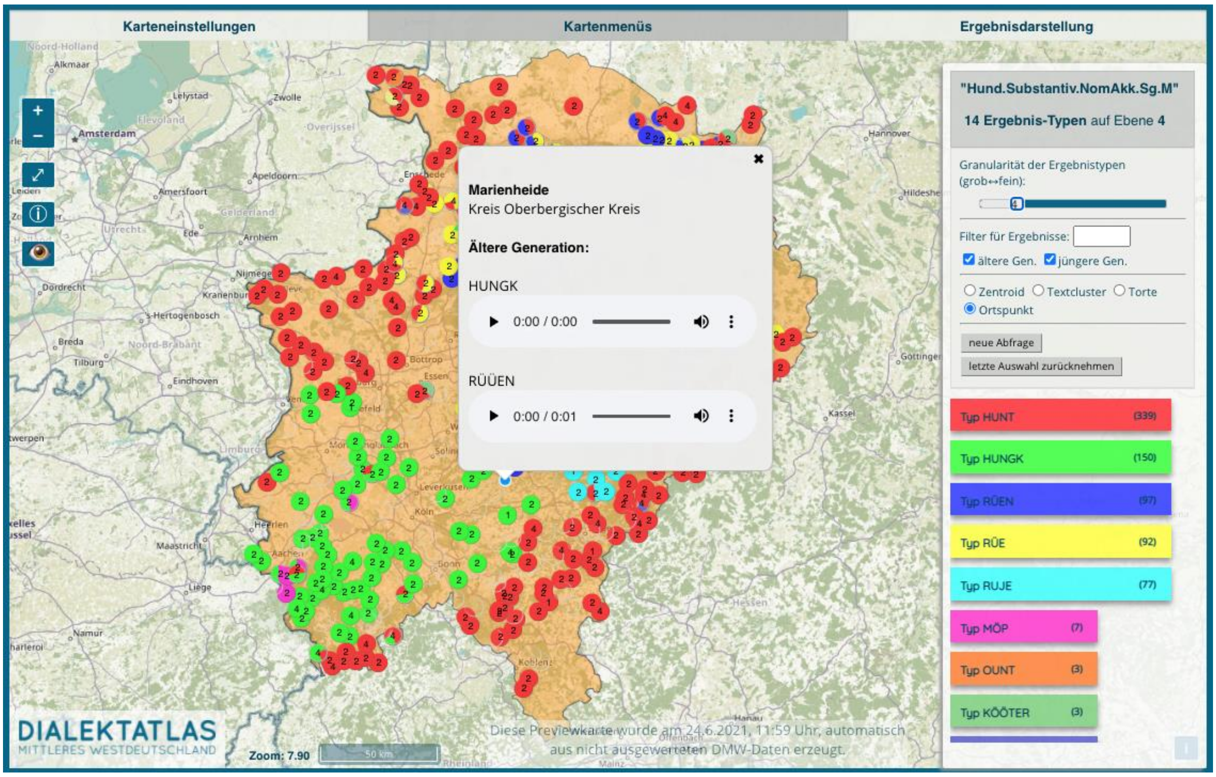
[7] See Spiekermann et al. (2016ff) for general information about the project; for an overview of the workflow, see Carstensen et al. (2020).

[8] The preview word maps as presented here have been introduced in the SiSAL project (Solau-Riebel – Vogel 2013-2016), albeit in a much smaller scale, and with much more manual data preparation.

dialectological annotations. Apart from different possibilities and habits of information presentation in modern times (non-print, user-friendly, computer-generated, interactive, explorative, conforming to GUI principles), an important disadvantage is the expectable time lag for classic dialect maps, because they require complete data sets for the selected phenomena. This makes preview maps the primary (and for the most part, only) option for the DMW, given the size of the project (number of informants, questions, phenomena) and the maxim to use digital technology throughout.

Irrespective of the classic/non-classic contrast regarding intellectual evaluation, current preview maps are different from most, if not all, computational approaches to dialectology in dialectometrics or geolinguistics (see Lameli et al. 2010 for an overview) in that they do *not* show statistical analyses of more than one item or feature ("global" analyses in Goebl's terms). Instead, they are intended to easily identify aspects of variant distribution in space for *particular* analyses/items, given massive data variation. The present paper describes generating *preview word maps* based on the transcriptions of uttered words like the one shown in Figure 1, to be distinguished from *preview phenomenon maps* showing the variants derived from theory-based analyses (of some phonetic, morphological, lexical, or syntactic aspect of uttered words) that will be developed later.

Figure 1 *Preview map of* 'Hund'



Marienheide
Kreis Oberbergischer Kreis

Ältere Generation:

HUNGK
▶ 0:00 / 0:00

RÜÜEN
▶ 0:00 / 0:01

"Hund.Substantiv.NomAkk.Sg.M"

14 Ergebnis-Typen auf Ebene **4**

Granularität der Ergebnistypen (grob↔fein): 4

Filter für Ergebnisse:
☑ ältere Gen. ☑ jüngere Gen.

○ Zentroid ○ Textcluster ○ Torte ● Ortspunkt

neue Abfrage
letzte Auswahl zurücknehmen

🟥 Typ HUNT	(339)
🟩 Typ HUNGK	(150)
🟦 Typ RÜEN	(97)
🟨 Typ RÜE	(92)
Typ RUJE	(77)
Typ MÖP	(7)
🟧 Typ OUNT	(3)
Typ KÖÖTER	(3)

Diese Previewkarte wurde am 24.6.2021, 11:59 Uhr, automatisch aus nicht ausgewerteten DMW-Daten erzeugt.

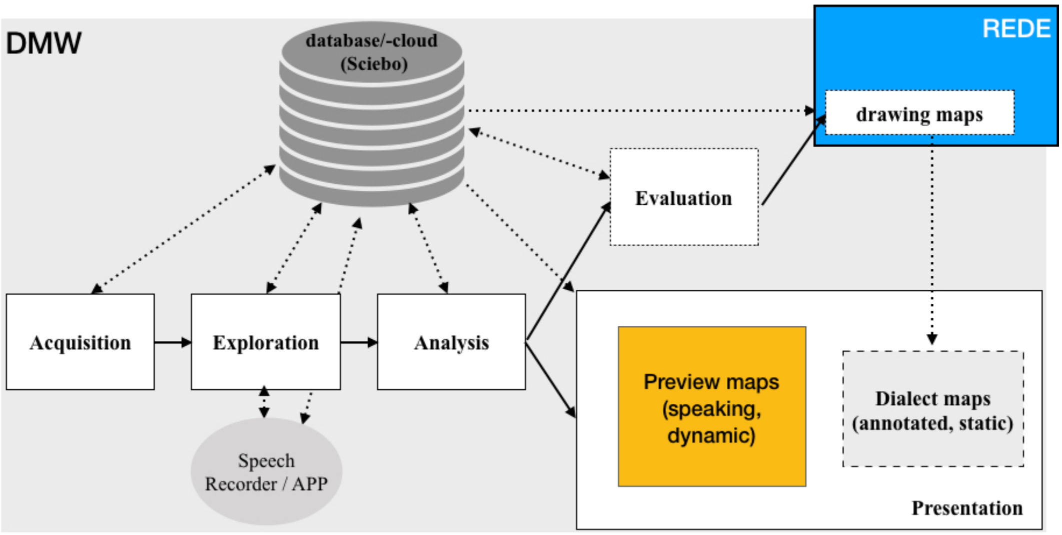
At the beginning, we faced a number of challenges to be met for a success of the DMW project: efficient, error-less, off-line digital fieldwork/interviews/recordings ("exploration"); efficient, error-avoiding, computer-assisted high-quality transcription; effective automatic visualization of dialect data using up-to-date web technology. To exemplify this with current numbers: as of June, 1st, 2022, we have cut and transcribed ~590.000 spoken words of more than 550 explored places, and in the presentation of a word map, the number of variants of a single word can easily reach 100, or even 200 (of already generalized IPA transcripts). Figure 1 shows how our preview maps can give a rough indication of data distribution in spite of massive data variation (in this case, more than 60 already generalized variant types for the word *Hund* (‘dog’), reduced to a display of 14 types) by choosing certain clustering and visualization options.

In following, I will first describe the work and data flow in the DMW project, with a focus on the computer-assisted transcription (aspects of the computer-assisted exploration are further discussed in Gehrke et al. 2020), introducing our literal “popular” transcription (POP). After that, I will give a detailed account of our POP-based algorithm for handling massive dialect data to achieve effective visualization. Finally, I will elaborate on the technical aspects of preview word map generation in the DMW project for the standard version, also showing the possibilities of its interactive and explorative use.

2. Work and data flow in the DMW project

Basically, technical aspects in the DMW project can be described as entering, retrieving and modifying data of our central database (apart from storage of audio data in the scientific cloud Sciebo). From early on, handling DMW data was conceived as human-computer interaction via (graphical user) interfaces: the *acquisition interface* for handling contact (action) information for places to be explored; the *exploration interface* for uploading personal information and (meta-)data of the interviews; the *analysis interface* as a work area to handle informant data (cutting sound files, transcribing utterances, and handling phenomena aspects); the *map (presentation) interface* presenting preview maps for selected queries of some user. Figure 2 gives an overview of the simplified work and data flow structure of the DMW project.

Figure 2 *Global view of work and data flow in the DMW project*



Note that production/drawing of classic, static dialect maps is planned but not yet done, hence the different graphical appearance as dashed components. It requires both a stage of intellectual evaluation of the analyzed data, and manual production of maps (which will be done with the map drawing tools provided by REDE, see Schmidt et al. 2008ff).

Over the years, additional functionalities and tools requiring database interactions were developed. One of them is the *IPA repair tool* that allows to easily spot transcription errors by listing selected data, and to quickly correct them via the offered shortcuts to the corresponding (parts of) the analysis interface (see below for more information). While the interfaces were originally viewed as wholistic web applications covering the central aspects of our work flow, we therefore rather built a structured internal web presentation for our project (with corresponding work flow areas) in which the interfaces, but also other web-based functionalities, were placed accordingly.

2.1 Acquisition

In general, we require exploration places to be a subset of the classic Wenker places constrained by a certain range of inhabitants (500 to 8000) and some scheme of distribution: we use a raster overlay of our exploration area to arrive at an equally distributed selection of places to be explored, each partner university being responsible for a certain part. The acquisition interface involves a WebGIS application showing area, raster and Wenker places, as well as filterable layered information about the places (aspects of the contact status and (partial) exploration status of age groups to be explored). A click on a place accesses the acquisition interface proper which allows to enter and view contact data as well as time stamped acquisition actions (contacting persons via phone or mail, placing advertisements, distribution of informant questionnaires etc.). Once established, this interface provided a remarkable facilitation of acquisition planning and coordination.

2.2 Exploration

For the digital explorations, we use the SpeechRecorder® app (Draxler and Jänsch 2004, 2019) with which the answers of the informants to the questions of the exploration questionnaire are automatically cut and systematically stored – an enormous saving of time, and hence, man power, for the subsequent analysis stage. Unfortunately, the SpeechRecorder was designed for use as a laboratory software, which does not guarantee uniqueness of identifiers or easy adjustments of the built-in questionnaire when used distributedly in different locations, on various computers, and in the field. We needed to come up with a quick solution for these problems since explorations were supposed to start immediately.

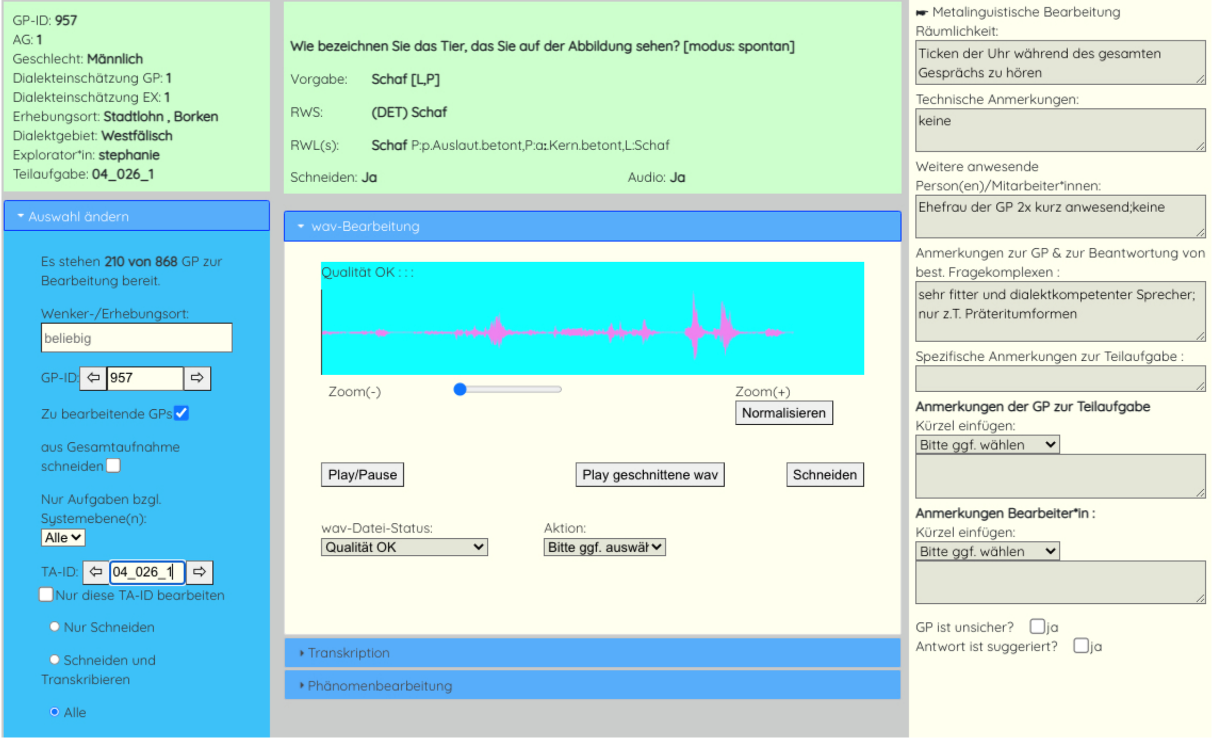
We opted for dedicated semi-automatic work flows using python scripts to handle the necessary organizational data structures. The first python script transforms the original exploration questionnaire into the XML format required by the SpeechRecorder. It is applied semi-automatically every time the questionnaire is modified (which happened quite often in the beginning of the project). Then, before an exploration, each explorator of university with id U requests a "SpeechRecorder project" for an informant with –a unique– identifier ID. This automatically fires the second python script, creating unique project files (using U and ID) and bundling them into a downloadable zipped SpeechRecorder project importable by the SpeechRecorder. U and ID, together with the question numbers of the "SpeechRecorder Script", are used to name the corresponding audio files of the answers. After the exploration, the folder containing audio files is uploaded into the cloud, accessible for use in the analysis and map interfaces.

The exploration interface is used by the explorator to store information noted in the *exploration protocol* (remarks about the setting –ambience, technicalities, other persons present, characteristics of the informant– and about aspects of specific answers). Even before the exploration, it is also used to enter all information about the informant (a prerequisite for the exploration).

2.3 Analysis

Analysis in the DMW project roughly divides into treating sound aspects of an answer, transcribing the relevant answer word(s), and handling phenomena information. The analysis interface (see Figure 3) allows to make certain selections, e.g., of place, informant, task, analysis part to deal with (in the blue area) or to view relevant (meta) information, for example, about informant, exploration setting, or task/analysis like question asked, and words to be analyzed (in the green area).[9] While wav-handling and transcription (see below) are indispensable for the generation of preview maps, phenomena handling will be performed in later stages of the project. For aspects of metalinguistic information (handling) see Gehrke et al. (2020).

Figure 3 *Analysis interface*



GP-ID: **957**
AG: **1**
Geschlecht: **Männlich**
Dialekteinschätzung GP: **1**
Dialekteinschätzung EX: **1**
Erhebungsort: **Stadtlohn , Borken**
Dialektgebiet: **Westfälisch**
Explorator*in: **stephanie**
Teilaufgabe: **04_026_1**

Wie bezeichnen Sie das Tier, das Sie auf der Abbildung sehen? [modus: spontan]

Vorgabe: **Schaf [L,P]**
RWS: **(DET) Schaf**
RWL(s): **Schaf** P:p.Auslaut.betont,P:aː.Kern.betont,L:Schaf
Schneiden: **Ja** Audio: **Ja**

Es stehen **210 von 868** GP zur Bearbeitung bereit.
Wenker-/Erhebungsort: beliebig
GP-ID: 957
Zu bearbeitende GPs ☑
aus Gesamtaufnahme schneiden □
Nur Aufgaben bzgl. Systemebene(n): Alle
TA-ID: 04_026_1
□ Nur diese TA-ID bearbeiten
○ Nur Schneiden
○ Schneiden und Transkribieren
● Alle

Qualität OK : : :
wav-Datei-Status: Qualität OK
Aktion: Bitte ggf. auswäł

Räumlichkeit: Ticken der Uhr während des gesamten Gesprächs zu hören
Technische Anmerkungen: keine
Weitere anwesende Person(en)/Mitarbeiter*innen: Ehefrau der GP 2x kurz anwesend;keine
Anmerkungen zur GP & zur Beantwortung von best. Fragekomplexen : sehr fitter und dialektkompetenter Sprecher; nur z.T. Präteritumformen
Spezifische Anmerkungen zur Teilaufgabe :
Anmerkungen der GP zur Teilaufgabe
Kürzel einfügen: Bitte ggf. wählen
Anmerkungen Bearbeiter*in :
Kürzel einfügen: Bitte ggf. wählen
GP ist unsicher? □ja
Antwort ist suggeriert? □ja

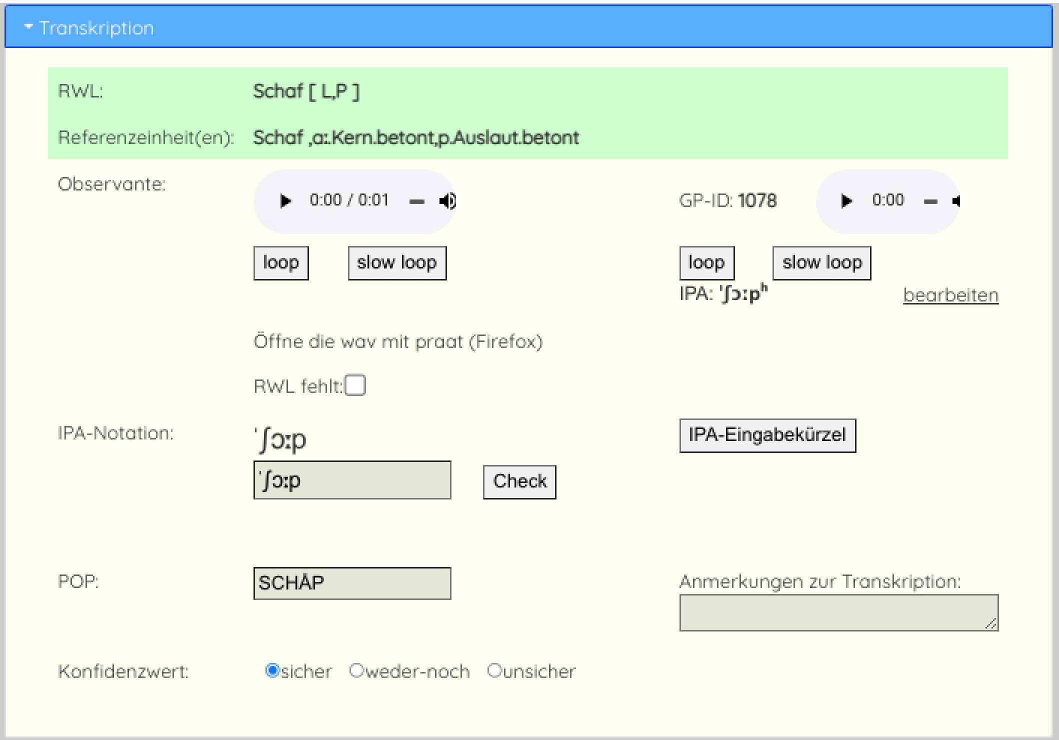
Sound-related analysis (in section “wav-Bearbeitung” of Figure 3) involves cutting audio files for audio presentations on preview maps (for users to be able to hear the non-standard pronunciation of a word, uttered as part of an answer to some question of the questionnaire). We specify the range of words to cut as so-called RWS (**R**elevant **W**ord(s) for cutting (=**S**chneiden)) and also present the information about the words to be analyzed as so-called RWL (**R**elevant **W**ord(s) for **L**inguistic analysis). For example, we might elicit a prepositional phrase *in the barn* with a question *Where does the farmer store the hay?*. Although we could be interested in various phenomena on distinct linguistic levels addressing different RWLs (say, lexical and phonetic aspects of RWLs *in* and *barn*, syntactic aspects of *in the* or *the barn*), and might want to present some of them audibly, we decided to just cut *one* audio file (containing the span of words with variants of all RWLs, in this case, *in the barn*) for each (sub)task as RWS, for economy reasons. Apart from that, the RWS of a single-noun RWL may be specified as having to include

[9] Task handling is restricted, however. To prevent certain analysis artifacts (“analyzer isoglosses”), each analyzing person is exclusively assigned a number of tasks by the coordinator.

the determiner of the noun, if uttered (see Figure 3). Sound-related analysis then means to perform some action (cutting the RWS, optionally improving sound quality), and to store relevant information (e.g., about the quality/status of the audio file or the answer). Note that an answer might be difficult to hear or identify (in case of multiple speakers or multiple different answers), or be lacking for different reasons (no or wrong answer given, question not asked due to unfinished interview, or question not in questionnaire at interview time).

The transcription part of the analysis interface cycles through the RWLs of the current task/question presenting the corresponding observed variants (“observants”) for phonetic analysis in each case. Transcription is different from simply entering IPA characters (say, via the keyboard) given the audio file, and is rather realized as a computer-assisted process involving a dedicated set of functionalities, with *IPA transcription tool* (ITT) referring to the area of the analysis interface in which this happens (see Figure 4). The ITT allows to loop both through the current observant (even in reduced speed) and the observant of the other informant of that place (both helps to identify the specifics of the pronunciation). Phenomenon-related information is displayed for the analyzing person to know what to listen to in particular. Transcription is always *computed* given the input into the corresponding field (i.e., transcription is semi-automatic), as detailed below. We use a literal “popular” transcription (so-called POP) as a readable version of an observant to be displayed on preview maps (see below). These POPs are computed *automatically* from the computed IPA of the input.

Figure 4 *The IPA transcription tool (ITT) as part of the analysis interface*



Transkription

RWL: Schaf [L,P]

Referenzeinheit(en): Schaf ,a:.Kern.betont,p.Auslaut.betont

Observante: 0:00 / 0:01

GP-ID: 1078 0:00

IPA: 'ʃɔ:pʰ

Öffne die wav mit praat (Firefox)

RWL fehlt: □

IPA-Notation: 'ʃɔ:p

'ʃɔ:p

POP: SCHÅP

Anmerkungen zur Transkription:

Konfidenzwert: ● sicher ○ weder-noch ○ unsicher

The phenomena handling part of the analysis interface will be used to deal with the ∼1200 phenomena to be investigated in the DMW project. Phenomena typically address *parts* of RWLs that have to be identified given the RWL transcript (or the RWS, mostly in the case of syntactic phenomena of nontranscribed observants). After analysis, the variants of some phenomenon's reference entity will be displayable on corresponding maps in the map interface. This part is still under construction, however.

2.4 Presentation

The results of the analysis of RWLs, as well as the RWSes, are presented on dynamic, unevaluated preview maps as "Speaking DMW" (also collectively called "Atlaskarten"/"Atlas maps").[10] We distinguish two versions of the map interface. First, the standard version for the presentation of word maps (visually displaying the variants of words), and of the Wenker sentences (both with a "click-and-hear" facility). Second, the "expert" version (only accessible via restricted, universitary Shibboleth access) showing variants of phenomenon-related RWL parts.[11] By definition, the preview maps involve no further evaluation, and can therefore be generated as soon as there are analyzed data. Correspondingly, the preview maps directly reflect the progress of the project. Only for the later stages of the project, we plan to evaluate (selected) data in order to present classic static annotated dialect maps for the explored region. Such maps will be constructed with the facilities provided by REDE (Schmidt et al. 2008ff).

3. Transcription in the DMW

3.1 Phonetic transcription in the ITT

Input to the ITT can be either letters, that is, characters of the set [aeioubdfghjklmnprstvwxyz](basic IPA symbols). Any other IPA (diacritic) symbol is specified differently, often in more than one way. In general, we only consider a selected set of IPA symbols (but note that we also code syllable boundaries, ambi-syllabic consonants, primary/secondary stress, and syllabic consonants).[12] To code them, certain textual entries can be made to specify IPA symbols ("sch" → [ʃ], "tsch" → [tʃ], "t-sch" → [t͡ʃ̣]), double characters either specify long phones ("aa" → [aː]) or ambi-syllabic consonants ("tt" → [ṭ]), and certain added characters specify slight variants ("e#"- > [ə], "r#" → [ɹ]). As an alternative, SAMPA (see Wells 1997) character(sequence)s can be used ("S" → [ʃ], "E" → [ɛ], "6" → [ɐ]), also in textual combinations ("t-S" → [t͡ʃ̣]). In the ITT, there is a help button listing all available textual, SAMPA, or combined character sequences coding IPA symbols.

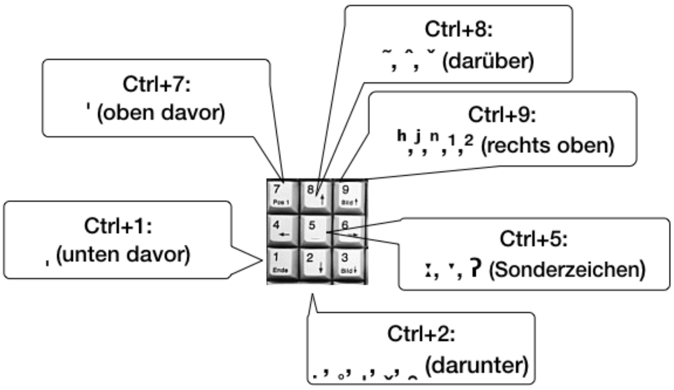
While they can be specified with SAMPA (e.g., "t_h" → [t^h]), there is an additional entry scheme especially for diacritics, based on the position of digits on the numeric keypad (see Figure 5): By Ctrl-entering a digit, the diacritics associated with digit's position relative to "5" are offered for selection (for example, the nasality diacritic is "directly above" a consonant phone, hence the use of 8, directly above 5, to access the corresponding set of diacritics). By Ctrl-entering other characters, phonetic variants of the character are presented for selection (for example, "Ctrl + ö": [øœɜ], "Ctrl + r": [ɹɻɽɾʀʁ]). Combined with an auto-completion mechanism that uses the IPA entered so far and the known IPA variants for the RWL (so far, from the database), the ITT therefore offers a quick and non-annoying way of entering IPA symbols for transcription.

[10] This is the external terminology for what is project-internally called "map interface".

[11] In addition to that, simple dialectometric reference point maps (see Goebl 2010) can be generated.

[12] (Non-)syllabicity is coded redundantly and automatically, as are affrication arcs.

Figure 5 *Ctrl-entering diacritics (from German tutorial)*



The *quality* of the transcriptions is secured on many levels, starting with the slow-loop option to listen to the observant, and ending with a global transcription review process whose start is scheduled in 2025. In between are further aspects of quality assurance.

Most importantly, a red “Check” button has to be pressed obligatorily before transcript storage. In that case, a number of tests are performed to check for obvious or possible transcription errors. *Actual errors* (clearly wrong input) lead to corresponding messages, while preventing transcript storage. To detect them, this requires some effort using regular expressions. For example, to discover missing syllable boundaries or stress markers, violations of phonotactic rules in a syllable (based on the sonority hierarchy of the involved phones) have to be uncovered. *Possible (typing) errors* lead to warnings (e.g., if rarely used SAMPA uppercase letters like A, M or V have been entered). Only in situations without actual errors the button turns green, and storing is possible.

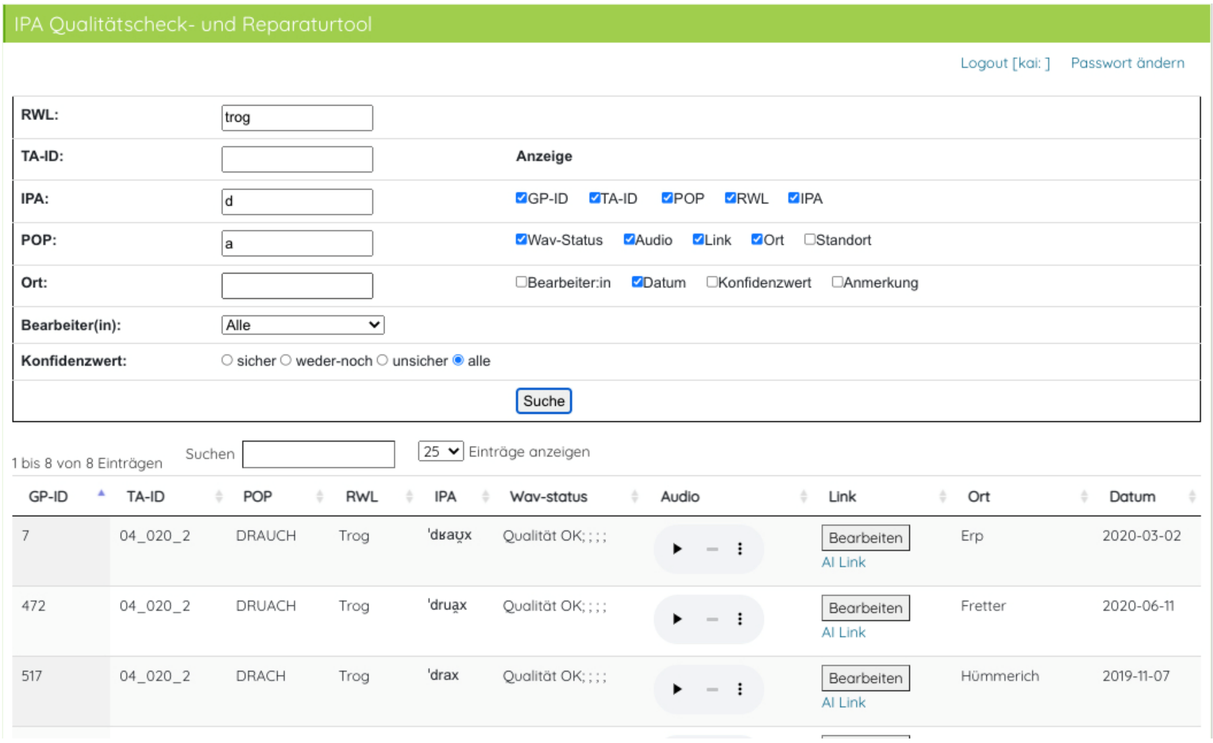
Each time the analyzing person hits Return, performs Ctrl-digit entering, or hits the Check button, the IPA is computed and displayed. At the same time, the literal POP transcription (see below) is computed. Both displays can be used as a feedback about the correctness of the intended transcription.[13]

Finally, the analyzing person has to specify her confidence with a three-valued option (confident, inconfident, in-between). Only then can the transcription be stored.

Outside the ITT, quality can be checked with the preview maps: if there is an outlier in some homogeneous distribution of variants, this is at least a hint to examine the corresponding transcription of the outlier. Likewise, the variants on the preview maps can be filtered by confidence value to identify unconfident transcripts. For both cases, there is an *IPA Repair tool* that offers the possibility to quickly enter an RWL and (at least either) a part of the IPA or POP, or of the place name. As an example, Figure 6 shows the query “list data for ‘Trog’, where IPA contains ‘d’ and POP contains ‘A’” and part of the table of the corresponding results. For easy inspection, the result section is sortable by a click on one of the column titles, and can be filtered by using the full-text search field. The observants can easily be listened to, and links to the corresponding instantiations of the ITT – and also the whole analysis interface– for immediate modification (“Bearbeiten”)/correction are provided.

[13] It should be obvious that transcription in the DMW project requires intensive training. We have Sharepoint pages introducing transcription basics (codes, use of ITT, DMW-specific transcription rules etc.). Above all, there is an IPA transcription *training tool* that mimics the ITT and allows to perform transcription with selected tasks and test data.

Figure 6 *The IPA Repair tool of the DMW project*



IPA Qualitätscheck- und Reparaturtool

Logout [kai:] Passwort ändern

RWL:	trog	
TA-ID:		Anzeige
IPA:	d	☑GP-ID ☑TA-ID ☑POP ☑RWL ☑IPA
POP:	a	☑Wav-Status ☑Audio ☑Link ☑Ort □Standort
Ort:		□Bearbeiter:in ☑Datum □Konfidenzwert □Anmerkung
Bearbeiter(in):	Alle	
Konfidenzwert:	○ sicher ○ weder-noch ○ unsicher ● alle	
	Suche	

1 bis 8 von 8 Einträgen Suchen 25 Einträge anzeigen

GP-ID	TA-ID	POP	RWL	IPA	Wav-status	Audio	Link	Ort	Datum
7	04_020_2	DRAUCH	Trog	'dʁaʊ̯x	Qualität OK; ; ; ;		Bearbeiten AI Link	Erp	2020-03-02
472	04_020_2	DRUACH	Trog	'drua̯x	Qualität OK; ; ; ;		Bearbeiten AI Link	Fretter	2020-06-11
517	04_020_2	DRACH	Trog	'drax	Qualität OK; ; ; ;		Bearbeiten AI Link	Hümmerich	2019-11-07

3.2 Literal transcription: POPs

Especially due to the many non-expert users of the DMW system, a transliteration of the IPA transcripts of utterances to their readable textual representation is mandatory at least for the word maps. We decided to use the *Hamburgian transcription conventions* ("Hamburger Transkriptionskonventionen", HTC, see Bieberstedt et al. 2016) for this purpose, whose maxim is to "[achieve] an exact reproduction of pronunciation as much as possible, while providing good readability, ensured by using the standard alphabet and a minimum number of special characters (only å)" (Bieberstedt et al. 2016:421, my translation). We call the resulting literal transcripts *POPs* (from "popular"), and write them in uppercase letters.

It is a characteristic of literal transcription that diacritics and prosodic markers are left out, and that certain non-ASCII symbol(sequence)s are systematically mapped on certain character(sequence)s. Examples of the HTC are [ɛ] →<E>, [ɔ] →<O>, [ɔː] →<Å>, [ɔːː] →<ÅÅ>, [ɔi̯] →<EU>, [ʃ] →<SCH>, [d͡ʒ] →<DJ>, [ŋ] →<NG>. Other rules determine how to transcribe syllabic consonants K (as <K'>) or how to mark relevant syllable boundaries (with <->). A large part of the HTC is concerned with the textual marking of vowel length (e.g., when to mark shortness with a double consonant: [kɪn̩ɐ] →<KINNER>, [lʊft] →<LUFT>/*<LUFFT>, [ʃvɪmt] → *<SCHWIMT>/<SCHWIMMT>). On the whole, the HTC provide a good standard for the task of literal transcription, and have been implemented in our system (although automatic realization is less than perfect). It must be noted that the HTC, as our POP conventions, have three main disadvantages, however.

First, they were designed especially for the Hamburgian Low German, while half of Northrhine-Westfalia covers Middle German areas showing regional phonetic phenomena not covered by the conventions, for example, the so-called “Tonakzent” (phonemically relevant intonation), the voiced velar fricative ([ɣ]), voiced labio-velar approximants ([w]), or different variants of /r/.

Second, their intended use was for directly/manually transcribing spoken dialect, the focus being on the identification of the words uttered. Because of that, they include rules that guarantee recognition of the lexeme, not only of the phonation: for example, while [ts] could be transcribed as <TS>, it has to be transcribed as <Z> at the beginning (as in [t͡saːn] → *Zahn* (tooth)), and as <TZ> at the end of the word/syllable ([kat͡s] → <KATZ>). This standard-language oriented transcription is most obvious in the double-consonant rule following the “morphological principle” in orthography: a consonant is written twice because the lexeme shows this duplication (for example, [brɛnt] is transcribed as <BRENNT>, not as <BRENT>, because the verb is *brennen* (burn)).[14] Similarly, the final unstressed central vowel [ɐ] is to be transcribed as <ER> (but as <A> after [ɛ] or [ee]). Since standard words are the starting points of word maps (“Holz”, “Zahn”, “Feuer”), all this would not have been necessary in the DMW. On the other hand, and thereby different from standard language, the conventions expect an additional hyphen in the case of syllable gaps as in *inadequate*: <IN-ÄDIKWET>. Unlike handling [ts], final [k] is not to be transcribed as <CK> (as in *dick* and *Stock*), but as <KK>. Furthermore, while [ɛ] is mapped to <E> by default, [ɛi] is to be transcribed as <ÄI> (so-called “Hamburger Diphthongierung”). This mixture of rules, adaptions to standard writing, and exceptions, complicate the automatic/ digital use of the conventions, which raises the possibility of errors and might slow down performance.[15]

The third disadvantage concerns the use of <E> and <O> for [ɛ] and [ɔ], respectively. Again, while this may be sufficient for literal transcription (and is also motivated by easy recognition of the words in their standard writing), it might be regarded as unsatisfactory in a dialect atlas due to the elimination of interesting variation.[16] For this reason, the map interface offers an option to select “narrow” literal transcription (“lautnahe Transkription”) in which [ɛ] is mapped to <Ä> (not to <E>), [ɜ]/[œ] to <Œ> (not to <Ö>), and [ɔ] to <Å> (which codes [ɔː] by default; therefore, not to <O>). Further candidates for narrow transcription could be [ʒ] (to be transcribed as <J> according to the HTC, and hence confusable with the transcription of [j]), [ɣ] (not as <CH>), or characteristic variants of /r/.

[14] The adaptation according to the morphological principle is (and can be) done only for the current RWL (and its standard form). Therefore, the variant DACHJESCHOS of *Dachboden* only has one final “s”, although JESCHOS corresponds to *Geschoss* with “ss”. The missing implicit human-in-the-loop, needed for aspects like these, could be regarded as another disadvantage of the HTC for our automatic literal transcription. Note, however, that the authors of the HTC are open to adaptations and modifications of the conventions in principle (Jürgen Ruge, pers. comm., see also Ruge 2019).

[15] In some cases, the HTC are inconsistent (in their goals) or incomplete: non-initial [ts] would lead to <WIRTZHAUS>, <MIITZHAUS>, <HOLTZ> for corresponding variants of *Wirtshaus*, *Mietshaus*, and *Holz*.

[16] Especially because phonation is indicated via POPs in preview maps, and because the IPA transcripts are not shown in the standard (lay)version of the map interface.

4. Handling massive variation in the DMW data for the presentation of preview maps: the DMW POP algorithm (DPA)

4.1 The challenge

For typical words, dozens of variants (IPA- types) can easily be observed (e.g., for the item *Kette* ('chain')) in our data, especially as our region comprises both Low German and High German parts. The situation gets worse if there are lexical variants of a word (for *Mädchen* ('girl'): added variants of *Deern*, *Wicht* and *Lüüt*), in which case the number of IPA types can go into the hundreds. Although literal transcription abstracts from specific pronunciation aspects, the number of POPs for an item/RWL still often is very high.

This presents several problems for a system that aims at generating user-friendly visualizations. Above all, there is the sheer *size* of data variance: Even with a smaller set of POP types, the number of signifiers is still too large for effective visualization of the data. Without further intellectual categorization of the signifiers, it is therefore necessary to add an automatic clustering step to prevent cluttering the preview maps with symbols. Then there is the problem of *symbolization* of signifying, say, more than 3-5 variant types. Existing dialect data visualization systems use specific symbols, color, or a mixture of both for that purpose. Especially when using symbols, they run into the problem of visual confusion, however, especially if the symbols are similar in some respect (e.g. circles containing lines representing different hachures).

The prime challenge therefore was to find a scheme of automatic categorization (a "variant/POP type mechanism") that replaces intellectual classification as effectively as possible. This excludes simple Levenshtein-based mechanisms as they are linguistically neither motivated nor constrained. Development time considerations forbade diving into the field of possible improvements of such methods (see Wieling et al. 2011). Run time considerations ruled out too complex computations, and user interaction requirements called for a non-continuous, qualitative and easily understandable (usually hierarchical) category structure. In general, this excludes quantitative clustering methods that map variation to a graded structure in which categories may not identifiable ("named"), and which has to be actively manipulated by the user for him/her to gradually select the level of detail for a presentation (as described in Haimerl 2005:544). Our *DMW-POP-algorithm (DPA)* is therefore different and uses a qualitative, multi-layered Soundex-inspired algorithm, which is described in the following.

4.2 Soundex

The Soundex algorithm (see, for example, Wilz 2005) was invented to allow for *phonetic search*, i.e., a phonetically motivated procedure that finds matches for a database lookup (especially of names) despite nonidentical search terms (for cases like Möller/Moeller, Schmidt/Schmitt/Schmied). In general, it operates by mapping phonetically similar names onto common index terms so that lookup reduces to matching index terms, at least for a first selection of probable candidates. More specifically, (one variant of) the algorithm consists of the following steps (simplified):

1. Retain the first letter of the word; map vowel-like letters to zero (A,E,I,O,U,Y,H,W → 0);
2. Map consonant(sequence)s on certain digits (B,F,P,V → 1; C,G,J,K,Q,S,X,Z → 2; D,T → 3; l → 4; M,N → 5; R → 6);

3. Reduce adjacent same digits to one digit (this also holds for the digit of the first letter), then delete zeros;
4. Truncate after three digits; for a shorter word, add zeros until the index term has four characters.

Accordingly, “Wikipedia” is mapped to “W213” (as is, for example, “Wakepod”). There are a number of variants of the Soundex algorithm (e.g., *Metaphone* for English, and “Kölner Verfahren/Phonetik” for German, see Wilz 2005), which mostly use a more sophisticated scheme of the mapping and/or allow for more than three digits (even one for the first letter) to code the index term. Note that one can distinguish between “technical” aspects (e.g., the structure of the resulting index term) and “content” aspects (different kinds of phonetic rules used) of the schemes.

4.3 DPA: technical aspects

The DPA departs in several respects from the basic Soundex scheme. First, it starts with approximate phonetic representations ((partial) literal transcripts of IPA) and, according to this, does not map *words* to approximate phonetic representations of words. Correspondingly, there is no need for sophisticated algorithms accounting for the different textual realizations of phones. Instead, the DPA maps approximate phonetic representations to *generalized* approximate phonetic representations. In particular, this means that the input to the indexing scheme, on the one hand, is an *extended* POP: it may still contain some IPA symbols ([ʃʒŋœɣxχ]) to ease processing (e.g., handling fricative symbols instead of <(s)ch>), and for narrow transcription. On the other hand, it is a *partial* POP because the conventions have not yet been fully applied.

Second, unlike Soundex, the DPA does not use a fixed, one-level set of principles for indexing. Instead, it applies the indexing on six levels at the same time, each with a different set of principles resulting in coding different *levels of generalization* of a POP, the levels being ordered from least general (i.e., only containing POPs) to most general. In the Javascript implementation, this results in an array *Types*, with Types[0] being the (final) POP, and Types[5] being the most general index term. When applied to a set of POPs, the POPs therefore are automatically clustered on corresponding *levels of granularity* of the set, from least granular (most general) to most granular (the level of POPs).

The third departure from the Soundex scheme is the use of a secondary sub-element mapping (on levels $1 < i < 5$). If in an index term set one term is mapped to XXX, and another to YYYXXX or XXXYYYY (on the same level), the second is remapped to XXX. This allows to capture the fact that certain words are sometimes uttered as parts of compounds, and that it makes sense to group them (for example, HEAFSLOOF and LOOF, and HEAPSLAUP and LAUP, respectively). Technically, this requires some checks to at least guarantee a high correctness probability of such automatically determined subpart relationship. For example, too short parts (for a level) have to be excluded, as well as accidental parts, so that, on level 4 for *Laub* (‘leaves’), the index terms of LOOF-AFFAL, HEASLOOF, LOOFHOOF, HEAFSLOOF, HERFSLOOF, LOOFTEPPISCH are mapped correctly to the index term of LOOF. Overall, this mapping scheme reduces cluster numbers per level significantly, without pruning the index terms arbitrarily.

Internally, this multi-layered indexing scheme requires considerable computation effort (sorting, remapping, counting, establishing information structures, relating clusters to their sets of POPs on five levels). This is why computation is restricted (at the moment, to less than 150 types), and the user is asked to set a filter to arrive at a smaller set of types (this might be changed, however).[17]

It is important to note that this granularity mechanism with its automatic determination of clusters, types, and symbolizations is the central, technical characteristic of the DPA, allowing for user-friendly, explorative, selective viewing of variants as the non-manual solution of the above cluttering problem.[18] In the map interface, variants will be presented according to the granularity level chosen, and can even be restricted to a cluster on that level.

4.4 DPA: content aspects

The content aspects of the DPA consist of applying some version of the indexing scheme on each level of granularity. They can be arbitrarily chosen, and are only constrained by the rule that with every level of granularity above POP level, the principles must become more general. In the current implementation, the following steps of increasing generality are used:

1. Abstract from specific vowels, build vowel categories, here: frontal (“vorne”), medial (“zentral”), back (“hinten”) vowels by “v/z/h”, correspondingly; mark identity (textual length marking) of following vowel in a vowel character sequence by “ = ” (else mark following vowel by “_”);
2. Abstract from gap marking and vowel length (leave out “-“ and “ = ”) and condense consonants CC to C.
3. Abstract from vowel quality/category by simply noting “_”; generalize plosive minimal pairs (P/B by “P”, T/D by “T”, G/K by “K”), and velar/uvular fricatives by “Ç”.
4. Abstract from vowel change (leave a single “_” for vowel(sequence)s)
5. Only mark first POP character.[19]

Table 1 lists the different types of codes collected and generated for the word *Ei* (‘egg’).

[17] Filtering is possible via selecting a cluster on some level (thereby restricting the set of variants to those of that cluster), or by textually filtering the POPs via some regular expression to restrict the current variant set.

[18] While the content aspects are no less important, they represent a tentative proposal, and are open to debate and modification.

[19] Noting the first character of the POP is more specific than expected. This is intended and –although it could easily be adapted to “only mark first generalized character”– it makes very much sense for cases like *Ei*.

Table 1 *Different types of codes for RWL* ‘Ei’

IPA-types	ˈʔɔɪ̯, ˈʔaːɪ̯, ˈʔaɪ̯, ˈʔaː.jə, ˈʔɛɪ̯, ˈʔɛɪ̯x, ˈʔæj̣ə, ˈʔɛç, ˈʔɪç, ˈʔɛː.jə, ˈʔɛː.ɣə, ˈʔɔɪ̯ç, ˈʔæɪ̯, ˈʔɔːə̯, ˈʔɔɣ̣ɛ, ˈʔɑɪ̯, ˈʔɛɣ̣ɐ, ˈaɪ̯, ˈʔɔːɪ̯, ˈʔɑː.jə, ˈʔaɪ̯.jə, ˈʔɛh, ˈʔɑːɪ̯, ˈʔɒɪ̯, ˈʔɛk, ˈʔaɪ̯ç, ˈʔeːə̯, ˈʔaːə̯, ˈʔɒːɪ̯, ˈʔɛçk, ˈʔɛː.ʝə, ˈʔɛɡ̣ɐ, ˈʔɪɣ̥ə, ˈʔɒɪ̯ç, ˈʔeɪ̯, ˈʔɛç̣ɐ, ˈʔʊç, ˈʔɛʝ
Level 0 (POPs)	AI, EI, ECH, AAI, EU, ÄI, AAJE, ICH, ÄÄJE, ÅI, EUCH, ECHER, EK, EH, OI, AICH, EGGER, EICH, ÄJJE, ÄÄCHE, ÅE, OCHE, AIJE, EEE, AAE, ECHK, ICHE, OICH, UCH, EJ
Level 1	z_, zÇ, z=_, v_, z=Jz, vÇ, h_, v=Jz, z_Ç, zXXzR, zK, zH, zGGzR, z_X, vJJz, v=Ɣz, hXXz, z_Jz, z= =, zÇK, vXXz, h_Ç, hÇ, zJ
Level 2	z_, zÇ, v_, zJz, vÇ, h_, vJz, z_Ç, zXzR, zK, zH, zGzR, z_X, vƔz, hXz, z_Jz, z, zÇK, vXz, h_Ç, hÇ, zJ
Level 3	__, _Ç, _J_, __Ç, _Ç_R, _Ç_, _K, _H, _K_R, __J_, _, _ÇK, _J
Level 4	_, _Ç, _J_, _Ç_R, _Ç_, _K, _H, _K_R, _ÇK, _J
Level 5	A, E, Ä, I, Å, O, U

For comparison, Table 2 presents the data for the longer word *Dachboden* (‘attic’).

Table 2 *Different types of codes for RWL* ‘Dachboden’

IPA-types	ˈbaɫ.kʰən, ˈbɔḷə.kən, ˈʃpiʃ̣ɐ, ˈʃpaɪ̯.ʃɛ, ˈʃpeː.ɕə, ˈbal.ɡ̊ə, ˈbal.kŋ̍, ˈbal.kn̩, ˈbaɫ.kən, ˈbal.kən, ˈbʏœ̯n, ˈʃpɪʃ̣ɐ, ˈzaʊ̯.lɐ, ˈzuḷə, buː.ˈa̯ː.dn̩, ˈʃpɪ.ɕə, ˈbal.kə, ˈʃpiː.kɐ, ˈdak.rɔʊ̯m, ˈdax.ʃʁoɐ̯tʰ, ˈbaḷə.kən, ˈbal.kʰən, ˈboː.dn̩, ˈbɔɾ̣ə, ˈsʏl.nɐ, ˈbaɫ.kn̩, ˈdax.boː.dn̩, ˈbal.kʰə, ˈʔʊḷɐn, ˈʔʊŋ̣ɐn, ˈʔʊḷəɹn, ˈʔʊḷɛn, ˈʃpaɪ̯.ɕɐ, ˈʔɔɡ̣əln, ˈʃpæə̯.ʃɐ, ˈzœḷɐ, ˈboː.də, ˈʔɔḷaɹn, ˈbøː.n̩, ˈʃpaɪ̯.ʒɐ, ˈʃpaɪ̯.ʃɐ, ˈdaːk.ˌbyɐ̯.dn̩, ˈbɐl.kən, ˈzœḷɐ.ɪs, ˈʔɔḷɐn, ˈdax.ˌboː.dn̩, ˈdax.ˌbɔɾ̣ən, ˈbɔɾ̣ə, ˈhʊl.dɐn, ˈʔʊl.dɐn, ˈløːf, ˈspiː.çɐ, ˈʃpaɪ̯.ʒɛ, ˈsœl.dɐ, ˈbʏṇə, ˈbal.ɡŋ̍, ˈbœḍn̩, ˈbyːə̯l, ˈspiˑ.ʃɐ, ˈʃpi.ʃl̩, ˈʃpiː.ʃɐ, ˈʃpɪʃ̣a, ˈʔɔḅə.ˌhaʊ̯s, ˈzœl.dɐ, ˈbal.ɡ̊ən, ˈʃpiː.çə, ˈbɔl.kn̩, ˈʃpiː.ɕɐ, ˈbyː.nə, ˈhɔḷɐn, ˈɡyɐ̯n, ˈbɑl.kŋ̍, ˈd͡ʒaʊ̯.ˌlət, ˈbœn, ˈʃpɔʁ̣ə, ˈʃpɪɕ̣ɐ, ˈʔʊə̯.bə.ˌhaʊ̯s, ˈʔuṇə.ˌdak, ˈbɔl.kən, ˈdax.lɔʊ̯.kən, ˈbyː.dn̩s, ˌʔuṇɐn.ˈdaḳə, ˈbyə̯n, ˈbyə̯.dn̩, ˈʔʊn.dɐm daːk, ˈbɔɐ̯n, ˈbalk, ˈdaː.bɔḍn̩, ˈʔoːl.dɐn, ˈdak.byə̯.dn̩, ˈboː.ən, ˈbyːə̯.dn̩, ˈhaː.nəlt, ˈbɔḍn̩, ˈdak.ˌboː.dən, ˈbyːə̯.nən, ˈzœḷa, ˈbuə̯.dn̩, ˈʃpaɪ̯.ʃə, ˈdaɣ.boːm, ˈʔoːl.dɐ, ˈʔoː.lə, ˈboɐ̯.dn̩, ˈbyːə̯n, ˈzœḷəɾ, ˈdak.ˌkaː.mɐ, ˈʔɔḷəɹn, ˈbʊə̯.dn̩, ˈbaḷə.kə, ˈdak.bɔṇn̩, ˈkʰɛḷɐ, ˈdax.ˌstuː.bə, ˈʃpiː.ʃɐx, ˈʃpaɪ̯.ɕə, ˈʃpɪt͡s.ˌbɔṇn̩, ˈbʊɾ̣ə, ˈʃpɪʃ̣ə, ˈʃpɪʃ̣ɔɐ̯, ˈdak.bɔḍn̩, ˈbʊḍn̩, ˈzɔḷɐ, ˈsœḷɐ, ˈdaḳaː.mɐ, ˈbœːn, ˈbœː.dn̩, ˈbyə̯.nə, ˈdœn.t͡sn̩, ˈzɔl.dɐ, ˌʔʊŋ̣ɐm.ˈdaː.kə, ˈboː.rə, ˈdiː.ɬ̣n, ˈdak.ˌbɔn, ˈboː.dən, ˈʔoːn.ˌdɔx, ˈʃpɪʑ̣ɐ, ˈspaɪ̯.ɕə, ˈbʏn.ʃən, ˈdaːk.boː.dn̩, ˈʃpeː.ɕɐ, ˈʔʊɾ̣ə, ˈzœḷə, ˈdax.ˌboa̯.dn̩, ˈbal.k̚n̩, ˈdax.ˌbo.dn̩, ˈdaːk.ʁaʊ̯m, ˈkʰaː.mɐ, ˈdax.jə.ʃɔs, ˈdax.ɡ̊aṃɐ, ˈʃpaɪ̯.çə, ˈʃpɛɪ̯.çə, ˈkaː.mɐn, ˈdak.ˌboə̯n, ˈdax.ˌbøːv, ˈdaːk.buə̯n.dn̩, ˈdaːx.kaṃəɹ, ˈdak.boː.dn̩, ˈdak.ˌbuə̯.dn̩, ˈbaɫ.k̚n̩, ˈʔɔɐ̯.vɪx, ˈʃpaɪ̯.ʑ̣ə, ˈʃpɪç̣ɐ, ˈʃpaɪ̯.çn̩, ˈʃpaɪ̯.ʝə, ˈʃpaɪ̯.ç̣ɐ, ˈbuə̯.dn̩, ˈbɔl.ɡən, ˈdak.bʊə̯.dn̩, ˈdax.ˌboː.d̚n̩, ˈʃpɪ.çɐ, ˈʃpɪ.ɕɐ, ˈdax.ˌbɔḍən, ˈhɔɪ̯.ˌboː.dn̩, ˈbal.ɡə, ˈdax.ˌboə̯, ˈhaː.nə.ˌbal.kən, ˈdax.kaː.mɐ, ˈʃpɪɕ̣ə, ˈʃpɪʃ̣əʁn, ˈʃpɪç̣ə, ˈbiə̯n, ˈbaɫ.kn̩.ˌʀuːm, ˈʃpiː.ʃəx, ˈʃpiˑ.ʃɐ, ˈdak.ˌboː.dn̩, ˈʃpɪṣɐ, ˈʃpɪt͡s.ˌboːm, ˈdak.ˌkːa.mɐ, ˈdak.ˌbɔḍn̩, ˈʃpaɪ̯ɕ̣ɐ, ˈʃpɪ.çə, ˈbaː.kn̩, ˈzœɫ.dɐ, ˈdak.ˌkʰɔṃən, ˈdak.ˌbʊə̯ḍn̩, ˈʃpi.çə, ˈbaʁ̣ən, ˈdak.ˌbœn, ˈʃpʏɕ̣əl, ˈdak.ˌkɔṃən, ˈdaːx.ˌkaṃɐ, ˈɡyə̯n, ˈdax.ˌbʊɾ̣ə, ˈdax.ˌbʊʁ̣ə, ˈdax.ˌbʊɐ̯, ˈʃpiː.çɐ, ˈbyœ̯n, ˈda.ˌboː.dn̩, ˈlynp͡f, ˈlœʏ̯f, ˈvɛɪ̯m, ˈdax.ˌbɔɾ̣ə, ˈdax.ˌboː.dən, ˈzœɬ̣ɐ, ˈbaɫ.k̬ən, ˈbʊə̯ḍ̚n̩, ˈbɔn, ˈʔup.kaṃɐ, ˈdak.boː.dən, ˈkaː.mɐ, ˈdax.ˌbal.kən, ˈhaːn.ˌʔoːlt, ˈhaːn.ˌʔolt, ˈbyːn, ˈbaʊ̯.bm̩, ˈzœl.də, ˈdax.ˌbuə̯.bm̩, ˈʃpiː.ɕə, ˈdax.ˌbɛɐ̯l.dn̩, ˈʃpaɪ̯.ʒə, ˈzœḷe,

	ˈʃpai̯.çɐ, ˈzoḷɐ, ˈdaːk.ˌboːm, ˈhai̯.ˌboː.dən, ˈdaːk.bøːn, ˈʔʊṇɐ ˈdak, ˈdaːx.ˌkaː.mɐ, ˈlɔf, ˈdax.ˌkaː.mɐ, ˈman.ˌzaː.də, ˈdax.ˌbuə̯.dṇ, ˈʃpɛɪ̯.ɕə, ˈdak.ˌkaṃɐ, ˈbʏə̯n, ˈdax.ˌboːn, ˈɡʁɔʊ̯.pə.ˌbal.kən, ˈdakʰ.ˌkaː.mɐ, ˈdax.ˌbɔḍṇ, ˈʃpaɪ̯.ʑɐ, ˈbyːɐ̯n, ˈdax.ˌbuə̯n, ˈʔoː.də, ˈʃpaɪ̯.ça, ˈʃtro.ˌbal.kən, ˈdaː.ˌkaː.mɐ, ˈzœːl.dɐ, ˈzœː.lɐ, ˈtaʊ̯.bṇ.ˌʃlax, ˈʃpaɪ̯ː.ʃɐ, ˈdax.ˌbœn, ˈzʏḷɐ, ˈhiː.lə, ˈʔʊə̯.və.nə, ˈdak.ˌbʊə̯.dṇ, ˈhɔɪ̯ç.ˌboː.dṇ, ˈbɔʊ̯.ə
Level 0 (POPs)	BALKEN, BOLLEKEN, SCHPISCHER, SCHPAISCHE, SCHPEESCHE, BALGE, BALKNG, BALKN, BÜÖN, SAULER, SULLE, BUU-ADN, SCHPISCHE, BALKE, SCHPIIKER, DAKROUM, DACHSCHROAT, BALLEKEN, BOODN, BORRE, SÜLNER, DACHBOODN, ULLAN, UNGAN, ULLERN, ULLEN, SCHPAISCHER, OGGELN, SCHPÄESCHER, SÖLLER, BOODE, OLLARN, BÖÖ-N, SCHPAIJER, DAAKBÜADN, SÖLLER-IS, OLLAN, DACHBORREN, HULDAN, ULDAN, LÖÖF, SPIICHER, SCHPAIJE, SÖLDER, BÜNNE, BALGNG, BÖDDN, BÜÜEL, SPISCHER, SCHPISCHL, SCHPIISCHER, SCHPISCHA, OBBEHAUS, BALGEN, SCHPIICHE, BOLKN, BÜÜNE, HOLLAN, GÜAN, DJAULET, BÖN, SCHPORRE, UEBEHAUS, UNNEDAK, BOLKEN, DACHLOUKEN, BÜÜDNS, UNNANDAKKE, BÜEN, BÜEDN, UNDAM DAAK, BOAN, BALK, DAABODDN, OOLDAN, DAKBÜEDN, BOO-EN, BÜÜEDN, HAANELT, BODDN, DAKBOODEN, BÜÜENEN, SÖLLA, BUEDN, DACHBOOM, OOLDER, OOLE, BOADN, BÜÜEN, DAK-KAAMER, OLLERN, BUODN, BALLEKE, DAKBON´, KELLER, DACHSTUUBE, SCHPIISCHACH, SCHPITZBON´, BURRE, SCHPISCHOA, DAKBODDN, BUDDN, SOLLER, DAKKAAMER, BÖÖN, BÖÖDN, BÜENE, DÖNZN, SOLDER, UNGAMDAAKE, BOORE, DII-LN, DAKBON, BOODEN, OONDOCH, SPAISCHE, BÜNSCHEN, DAAKBOODN, SCHPEESCHER, URRE, SÖLLE, DACHBOADN, DACHBODN, DAAKRAUM, KAAMER, DACHJESCHOS, DACHGAMMER, SCHPAICHE, SCHPEICHE, KAAMAN, DAKBOEN, DACHBÖÖW, DAAKBUONDN, DAACHKAMMER, DAKBOODN, DAKBUODN, OAWICH, SCHPICHER, SCHPAICHN, SCHPAICHER, BOLGEN, DACHBODDEN, HEUBOODN, DACHBOE, HAANEBALKEN, DACHKAAMER, SCHPISCHERN, SCHPICHE, BIEN, BALKN-RUUM, SCHPIISCHECH, SCHPISSER, SCHPITZBOOM, DAK-KKAMER, BAAKN, DAK-KOMMON, DAKBUODDN, BARREN, DAKBÖN, SCHPÜSCHEL, GÜEN, DACHBURRE, DACHBUA, SCHPIICHER, DABOODN, LÜNPF, LÖÜF, WEIM, DACHBORRE, DACHBOODEN, BON, UPKAMMER, DACHBALKEN, HAAN-OOLT, HAAN-OLT, BÜÜN, BAUBM, SÖLDE, DACHBUOBM, SCHPIISCHE, DACHBEALDN, DAAKBOOM, HAIBOODEN, DAAKBÖÖN, UNNA DAK, DAACHKAAMER, LOF, MANSAADE, DACHBUODN, SCHPEISCHE, DAK-KAMMER, DACHBOON, GROUPEBALKEN, DACHBODDN, BÜÜAN, DACHBUON, OODE, SCHPAICHA, SCHTROBALKEN, DAAKAAMER, SÖÖLDER, SÖÖLER, TAUBN-SCHLACH, SCHPAIISCHER, DACHBÖN, SÜLLER, HIILE, UEWENE, DAKBUEDN, HEUCHBOODN, BOU-E
Level 1	BzLKzN, BzLKN, SvLLzR, ΣPz_ΣzR, BzLKŊ, Bv_N, ΣPvΣΣzR, SvLDzR, ΣPz_Σz, DzXBh = DN, SvLLz, Bh = DN, hLLzN, ΣPvΣΣz, BvN, BzLGz, ΣPvXXzR, BzLKz, Bv = _N, ΣPz_Çz, ΣPz_ÇzR, Lv = F, BhDDN, Bh_DN, ShLDzR, DzXBhRRz, DzK-Kz = MzR, ShLLzR, ΣPvÇzR, ΣPvΣz, ΣPz_Jz, ΣPv = ΣzR, DzKBh_DN, ΣPvΣzR, BzLLzKzN, BhRRz, hLLzRN, Bh = Dz, h = LDzN, h = Lz, DzKBhDDN, Bv = N, DzKBh = DN, DzXBhDDzN, ΣPvXXz, DzXBh = N, BhLLzKzN, ΣPz_JzR, ΣPvΣL, Bv = Nz, Gv_N, BzLK, Bv = _DN, DzKBh = DzN, h = LDzR, ΣPv = ΣzX, Bv_Nz, Bh = Rz, DzXBh_DN, Kz = MzR, Dz = XKzMMzR, DzXBh_, DzXKz = MzR, ΣPvÇz, DzK-

	KhMMhN, ΣPv = ÇzR, DzXBzLKzN, Bz_BM, ΣPv = Σz, ΣPz = Jz, ΣPz = Σz, Sz_LzR, ShLLz, Bh = -_DN, ΣPv = KzR, DzKRh_M, DzXΣRh_T, SvLNzR, hŊŊzN, hGGzLN, ΣPv_ΣzR, Bv = -N, Dz = KBv_DN, SvLLzR-_S, DzXBhRRzN, HhLDzN, hLDzN, SPv = ÇzR, BvNNz, BzLGŊ, BvDDN, Bv = _L, SPvΣzR, hBBzHz_S, BzLGzN, ΣPv = Çz, BhLKN, HhLLzN, DJz_LzT, ΣPhRRz, h_BzHz_S, hNNzDzK, BhLKzN, DzXLh_KzN, Bv = DNS, hNNzNDzKKz, Bv_DN, hNDzM Dz = K, Bh_N, Dz = BhDDN, DzKBv_DN, Bh = -_N, Hz = NzLT, Bv = _NzN, DzƔBh = M, BzLLzKz, DzKBhN´, KzLLzR, DzXSTh = Bz, ΣPvTZBhN´, ΣPvΣΣh_, DzKKz = MzR, Bv = DN, DvNZN, hŊŊzMDz = Kz, Dv = -LN, DzKBhN, Bh = DzN, h = NDhX, SPz_Σz, BvNΣzN, Dz = KBh = DN, ΣPz = ΣzR, hRRz, DzXBhDN, Dz = KRz_M, DzXJzΣhS, DzXGzMMzR, Kz = MzN, DzKBh_N, DzXBv = W, Dz = KBh_NDN, h_WvX, ΣPz_ÇN, BhLGzN, Hz_Bh = DN, Hz = NzBzLKzN, ΣPvΣΣzRN, BzLKN-Rh = M, ΣPvSSzR, ΣPvTZBh = M, DzK-KKzMzR, ΣPz_ΣΣzR, Bz = KN, DzKBh_DDN, BzRRzN, DzKBvN, ΣPvΣΣzL, KzMMz, v_MSzL, DzBh = DN, LvNPF, Lv_F, Wz_M, DzXBh = DzN, BhN, hPKzMMzR, Hz = N-_ = LT, Hz = N-_LT, SvLDz, DzXBh_BM, DzXBz_LDN, Dz = KBh = M, Hz_Bh = DzN, Dz = KBv = N, hNNz DzK, Dz = XKz = MzR, LhF, MzNSz = Dz, DzK-KzMMzR, GRh_PzBzLKzN, DzXBhDDN, DzXBh_N, h = Dz, Bz = LKzN, SvLLh, ΣTRhBzLKzN, Dz = Kz = MzR, Sv = LDzR, Sv = LzR, Tz_BN-ΣLzX, ΣPz_ = ΣzR, DzXBvN, Hv = Lz, h_WzNz, Hz_ÇBh = DN, Bh_-_
Level 2	BzLK, SvLz, ΣPvΣz, ΣPz_Σz, BhDN, Bv_N, hLz, BvN, SvLDz, KzMz, Bh_DN, hRz, ΣPz_Çz, hDz, ΣPvXz, BzLGz, ΣPvÇz, hLDzR, LvF, ΣPz_Jz, BhN, hLDzN, Bv_DN, Bh_N, BzLzKz, BvDN, DzXBh_, BhLzKzN, ΣPzΣz, ΣPvΣL, Gv_N, DzKhMhN, HzN_LT, Bz_BM, ΣPzJz, Sz_LzR, ΣPvKzR, DzKRh_M, DzXΣRh_T, SvLNzR, hŊzN, hGzLN, ΣPv_ΣzR, SPvÇzR, BzLGŊ, Bv_L, SPvΣzR, hBzHz_S, BhLKN, DJz_LzT, h_BzHz_S, hNzDzK, BhLKzN, DzXLh_KzN, hNzNDzKz, hNDzM DzK, HzNzLT, DzƔBhM, KzLzR, DzXSThBz, ΣPvTZBhN´, ΣPvΣh_, DvNZN, hŊzMDzKz, DvLN, hNDhX, SPz_Σz, DzKRz_M, DzXJzΣhS, DzXGzMzR, DzXBvW, h_WvX, ΣPz_ÇN, BhLGzN, ΣPvSzR, ΣPvTZBhM, BzKN, BzRzN, v_MSzL, LvNPF, Lv_F, Wz_M, DzXBz_LDN, DzKBhM, hNz DzK, LhF, MzNSzDz, SvLh, Tz_BNΣLzX, HvLz, h_WzNz, Bh__
Level 3	P_LK, S_L_, ΣP_Σ_, ΣP__Σ_, P__N, P_DN, P_N, S_LD_, K_M_, ΣP_Ç_, P__DN, P_R_, ΣP__Ç_, _L_N, P_D_, L_F, ΣP__J_, P_L_K_, _LD_N, _L_RN, H_N_LD, _L_, D_ÇP__, P__PM, D_KR__M, ΣP_ΣL, _P_H__S, K__N, _LD_R, ΣP_J_, H_L_, S__L_R, ΣP_K_R, D_ÇΣR__D, S_LN_R, _Ŋ_N, _K_LN, H_LD_N, SP_Ç_R, P__L, SP_Σ_R, DJ__L_D, _N_D_K, D_ÇL__K_N, _N_ND_K_, _ND_M D_K, D_ÇP_M, K_L_R, D_ÇSD_P_, ΣP_DZP_N´, D_NZN, _Ŋ_MD_K_, D_LN, _ND_Ç, SP__Σ_, _R_, D_ÇJ_Σ_S, D_ÇP_F, __F_Ç, ΣP__ÇN, ΣP_S_R, ΣP_DZP_M, P_KN, __MS_L, L_NPF, L__F, F__M, D_KP_M, _N_ D_K, M_NS_D_, _D_, D__PNΣL_Ç, __F_N_, P___
Level 4	P_L, ΣP_Σ_, P_N, S_L_, P_DN, ΣP_Ç_, S_LD_, K_M_, P_R_, _L_N, P_D_, L_F, ΣP_J_, K_N, D_ÇP_, _LD_N, _L_RN, _L_, P_PM, D_KR_M, ΣP_ΣL, _P_H_S, _LD_R, SP_Σ_, H_NLD, H_L_, ΣP_K_R, D_ÇΣR_D, S_LN_R, _Ŋ_N, _K_LN, H_LD_N, SP_Ç_R, DJ_L_D, _N_D_K, _N_ND_K_, _ND_M D_K, H_N_LD, K_L_R, D_ÇSD_P_, ΣP_DZP_N´, D_NZN, _Ŋ_MD_K_, D_LN, _ND_Ç, _R_, D_ÇJ_Σ_S, _F_Ç, ΣP_ÇN, ΣP_S_R, ΣP_DZP_M, P_KN, _MS_L, L_NPF, F_M, D_KP_M, _N_ D_K, M_NS_D_, _D_, D_PNΣL_Ç, _F_N_, P_
Level 5	B, Σ, S, D, O, U, H, L, K, G, W, M, T

The codes in Table 1 and Table 2 already represent the final cluster categories, i.e., after mapping complex codes transitively to existing subcodes (see above HEAPSLAUP→LAUP example) and removing the redundant complex codes. Such a mapping is shown for three levels of *Dachboden* (other words involve fewer mappings, even none, as *Ei*) in Table 3.

Table 3 *Remappings of codes of* 'Dachboden'

Level 2	BhDz→hDz, BhDzN→hDz, BhRz→hRz, BvDNS→BvDN, BvNz→BvN, BvNΣzN→BvN, Bv_Nz→Bv_N, Bv_NzN→Bv_N, BzLGzN→BzLGz, BzLKN→BzLK, BzLKNRhM→BzLK, BzLKz→BzLK, BzLKzN→ BzLK, BzLKŊ→BzLK, BzLzKzN→BzLzKz, DzBhDN→BhDN, DzKBhDN→BhDN, DzKBhDzN→hDz, DzKBhN→BhN, DzKBhN´→BhN, DzKBh_DN→Bh_DN, DzKBh_N→Bh_N, DzKBh_NDN→Bh_N, DzKBvN→BvN, DzKBv_DN→Bv_DN, DzKzMz→KzMz, DzXBhDN→BhDN, DzXBhDzN→hDz, DzXBhN→BhN, DzXBhRz→hRz, DzXBhRzN→hRz, DzXBh_BM→DzXBh_, DzXBh_DN→Bh_DN, DzXBh_N→Bh_N, DzXBvN→BvN, DzXBzLKzN→BzLK, DzXKzMz→KzMz, DzXKzMzR→KzMz, GRh_PzBzLKzN→BzLK, HhLDzN→hLDz, HhLzN→hLz, Hh_BhDN→BhDN, Hh_ÇBhDN→BhDN, HzNzBzLKzN→BzLK, Hz_BhDzN→hDz, KzMzN→KzMz, ShLDz→hLDz, ShLz→hLz, SvLzR→SvLz, SvLz_S→SvLz, hLDzN→hLDz, hLzN→hLz, hLzRN→hLz, hPKzMz→KzMz, ΣPhRz→hRz, ΣPvΣzL→ΣPvΣz, ΣPvΣzRN→ΣPvΣz, ΣPvΣzX→ΣPvΣz, ΣTRhBzLKzN→BzLK
Level 3	D_KP_DN→P_DN, D_KP_D_N→P_D_, D_KP_N→P_N, D_KP_N´→P_N, D_KP__DN→P__DN, D_KP__N→ P__N, D_KP__NDN→P__N, D_K_M_→K_M_, D_K_M_N→K_M_, D_P_DN→P_DN, D_ÇK_M_→K_M_, D_ÇK_M_R→K_M_, D_ÇP_DN→P_DN, D_ÇP_D_N→P_D_, D_ÇP_LK_N→P_LK, D_ÇP_N→P_N, D_ÇP_R_→ P_R_, D_ÇP_R_N→P_R_, D_ÇP__DN→P__DN, D_ÇP__LDN→D_ÇP__, D_ÇP__N→P__N, D_ÇP__PM→P__PM, H_L_N→H_L_, H_N_P_LK_N→P_LK, H__P_DN→P_DN, H__P_D_N→P_D_, H__ÇP_DN→P_DN, KR__P_P_LK_N→P_LK, K_M_N→K_M_, P_DNS→P_DN, P_D_N→P_D_, P_LKN→P_LK, P_LKNR_M→P_LK, P_LK_→P_LK, P_LK_N→P_LK, P_LKŊ→P_LK, P_L_K_N→P_L_K_, P_N_→P_N, P_NΣ_N→P_N, P_R_N→ P_R_, P__N_→P__N, P__N_N→P__N, S_L_R→S_L_, S_L__S→S_L_, ΣDR_P_LK_N→P_LK, ΣP_R_→ P_R_, ΣP_Σ_L→ΣP_Σ_, ΣP_Σ_RN→ΣP_Σ_, ΣP_Σ__→ΣP_Σ_, ΣP_Σ_Ç→ΣP_Σ_, _PK_M_→K_M_, __P_H__S→ _P_H__S
Level 4	D_KP_DN→P_DN, D_KP_D_N→P_D_, D_KP_N→P_N, D_KP_NDN→P_N, D_KP_N´→P_N, D_K_M_→K_M_, D_K_M_N→K_M_, D_P_DN→P_DN, D_ÇK_M_→K_M_, D_ÇK_M_R→K_M_, D_ÇL_K_N→K_N, D_ÇP_DN→ P_DN, D_ÇP_D_N→P_D_, D_ÇP_F→D_ÇP_, D_ÇP_LDN→D_ÇP_, D_ÇP_LK_N→K_N, D_ÇP_M→D_ÇP_, D_ÇP_N→P_N, D_ÇP_PM→P_PM, D_ÇP_R_→P_R_, D_ÇP_R_N→D_ÇP_, H_L_N→H_L_, H_N_P_LK_N→ K_N, H_P_DN→P_DN, H_P_D_N→P_D_, H_ÇP_DN→P_DN, KR_P_P_LK_N→K_N, K_M_N→K_M_, P_DNS→P_DN, P_D_N→P_D_, P_LK→P_L, P_LKN→P_L, P_LKNR_M→P_L, P_LK_→P_L, P_LK_N→P_L, P_LKŊ→P_L, P_L_K_→P_L, P_L_K_N→P_L, P_N_→P_N, P_N_N→P_N, P_NΣ_N→P_N, P_R_N→P_R_, S_L_R→S_L_, S_L_S→S_L_, ΣDR_P_LK_N→K_N, ΣP_R_→P_R_, ΣP_Σ_L→ΣP_Σ_, ΣP_Σ_RN→ΣP_Σ_, ΣP_Σ_Ç→ΣP_Σ_, _PK_M_→K_M_

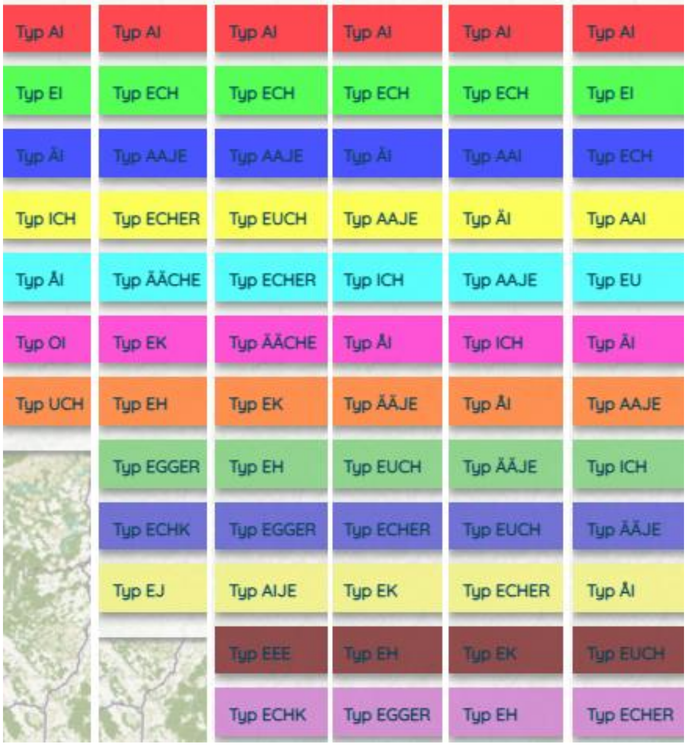
4.5 DPA: application

Recall that the purpose of POP clustering is to arrive at a smaller set of POP types in order to obviate cluttered maps and corresponding visual confusion, and to allow for intuitive use and easy interaction with the ‘Speaking DMW’. Basically, this is realized by letting the user select the granularity level he/she is interested in, thereby reducing the complexity of presentation and interaction. With a level being set, the POP types to be shown in the legend can then be specified, and the level of detail of the preview map can be determined. This is implemented as follows.

First, the clusters of a level (as well as their POPs) are sorted according to their relevance, i.e., the frequency of their POP *instances*. In the legend, the clusters can then be offered in decreasing order, starting with the most relevant one. Second, each cluster is *termed* to be easily identifiable, not just by the set of POPs it represents (which is shown on mouseover of the cluster field). Unfortunately, it is not possible to come up with an automatic procedure to *generate* a term for a cluster (based on its POPs). However, simply *using* its most relevant POP as its label suffices for this purpose, as the DPA results in distinctive clusters by definition. Third, we assign each of the *twelve* highest ranked clusters of a level one of a set of twelve maximally contrastive colors (correspondingly ranked by their salience). While the number is arbitrary and debatable, the use of colors as opposed to specific symbols is supposed to make the POP types easily recognizable (note that we offer a secondary color palette for people with disabilities in color perception) and to avoid visual confusion. To signify data distribution and diversity on the preview map, we mainly use pie charts (see below). They can be regarded as the only option for the signification of type clusters on our preview maps, both due to the large number of explored places, and to the fact that type diversity per place is frequent for automatically categorised data.

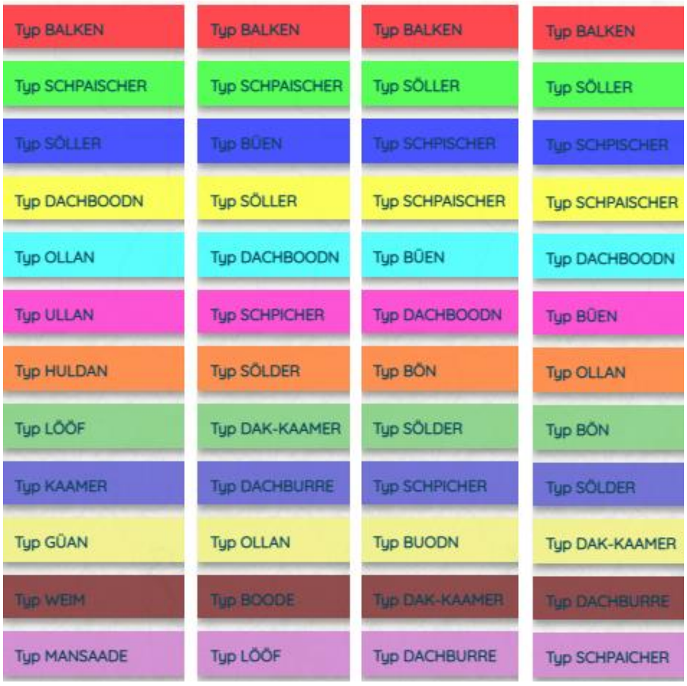
To exemplify the application of the DPA, Figure 7 and Figure 8 show the menus of decreasing granularity level aligned for comparison, each with their clickable, colored, and typed fields corresponding to the clusters. Recall that the up to twelve colored fields are ordered by the number of corresponding variants (further gray-colored fields/clusters not shown here).

Figure 7 *Clickable variant type fields in the legend for the granularity levels of* ‘Ei’ *(5 to 0)*



Typ AI	Typ AI	Typ AI	Typ AI	Typ AI	Typ AI
Typ EI	Typ ECH	Typ ECH	Typ ECH	Typ ECH	Typ EI
Typ ÄI	Typ AAJE	Typ AAJE	Typ ÄI	Typ AAI	Typ ECH
Typ ICH	Typ ECHER	Typ EUCH	Typ AAJE	Typ ÄI	Typ AAI
Typ ÅI	Typ ÄÄCHE	Typ ECHER	Typ ICH	Typ AAJE	Typ EU
Typ OI	Typ EK	Typ ÄÄCHE	Typ ÅI	Typ ICH	Typ ÅI
Typ UCH	Typ EH	Typ EK	Typ ÄÄJE	Typ ÅI	Typ AAJE
	Typ EGGER	Typ EH	Typ EUCH	Typ ÄÄJE	Typ ICH
	Typ ECHK	Typ EGGER	Typ ECHER	Typ EUCH	Typ ÄÄJE
	Typ EJ	Typ AIJE	Typ EK	Typ ECHER	Typ ÅI
		Typ EEE	Typ EH	Typ EK	Typ EUCH
		Typ ECHK	Typ EGGER	Typ EH	Typ ECHER

Figure 8 *Clickable variant type fields in the legend for the granularity levels of* ‘Dachboden’ *(5 to 2)*

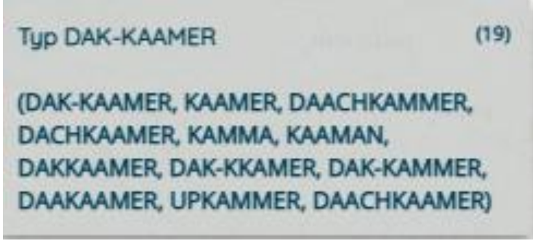


Typ BALKEN	Typ BALKEN	Typ BALKEN	Typ BALKEN
Typ SCHPAISCHER	Typ SCHPAISCHER	Typ SÖLLER	Typ SÖLLER
Typ SÖLLER	Typ BÜEN	Typ SCHPISCHER	Typ SCHPISCHER
Typ DACHBOODN	Typ SÖLLER	Typ SCHPAISCHER	Typ SCHPAISCHER
Typ OLLAN	Typ DACHBOODN	Typ BÜEN	Typ DACHBOODN
Typ ULLAN	Typ SCHPICHER	Typ DACHBOODN	Typ BÜEN
Typ HULDAN	Typ SÖLDER	Typ BÖN	Typ OLLAN
Typ LÖÖF	Typ DAK-KAAMER	Typ SÖLDER	Typ BÖN
Typ KAAMER	Typ DACHBURRE	Typ SCHPICHER	Typ SÖLDER
Typ GÜAN	Typ OLLAN	Typ BUODN	Typ DAK-KAAMER
Typ WEIM	Typ BOODE	Typ DAK-KAAMER	Typ DACHBURRE
Typ MANSAADE	Typ LÖÖF	Typ DACHBURRE	Typ SCHPAICHER

It can easily be seen in Figure 8 that with the “Typ Balken” fields always ranked highest, they correspond to the most frequent clusters (probably different in each case), while the ranks of other types change. For example, type BÜEN (level 4, second column, rank three, darkblue) is ranked higher than the same type on level 3 (third column, rank five, lightblue). This is due to the fact that while vowel aspects are conflated on level 4, level 3 distinguishes vowel change (diphthongs) from non-change (monophthongs), hence the level-3 differentiation of types BÜEN and BÖN (rank seven, darkorange), both with smaller frequency values.

Figure 9 depicts a single POP type field on mouse-over (grayed out *Dachboden* cluster field on level 2). The cluster is determined by the common code KzMz (of KAMA), and named by the most frequent POP DAK-KAAMMER (the overall number of occurrences of cluster variants given in brackets).

Figure 9 *A level-2 cluster for* ‘Dachboden’



Typ DAK-KAAMER (19)

(DAK-KAAMER, KAAMER, DAACHKAMMER, DACHKAAMER, KAMMA, KAAMAN, DAKKAAMER, DAK-KKAMER, DAK-KAMMER, DAAKAAMER, UPKAMMER, DAACHKAAMER)

Observe that no principles of dialectology are involved in this multi-layered clustering scheme. Apart from the fact that none were available at the time of its development, reaction time of the map interface is essential for user experience (computing level info and corresponding map presentation of the data on level choice). It can be expected that any more sophisticated scheme (for example, considering location aspects of variants for categorization) would slow down performance noticeably.

5. Aspects of Preview maps

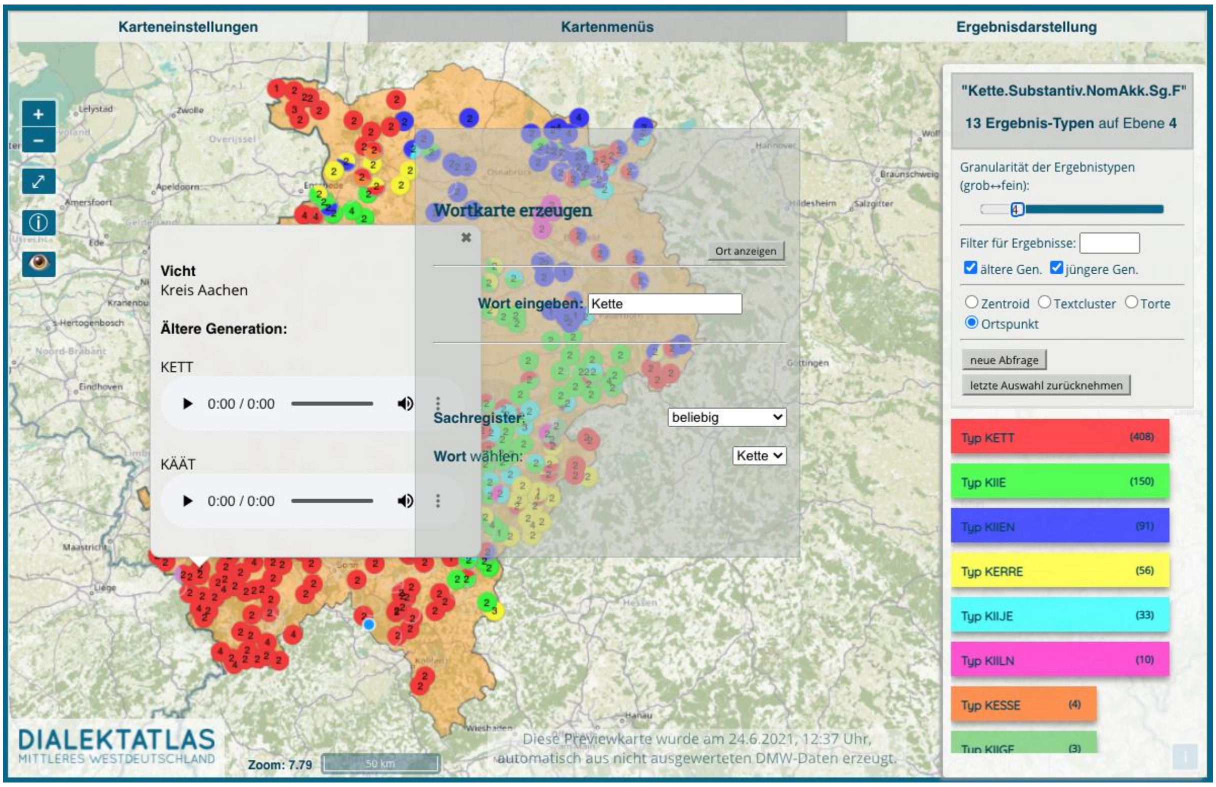
There are a number of requirements regarding content (presentation), interaction, and (web) technology to be considered in implementing the map interface. In the following, these requirements and their corresponding solutions are discussed.

5.1 Content (presentation)

At the core of the DMW system, there is the mass of diverse dialect(ological) data, and the requirement to taylor its presentation to different *user* classes (lay people and experts). More specifically, there are object and meta data of exploration and analysis, some of which are relevant for the map interface. Primarily, the preview maps are supposed to show the analysis data for some exploration item (a RWL), and to provide the facility to hear the correspondingly cutted audios (the observants of the RWSes) of the informants at some explored place.

Within this basic functionality of the “Speaking DMW”, one has to distinguish the audio-visual presentation of variants of words via the POP mechanism, and the possibility to select and hear (multiple) Wenker sentences. In the map interface, *word maps* and *Wenker maps* are offered for this distinction, respectively (see Figure 10 and Figure 11). Both figures show the removable central half-transparent map query menus and the popup-windows containing the wav/mp4 players at a certain mouse-clicked place. The map types are different in that Wenker maps lack the legend of word maps. Also, while a word map popup contains players for the variants of each person (i.e., audio on demand), playing the Wenker sentences starts on click, and as a looped sequence of all wavs/mp4s of the selected Wenker sentences, and of all persons at that place (navigation backward and forward is offered).

Figure 10 *Word map of* 'Kette'



Vicht
Kreis Aachen

Ältere Generation:

KETT
▶ 0:00 / 0:00

KÄÄT
▶ 0:00 / 0:00

Wortkarte erzeugen

Ort anzeigen

Wort eingeben: Kette

Sachregister: beliebig

Wort wählen: Kette

"Kette.Substantiv.NomAkk.Sg.F"
13 Ergebnis-Typen auf Ebene 4

Granularität der Ergebnistypen (grob↔fein): 4

Filter für Ergebnisse:

☑ ältere Gen. ☑ jüngere Gen.

○ Zentroid ○ Textcluster ○ Torte
● Ortspunkt

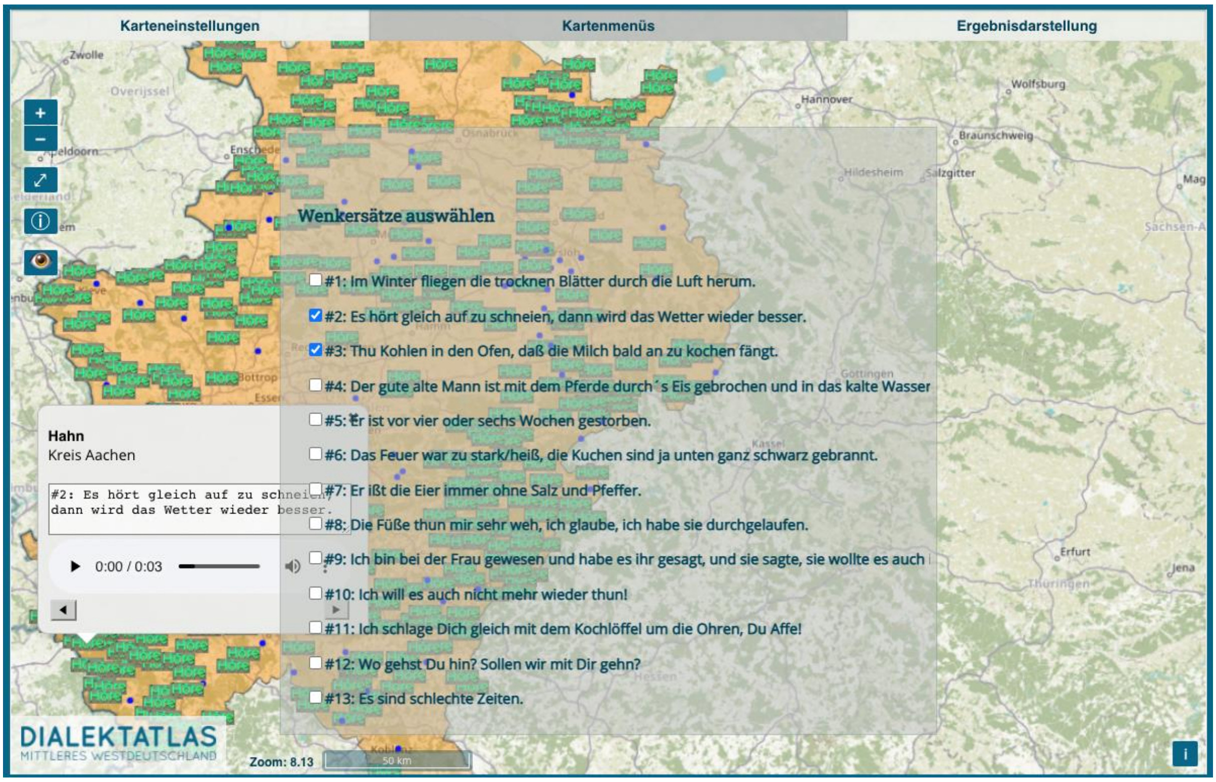
neue Abfrage

letzte Auswahl zurücknehmen

🟥 Typ KETT	(408)
🟩 Typ KIIE	(150)
🟦 Typ KIIEN	(91)
🟨 Typ KERRE	(56)
Typ KIIJE	(33)
🟪 Typ KIILN	(10)
🟧 Typ KESSE	(4)
Typ KIIGE	(3)

Diese Previewkarte wurde am 24.6.2021, 12:37 Uhr, automatisch aus nicht ausgewerteten DMW-Daten erzeugt.

Figure 11 *Wenker map of selected Wenker sentences*



Wenkersätze auswählen

☐ #1: Im Winter fliegen die trocknen Blätter durch die Luft herum.
☑ #2: Es hört gleich auf zu schneien, dann wird das Wetter wieder besser.
☑ #3: Thu Kohlen in den Ofen, daß die Milch bald an zu kochen fängt.
☐ #4: Der gute alte Mann ist mit dem Pferde durch´s Eis gebrochen und in das kalte Wasser
☐ #5: Er ist vor vier oder sechs Wochen gestorben.
☐ #6: Das Feuer war zu stark/heiß, die Kuchen sind ja unten ganz schwarz gebrannt.
☐ #7: Er ißt die Eier immer ohne Salz und Pfeffer.
☐ #8: Die Füße thun mir sehr weh, ich glaube, ich habe sie durchgelaufen.
☐ #9: Ich bin bei der Frau gewesen und habe es ihr gesagt, und sie sagte, sie wollte es auch
☐ #10: Ich will es auch nicht mehr wieder thun!
☐ #11: Ich schlage Dich gleich mit dem Kochlöffel um die Ohren, Du Affe!
☐ #12: Wo gehst Du hin? Sollen wir mit Dir gehn?
☐ #13: Es sind schlechte Zeiten.

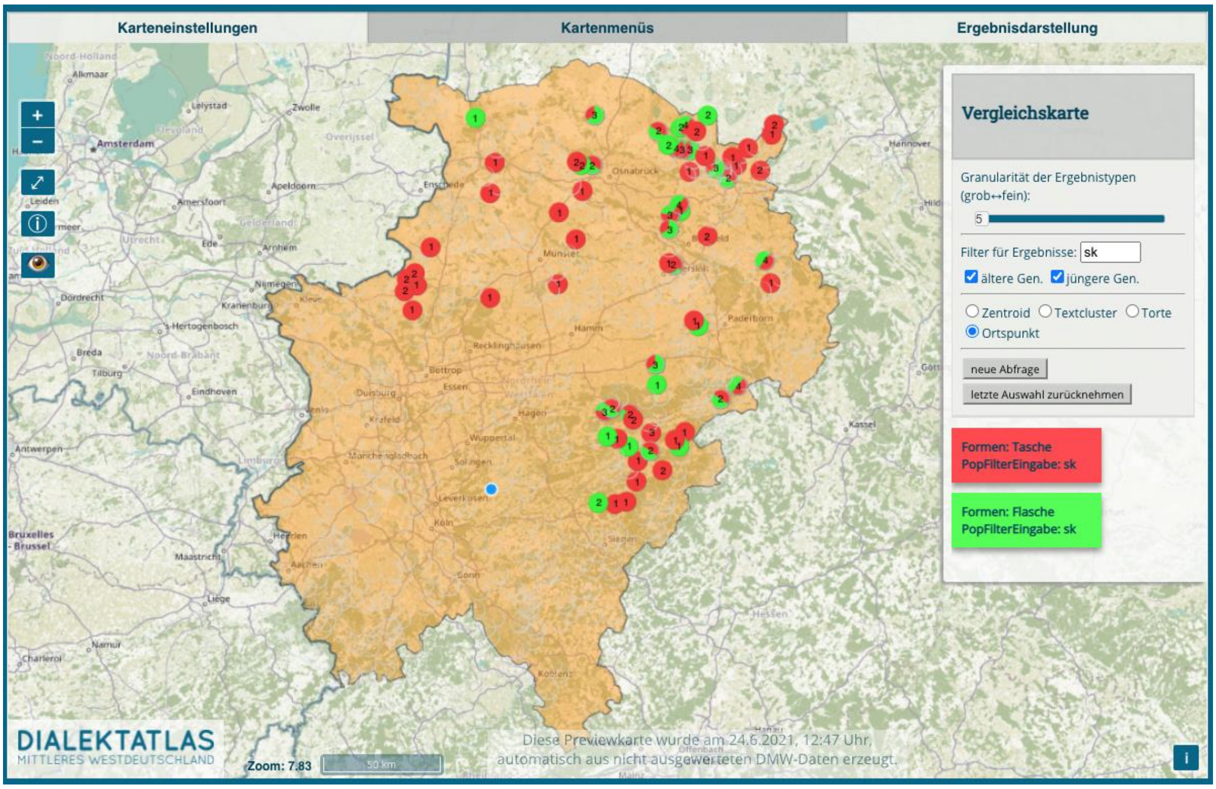
Hahn
Kreis Aachen

```
#2: Es hört gleich auf zu schneien,
dann wird das Wetter wieder besser.
```

▶ 0:00 / 0:03

It is also possible to construct several word maps at the same time, and to compare the corresponding distributions on a so-called *comparison* map ("Vergleichskarte"). Figure 12 shows such a display, in which (only) occurrences containing *sk* of *two* words (*Tasche, Flasche*) are mapped. In the expert version, phenomenon-related analysis data will be presented on corresponding phenomena preview maps. In the legend of a comparison map, the variant type fields do not show POP(type)s, but the specifications of the query for the corresponding map-to-be-compared (i.e., the list of option choices made).

Figure 12 *Comparison map*



Vergleichskarte

Granularität der Ergebnistypen (grob↔fein): 5

Filter für Ergebnisse: sk

☑ ältere Gen. ☑ jüngere Gen.

○ Zentroid ○ Textcluster ○ Torte ● Ortspunkt

🟥 Formen: Tasche PopFilterEingabe: sk

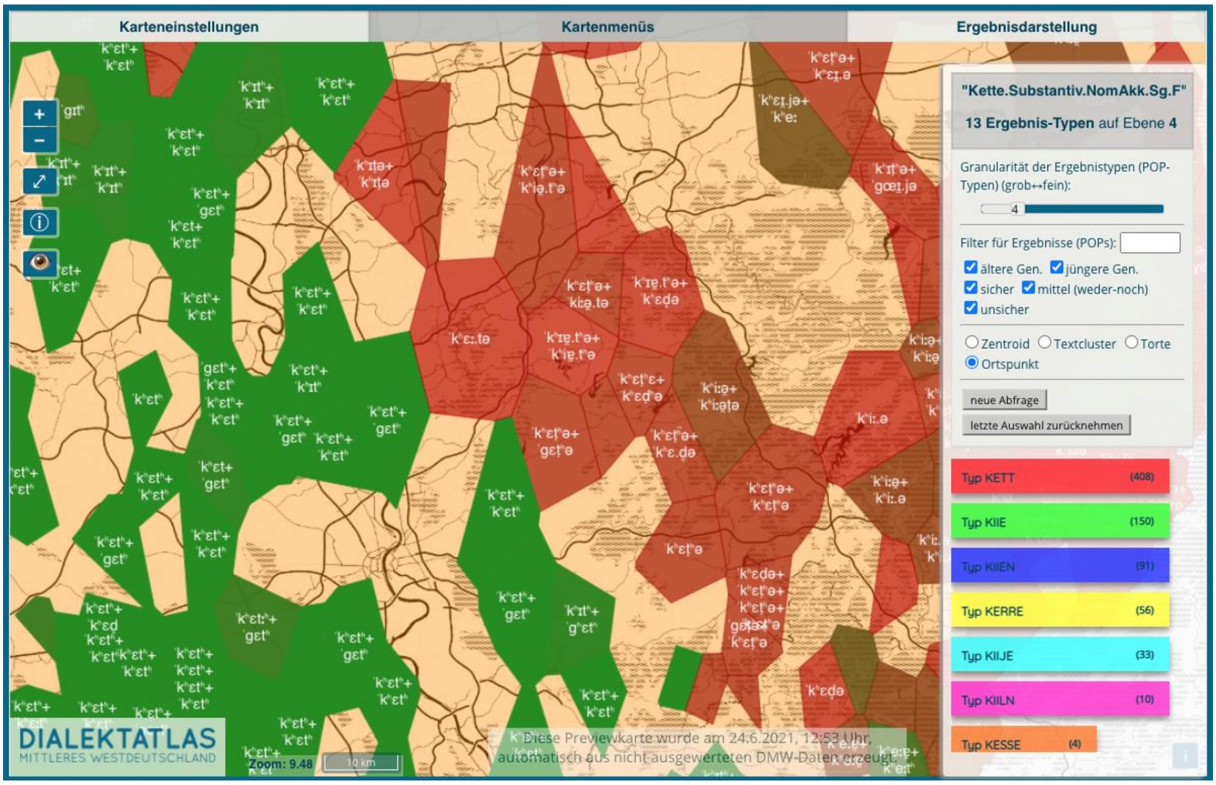
🟩 Formen: Flasche PopFilterEingabe: sk

Diese Previewkarte wurde am 24.6.2021, 12:47 Uhr, automatisch aus nicht ausgewerteten DMW-Daten erzeugt.

The currently available dialectometric type is a reference point map, which codes distances between variants of a clicked-on reference cell and each other cell as colors on a linear green-red scale (close to distant).[20] There are two mechanisms to ensure adequacy of this scale for the probable case of very different variants (where red would be the dominant color). First, the sorted domain of values is cut off at some point, and outliers receive the maximum value ("clamping"), which better reflects the differences in the proximal range of values, while only conflating different distant values. Second, reference point comparison is performed for the places of the browser window's content, i.e., the currently "visible" area. Different from a global process, this better mirrors the finer distinctions of closer related variants in zoomed-in situations. Figure 13 displays such a reference point map for *Kette*, showing "regions of similarity".

[20] Since each place typically has more than one, and potentially different, variants, the smallest value (i.e., of the closest variants) is selected. As to the empty/transparent cells, data for the corresponding place are either not yet analyzed, or have been sorted out as invalid or irrelevant.

Figure 13 *Reference point map*



"Kette.Substantiv.NomAkk.Sg.F"

13 Ergebnis-Typen auf Ebene **4**

Granularität der Ergebnistypen (POP-Typen) (grob↔fein): 4

Filter für Ergebnisse (POPs):

☑ ältere Gen. ☑ jüngere Gen.
☑ sicher ☑ mittel (weder-noch)
☑ unsicher

○ Zentroid ○ Textcluster ○ Torte
● Ortspunkt

neue Abfrage

letzte Auswahl zurücknehmen

🟥 Typ KETT	(408)
🟩 Typ KIIE	(150)
🟦 Typ KIIEN	(91)
🟨 Typ KERRE	(56)
Typ KIIJE	(33)
🟪 Typ KIILN	(10)
🟧 Typ KESSE	(4)

Diese Previewkarte wurde am 24.6.2021, 12:53 Uhr, automatisch aus nicht ausgewerteten DMW-Daten erzeugt.

In general, meta data are used to restrict data presentation, for example, to some age group, or to some confidence-of-transcription level (the POP mechanism can be regarded as another case in point, with the index terms representing similarity data about the variants).

For the presentation of our mass data of variants, it is necessary to guarantee the *perceptibility* of data variation and distribution. As to *data variation*, its confusion is prevented by the POP mechanism, with which different similarity-based clusters of the data can be computed (and selected) on six levels of granularity with the DPA described above. By starting with level 5, this corresponds to having a coarse overview at the beginning.

The perception of *data distribution* is facilitated by four different kinds of *locational clusters* and their distinctive symbolization on preview maps, and by offering four corresponding options of presentation. All of them share the use of *color* for the symbolization of variant types, as the use of specific symbols would easily lead to visual cluttering for higher numbers of types (and is therefore forbidden in the face of our data variety).[21]

The first option is *place*-oriented data clustering ("Ortspunktdarstellung"), according to which the data of a place (from up to four persons) are clustered and presented as a small, point-like pie chart (the pie chart showing corresponding color portions of the data distribution, as well as the number of persons), as in Figure 1, Figure 10 and Figure 12.

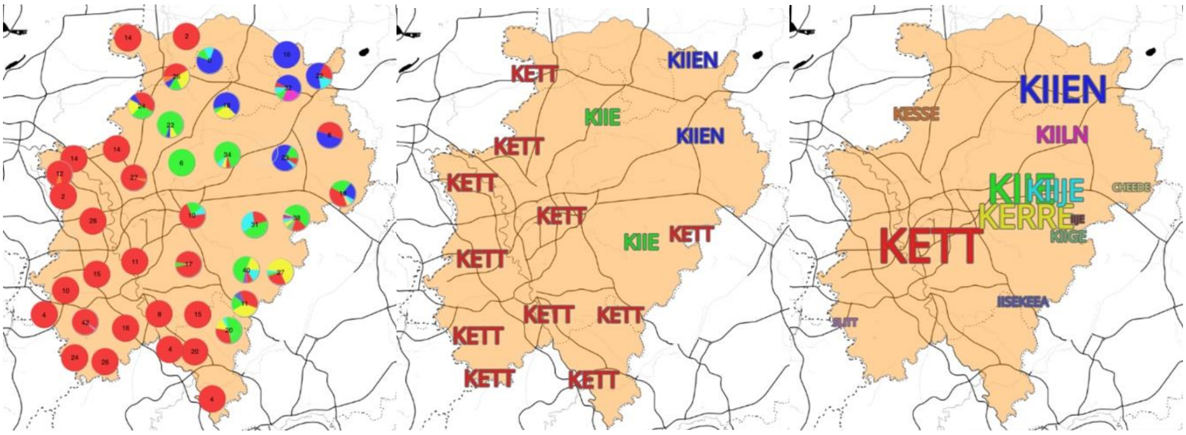
[21] The perception of the (up to twelve) colours on word maps used to distinguish types is enhanced by the selection of high-contrastive ones for this purpose. This use of colours therefore is a basic feature of the DMW system, and marks our secondary colour palette as a stopgap for people with disabilities in colour perception.

The second option uses Open Layers' facility of *cluster maps*, by which data from multiple places are automatically clustered according to some scheme (grouping via distance to some raster points, whose application on different zoom levels leads to cluster sets of varying granularity). Here, the same, but bigger, pie charts are employed ("Tortendarstellung"), only that the data are gathered from the places of the respective cluster (see Figure 14.a).

While such cluster maps combine detailed information presentation with some scheme of visual abstraction, they still might be confusing, because the color-type relationship must be kept in mind or looked up, and may distract from the gist of the data distribution. Hence, the third option of *text cluster maps* allows to only show the main (POP) type of each cluster as colored text, which allows a constricted, but intelligible, raw view on the data distribution at a certain zoom level (see Figure 14.b). In case of prominent secondary types (more than 40% portion), they are each displayed directly below the primary ones.

What is still missing, is a *wholistic* view on data distribution that abstracts from different-zoomlevel variation. For this reason, the fourth option clusters *all* place data of a variant type, computes the spatial center ('centroid') of the cluster ("Zentroiddarstellung"), and presents the colored POP-type text of the cluster at that point (text size varying with the number of variants), as in Figure 14.c. This allows to display the relative position of different variant types' bulks of variants even in cases where they widely overlap.[22]

Figure 14 a. Cluster b. Text cluster c. Centroid



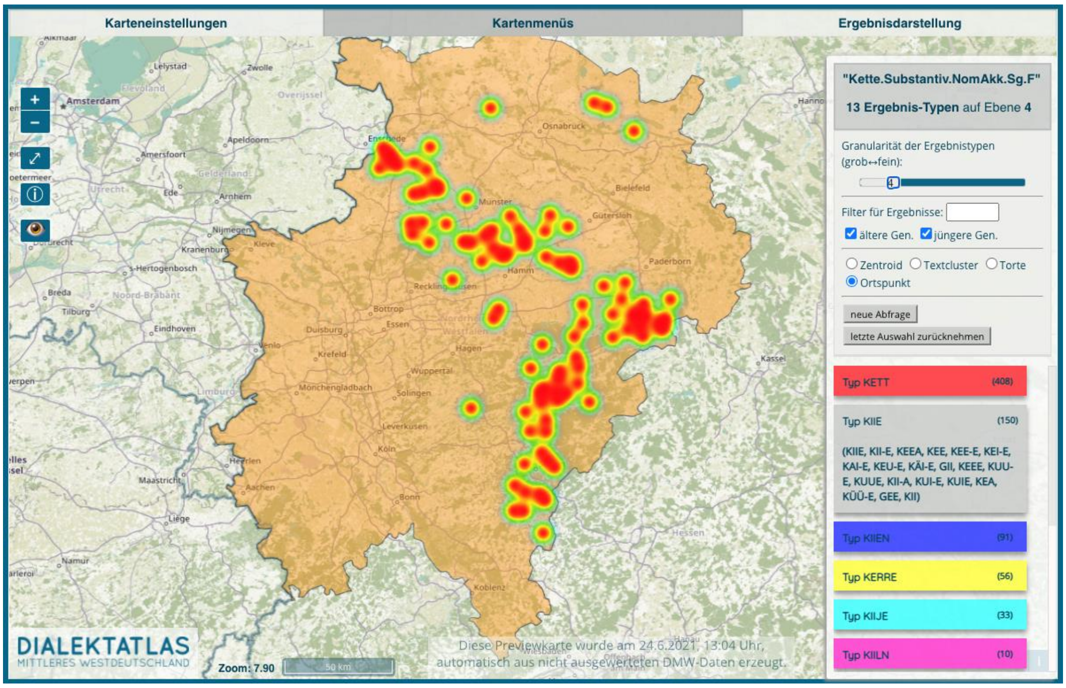
Aside from these word-map means of facilitating distribution perception, reference point maps can be used to recognize fine-grained differences and similarities in the data.

Another means to allow for better perceptibility of the data is *data selection for presentation*. The map interface offers three facilities to do that. The first is *momentary variant type selection* by the use of Open Layers' *heat maps*. They are generated if a user does a mouseover on a variant/POP type field in the legend. In a heat map, the places of the selected POP type are foregrounded with a blurred color scheme (red core, yellow inner border, green outer border). In the case of neighbouring places, this results in a confluent, intensified red coloring, highlighting that region (and therefore marking relative

[22] Again, this computation is window-dependent. Especially when zooming out, therefore, "centroid" view must be re-selected.

distribution). At the same time, all other information is hidden (compare Figure 15 and Figure 16). On mouseout, everything is restored.

Figure 15 *Heatmap of POP type KIIE of* ‘Kette’ *(level 4)*



Diese Previewkarte wurde am 24.6.2021, 13:04 Uhr, automatisch aus nicht ausgewerteten DMW-Daten erzeugt.

"Kette.Substantiv.NomAkk.Sg.F"
13 Ergebnis-Typen auf Ebene 4

Granularität der Ergebnistypen (grob↔fein): 4

Filter für Ergebnisse: □
☑ ältere Gen. ☑ jüngere Gen.
○ Zentroid ○ Textcluster ○ Torte ● Ortspunkt

Typ KETT (408)

Typ KIIE (150)
(KIIE, KII-E, KEEA, KEE, KEE-E, KEI-E, KAI-E, KEU-E, KÄI-E, GII, KEEE, KUU-E, KUUE, KII-A, KUI-E, KUIE, KEA, KÜÜ-E, GEE, KII)

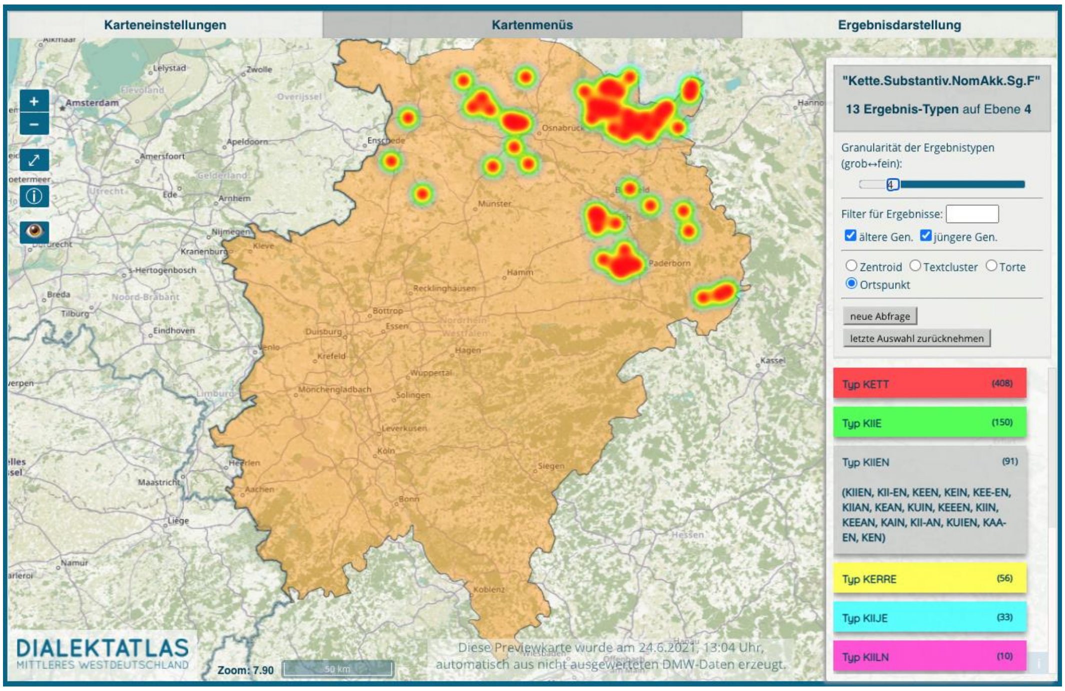
Typ KIIEN (91)

Typ KERRE (56)

Typ KIIJE (33)

Typ KIILN (10)

Figure 16 *Heatmap of POP type KIIEN of* ‘Kette’ *(level 4)*



Diese Previewkarte wurde am 24.6.2021, 13:04 Uhr, automatisch aus nicht ausgewerteten DMW-Daten erzeugt.

"Kette.Substantiv.NomAkk.Sg.F"
13 Ergebnis-Typen auf Ebene 4

Granularität der Ergebnistypen (grob↔fein): 4

Filter für Ergebnisse: □
☑ ältere Gen. ☑ jüngere Gen.
○ Zentroid ○ Textcluster ○ Torte ● Ortspunkt

Typ KETT (408)

Typ KIIE (150)

Typ KIIEN (91)
(KIIEN, KII-EN, KEEN, KEIN, KEE-EN, KIIAN, KEAN, KUIN, KEEEN, KIIN, KEEAN, KAIN, KII-AN, KUIEN, KAA-EN, KEN)

Typ KERRE (56)

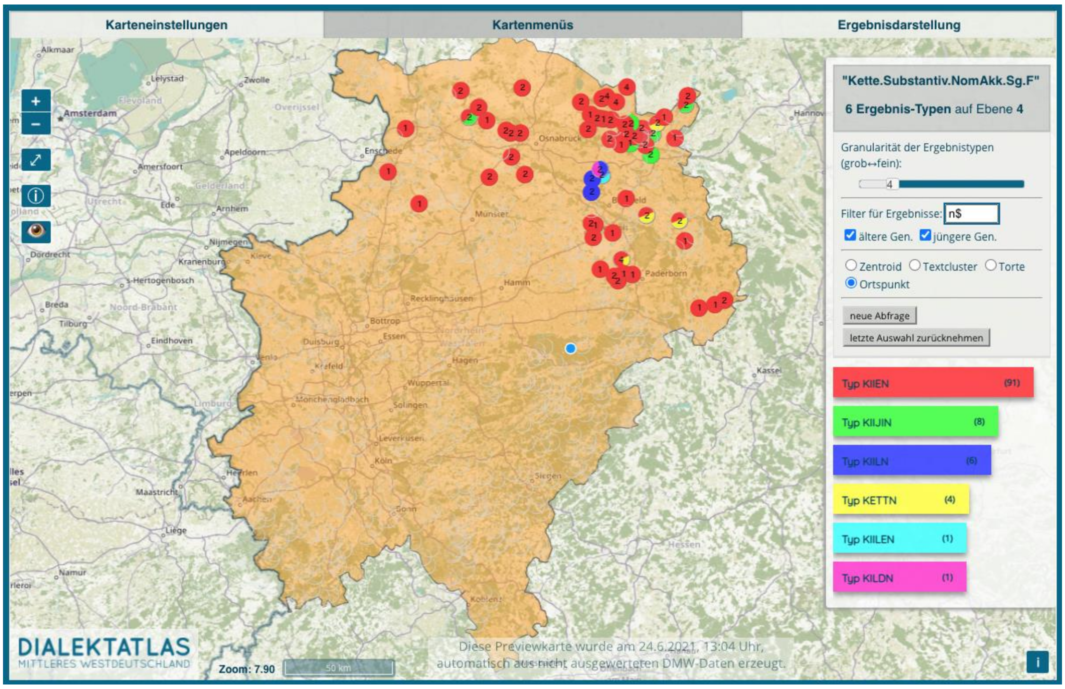
Typ KIIJE (33)

Typ KIILN (10)

The second is *permanent variant type selection* by double-clicking a POP field in the legend, given some granularity level. In this case, presentation is restricted to the data corresponding to the current-level POP type in question. This is especially useful for the investigation of a certain class of variants, before lowering the granularity level (again). The selection remains active until it is explicitly taken back by clicking on “letzte Auswahl zurücknehmen” (“Undo last selection”).

The third aspect is *filtering*. This is done by entering regular expressions in the corresponding field of the legend, which are used to filter the set of POPs before (re)application of the DPA. Filtering can be performed on any level of granularity (see Figure 17 for a *Kette* word map with the filter set to “ends with a ‘n’” on granularity level 4).

Figure 17 *Word map* Kette *restricted to variants ending with* ‘n’



"Kette.Substantiv.NomAkk.Sg.F"

6 Ergebnis-Typen auf Ebene 4

Granularität der Ergebnistypen (grob↔fein): 4

Filter für Ergebnisse: n$

☑ ältere Gen. ☑ jüngere Gen.

○ Zentroid ○ Textcluster ○ Torte ● Ortspunkt

Diese Previewkarte wurde am 24.6.2021, 13:04 Uhr, automatisch aus nicht ausgewerteten DMW-Daten erzeugt.

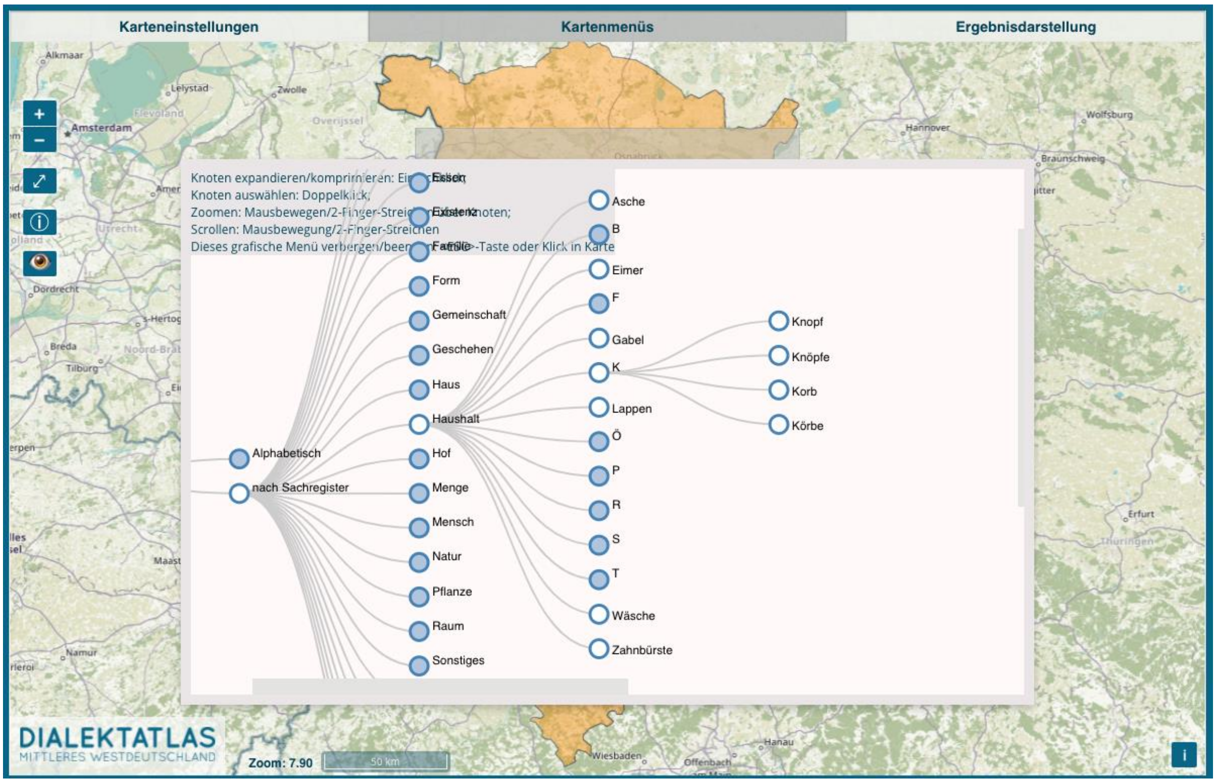
Needless to say that these means of dealing with the perceptibility of data variation and distribution can be used in combination. For example, by setting a granularity level, clicking on a POP type field, setting a lower granularity level, and finally entering a filter expression. By undoing last actions and entering a different part/path of the possible options, this procedure corresponds to *exploring* the distribution of some RWL’s variants.

5.2 Interaction

There are some requirements regarding user interaction in the map interface. First of all, this includes aspects of *usability* and ergonomics, given the two basic classes of users (lay people, experts): especially, an intuitive, easy handling of the interface, which should not only be (visually) parsimonious, but also self-explanatory at best. Then, the map interface should allow *different perspectives and choices*, as well as individual preferences for one or the other. Finally, interaction should also be *explorative*. This requirement for visualization applications can be described by the following steps of visual analytics (Keim et al. 2010): Provide *overview* (most important things first); offer *intuitive options* (for example, zooming/filtering, restricting, focusing, navigating); provide *details on demand*; show *relationships*; *easy switching* between aspects to be displayed. We tried to meet these needs as follows.

First, there is no distinction between a query page (specifying the type of data to be presented) and a map page (which was the case in the small precursor project SiSAL, see Solau-Riebel and Vogel 2013-2016), simplifying interaction. Instead, the centered query menus –being the currently focused elements of interaction– are always displayed *on top* of the geo-centered map and data layers (even with a graphical selection menu on top of the standard query menu, see Figure 18). Likewise, removing the top level corresponds directly to falling back to the next important lower level, or established state of presentation. Interaction is further facilitated by offering additional shortcuts to close menus and popups.

Figure 18 *Graphical selection menu based on a D3-style hierarchy*



Knoten expandieren/komprimieren: Einfachklick;
Knoten auswählen: Doppelklick;
Zoomen: Mausbewegen/2-Finger-Streichen über Knoten;
Scrollen: Mausbewegung/2-Finger-Streichen
Dieses grafische Menü verbergen/beenden: <ESC>-Taste oder Klick in Karte

Second, the *complexity* of the menus is simple by default, but changes on demand. For example, our basic query menu of word maps contains only three options (entering a word, selecting an ontological domain, and selecting from a word list). However, there is a check box allowing for the display of more options (selecting from a lemma list, choosing a part of speech (which will show pertinent suboptions), and choosing from a graphical menu). The Wenker menu simply lists all wenker sentences to choose from with check boxes, yet entering parts of words acts as a filter and immediately leads to a reduced check list. In general, menus of a certain type not only (dis)appear on mouse click, but also on certain mouseovers or keyboard entries, where appropriate. Correspondingly, users may develop their own preferences how to interact with the system.

Third, we follow the principle of successive refinement of a query by offering ontological or linguistic categories to select from (in part, hierarchically). Technically, this is supported by automatically restricting *all* options to the remaining compatible values after some selection ("self-restricting options"). To further improve performance, specifying a query is detached from database lookup, the latter being triggered only if some condition is met (for example, “only one item left in the list of available words” on a word map). Query specification is therefore a light and quick process, which, with the possibility of undoing the last choice made (by clicking “letzte Auswahl zurücknehmen”), can be performed exploratively, too.

Fourth, *explorativity* is at the heart of the map interface (as has already been shown). For example, both the query selection and the granularity mechanism start with an overview; mouse over a POP field only shows the corresponding places (as a heat map) of that variant; detailed information about the data represented by a pie chart can be requested and is presented in a popup (percentage of the variant types at that location, with absolute numbers of places and persons); the relationship of similar pronunciation is made visible on reference point maps; setting a different background map or using a different kind of map is only one click away, and likewise –apart from the four data distribution display options–, zooming in or out in cluster maps instantaneously yields different perspectives on the data. A final aspect of comfortable explorativity is the use of comparison maps, where selected variants of possibly different words can be easily specified and compared.

There is a qualification regarding usability, however. Although our interaction scheme works well ergonomically, both with respect to content and query, it is far from being self-explanatory. Instead of abandoning our interaction principles, however, we simply decided to offer videos of the basic functionality at the entry to the map interface, and an elaborate help menu. This design choice for a steeper learning curve, but long-term interaction benefit, corresponds to a preference for multiple-time visiting interested users of the map interface.

5.3 Technical aspects

The basic technical requirements of our map interface can be summarized as follows: to be able to automatically generate dynamic preview maps with state-of-the-art web software components; to have a responsive application, both with regard to query and presentation; and to have a persistent implementation to some extent.

We currently use a Parcel-bundled Open Layers 5.3.0 ES6-style Javascript application featuring JQuery(-UI) and D3 npm-organized modules[23] that accesses a MySQL database for text data, and provides links to the scientific cloud Sciebo for audio data. For tuning the selection, and presentation, of values from lists for certain input fields, we use the *awesomplete* autocompletion package by Lea Verou. The chores of linguistic data preparation for the query menu options (e.g., for hierarchical graphical menus like the one in Figure 18, but especially for the phenomena-related options of the expert version not described here) are performed separately with Node.js. As to map generation, we needed to use Voronoi cells of the explored places for our reference point maps (to provide for colorable areas). These cells (which are generated in QGIS® and imported as kml data) are also used to react tolerantly on place clicks in word maps. For fluid interaction, the map interface is heavily event-driven, minimizing clicking effort (for example, using mouseover for the initiation of some action, or

[23] Not loading such components via – potentially defunct – links to software sites guarantees availability of the used software components, and hence, some persistence of the map interface. For long-term persistence, we may use a Docker-style implementation of the DMW system.

automatically focusing on an input field if its parent is foregrounded). In addition to that, database calls are minimized, at least for the present map types.

6. Conclusion

Starting with an overview of the data and work flows, and of transcription, in the DMW project, this article described the algorithmic and technical aspects of the project's preview word map generation. It was shown how the problem of massive dialect data variation can be overcome by using a multi-level Soundex-like indexing scheme of the transcribed observants, ultimately achieving effective, digital, dynamic, interactive visualization. In the course of this, different aspects of preview word maps were explained, and examples for their explorative use in what is called “Speaking DMW” or “Dynamic atlas maps of the DMW project” were presented.

7. Acknowledgments

I would like to thank Aynalem Misganaw (from the center of information and media technology, ZIMT, of the University of Siegen) for her substantial database-related work, which made all this possible. She implemented all front ends except the map interface. I am grateful to Hans Goebl for fruitful discussions. I also thank Ambra Ottersbach and Nadine Wallmeier for helpful comments on versions of this paper. Finally, thanks go to all (student) colleagues whose feedback has helped shaping the DMW system to what it is now.

8. References

Bieberstedt, Andreas, Ruge, Jürgen and Schröder, Ingrid 2016: Hamburger Transkriptionskonventionen. In Bieberstedt, Andreas, Ruge, Jürgen and Schröder, Ingrid (eds.): *Hamburgisch. Struktur, Gebrauch, Wahrnehmung der Regionalsprache im urbanen Raum*. Frankfurt am Main u. a. (Sprache in der Gesellschaft, 34), 421–428.

Carstensen, Kai-Uwe, Spiekermann, Helmut, Tophinke, Doris, Vogel, Petra M. and Wich-Reif, Claudia 2020: Zur Methodik des Dialektatlas Mittleres Westdeutschland (DMW). *Korrespondenzblatt des Vereins für niederdeutsche Sprachforschung* 127: 107–114.

Draxler, Christoph and Jänsch, Klaus 2004: SpeechRecorder – a Universal Platform Independent Multi-Channel Audio Recording Software. In: *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, 559–562.

Draxler, Christoph and Jänsch, Klaus 2019: SpeechRecorder.
URL: https://www.bas.uni-muenchen.de/Bas/software/speechrecorder/ (last accessed 04.06.2022).

Gehrke, Gero, Kuhmichel, Katrin, Sauermilch, Stephanie and Wallmeier, Nadine 2020: Dialektatlas Mittleres Westdeutschland (DMW) – Methodik, Akquise, Exploration und Analyse. *Niederdeutsches Wort*. 60: 7–33.

Goebl, Hans 2010: Dialectometry and quantitative mapping. In: Lameli, Alfred, Kehrein, Roland and Rabanus, Stefan (eds.): *Language and Space. An International Handbook of Linguistic Variation, vol. 2: Language Mapping* (Handbücher der Sprach- und Kommunikationswissenschaft [HSK] 30.2.), Berlin: de Gruyter 2010; 1st part: 433-457 (text), 2d part (maps): 2201–2212.

Haimerl, Edgar 2005: Taxierungsalgorithmen. In: *Quantitative Linguistics; an International Handbook*. Eds. Köhler, R. Altmann, G. De Gruyter. 532–547.

Keim, Daniel, Kohlhammer, Jörn, Ellis, Geoffrey and Mansmann, Florian (eds.) 2010: *Mastering the information age: solving problems with visual analytics*. Goslar: Eurographics Association.

Lameli, Alfred, Kehrein, Roland and Rabanus, Stefan (eds.) 2010: *Language and Space. An International Handbook of Linguistic Variation, vol. 2: Language Mapping* (Handbücher der Sprach- und Kommunikationswissenschaft [HSK] 30.2.), Berlin: de Gruyter.

Ruge, Jürgen 2019: *Hamburger Transkriptionskonventionen: lautnah – lesbar – auf Modifikation ausgelegt* [Hamburgian transcription conventions: close to phonation – readable – open to modification]. Talk given on the DMW Workshop „Methoden der Transkription und Transliteration dialektaler Daten“ [“Methods of transcription and transliteration of dialect data”] (Münster, 09.08.2019).

Schmidt, Jürgen Erich, Herrgen, Joachim and Kehrein, Roland (eds.) 2008ff.: *Regionalsprache.de (REDE). Forschungsplattform zu den modernen Regionalsprachen des Deutschen*. With the collaboration of Dennis Bock, Brigitte Ganswindt, Heiko Girnth, Simon Kasper, Roland Kehrein, Alfred Lameli, Slawomir Messner, Christoph Purschke, Anna Wolańska. Marburg: Forschungszentrum Deutscher Sprachatlas.

Solau-Riebel, Petra and Vogel, Petra M. 2013-2016: Siegerländer Sprachatlas (SiSAL). URL: http://www.mundart.sisal.uni-siegen.de (last accessed: 01.05.2021).

Spiekermann, Helmut H., Tophinke, Doris, Vogel, Petra M. and Wich-Reif, Claudia (eds.) 2016ff.: Dialektatlas Mittleres Westdeutschland (DMW). Siegen: Universität Siegen [URL: https://www.dmw-projekt.de/].

Wells, J.C. 1997: SAMPA computer readable phonetic alphabet. In: Gibbon, Dafydd, Moore, Roger and Winski, Richard (eds.): *Handbook of Standards and Resources for Spoken Language Systems*. Berlin and New York: Mouton de Gruyter. Part IV, section B. 684–732.

Wieling, Martijn, Nerbonne, John and Baayen, R. Harald 2011: Quantitative Social Dialectology: Explaining Linguistic Variation Geographically and Socially. *PLoS ONE*, 6(9): e23613. doi:10.1371/journal.pone.0023613.

Wilz, Martin 2005: *Aspekte der Kodierung phonetischer Ähnlichkeiten in deutschen Eigennamen*. Magisterarbeit an der Philosophischen Fakultät der Universität zu Köln.