The Romani Morpho-Syntax (RMS) database

Yaron Matras, Christopher White and Viktor Elšík

1. Background and aims

Despite having now become the largest minority language in the European Union – with upwards of 3.5 million speakers dispersed mainly in central and southeastern Europe – Romani is still considered one of the continent's lesser-known languages. Yet interest in the language is prompted by its very special position in a number of areas: its history – Romani is the only modern Indo-Aryan language that has been spoken exclusively in Europe since the middle ages; its geography – Romani is exceptional in not covering a coherent territory, but rather being dispersed in 'diaspora' communities, often characterised by repeated migrations; its structural-typological characteristics – Romani dialects have absorbed structural influences from a variety of different languages, and in the absence of a unifying standard, have developed in diverse directions; and its socio-political status – with growing European integration, efforts are underway to take into consideration the special needs of the Romani people at various levels, and this includes expanding the usage domains of the Romani language.

In all these areas, a comparative approach to the diverse dialects of Romani is essential: In the absence of written documentation on earlier stages of the language, reconstruction relies on a comparative study of the dialects. The comparative sample of Romani dialects provides an opportunity to observe regularities of structural change, including contact-induced change (see Elšík and Matras 2006). Applied questions of language codification, standardisation, and the mutual comprehensibility of Romani dialects are also best addressed by comparing lexical and grammatical structures.

These considerations were behind the creation of a central corpus of Romani dialects that would facilitate structural comparison among them. Work on the RMS (Romani Morpho-Syntax) database began in 1998, with the intention of creating an electronic resource that would store both linguistic data, and 'metadata' in the form of answers to analytical questions, and so would allow queries on entire sets of data. Organised in a format resembling a grammatical description, and aiming to cover all aspects of structural variation among the dialects, RMS is quite possibly the only existing comprehensive comparative grammar in electronic form. It is also one of the larger projects of its kind. Its development has been supported by grants from the Arts and Humanities Research Board (AHRB), the Economic and Social Research Council (ESRC), and the Open Society Institute (OSI), with a total accumulated budget of around £565,000 (€840,000). In various phases, the project has so far employed three co-workers – a Research Associate, a Programmer, and an Archive Manager – on a full-time basis, around a dozen part-time research and technical assistants, and around 50 part-time fieldwork assistants working in altogether 20 different countries. The project's data archive now contains some 300 original recordings, as well as data extracted from numerous published sources (grammatical descriptions and texts). An earlier form of the database has been accessible online to a small circle of researchers specialising in Romani via a special server since 2001. It has served as a data basis for several monograph-length comparative investigations of Romani, including Matras (2002), Boretzky and Igla (2004), and Elšík and Matras (2006), and is currently providing a data management frame for several ongoing PhD dissertations in Romani linguistics, at several different institutions. At the time of writing, the database is undergoing a technical transformation to a new application with a web interface, which will gradually become publicly accessible via the project's website: http://romani.humanities.manchester.ac.uk/.

In the present contribution, we outline the aims, scope and content structure of the database, data collection strategies, the different phases in the technical development of the resource, the query structure, and future prospects. Other brief descriptions of RMS can be found in some of our earlier work – Matras (2004a: 281–285) and Elšík and Matras (2006: 55–64) – as well as on the project website.

2. The linguistic investigation of Romani

Proto-Romani – the term given to development phases of the language in its pre-European period – appears to have originated in the Central areas of India, during the early transition period from Old to Middle Indo-Aryan (300 BC-500 AD). As pointed out already by Turner (1926), Romani shares ancient innovations from this period with other Central languages of India, such as Hindi/Urdu and Punjabi, whereas developments from a later date – the transition period to Early New Indo-Aryan (ca. 500 AD-800 AD) – are shared with the languages of the Northwest, such as Kashmiri (see Matras 2002, ch. 3). These include on the one hand archaisms, which were retained

in the Northwest, but not in the Central languages (such as the presence of certain consonant clusters, e.g. *tr-* in *trin* 'three'); general innovations that encompassed the entire Indo-Aryan speaking region (such as the reduction of nominal case and inflected past tense of the verb); as well as innovations that are limited to the Northwest (such as the development of a new person concord system in the past tense). This evidence points to an early migration history within India, even before the language left the subcontinent. Later phases in Proto-Romani are characterised by unique innovations, while in some domains the language maintains Middle Indo-Aryan archaisms: e.g. the persistence of a consonantal present-tense conjugation and consonantal forms of nominal case-endings. Already Pott (1844–1845) drew attention to the layers of Iranian, Armenian and Greek loanwords, which characterise later phases of Proto-Romani (outside of India) and which arguably constitute evidence of prolonged contacts with the respective western Asian populations.

The immense lexical and grammatical impact of medieval Greek, first highlighted by Miklosich (1872–1880), is now accepted as the beginning of a new stage in the language – called Early Romani – which was characterised by the structural-typological Europeanisation, or specifically *Balkanisation* (Matras 1994) of Romani. Early Romani is regarded as the precursor of the modern dialects of Romani, which emerged gradually following the dispersion of Romani-speaking populations across Europe in the period paralleling the decline of the Byzantine Empire, from around 1350 onwards. The earliest written attestations of Romani from around 1542 (Britain), 1570–1597 (Germany and France), and 1668 (Thrace), and numerous sources from the early 1700s, already represent the kind of dialectal variation found in Romani today, while the geographical distribution patterns of structural variants seems to point largely to developments *in situ*, rather than 'genetic' inheritance (although this point remains controversial in Romani linguistics). From this one might conclude that the bulk of developments separating the dialects occurred during the period of settlement (which followed the period of migration), in the 16th century (see Matras 2005).

Miklosich's dialectological work on Romani divided the dialects based on a similar assumption, according to the peoples amongst whom the Roma had settled. This tradition was broken by Gilliat-Smith (1915), who described the geographical overlap of distinct dialect groups in northern Bulgaria, highlighting the need to take successive migrations and continuous networking among historically related groups into account. During most of the 20th century, classification work in Romani dialectology relied on loose impressions of structural similarities, recognising geographically proximate groups on the one hand, as well as isolated, out-migrant offshoots of those groups on the other. Recently, with the availability of a larger dataset and some intense advances in the geographical plotting of linguistic features, a debate between two interpretations has occupied the centre stage in Romani linguistics: the first attributes regional differences to the diffusion of innovations in geographical space following settlement (Matras 2002, 2005), the second attributes them to older – so-called 'genetic' – differences that existed prior to settlement, and that were brought to their current locations by groups or tribes speaking distinct dialects (Bakker 1999, Boretzky 1999a and 1999b, Boretzky and Igla 2004). In evaluating the evidence, the role of identifying archaisms vs. innovations is of course crucial. In the absence of historical documentation, the procedure for establishing the position of a feature relies largely on a comparative interpretation of the datasets.

Other issues arising in recent years from the analysis of Romani include the potential for typological drift and change in a language that is strictly oral and enjoys little institutional support and so no regulation either; and the extent and quality of contact influence in a language whose adult speakers are all bilingual, which is a marginalised and often oppressed language, limited to basilectal functions. Its dialects being in contact, under comparable socioliguistic conditions, with dozens of languages as far apart as Basque, Welsh, Finnish, Croatian, Hungarian, and Turkish, Romani provides an excellent sample with which to study the lexical and structural effects of language contact.

3. The RMS agenda and implementation strategy

Aiming to provide a tool to facilitate research into such areas, the RMS database was created with the following domains of analysis in mind:

1) Historical, aiming to compare dialect specific innovations, and so to cover a dimension that is specific to Romani, focusing on developments of form to form, and from form to function;
2) Typological, aiming to examine the structural representation of functions across a sample of dialects, thus covering relations between function and form, and among clusters of functions;
3) Contact-theoretical, aiming to examine contact influences, and so, for this purpose, tagging structures by etymology and etymological layers (representing 'depth' of borrowing);

4) Dialectological, aiming to examine the link between innovations and their geographical distribution in what is considered to be a non-territorial (insular) language, thereby critically addressing the notion of a 'genetic' classification of dialects.

The initial phase of the database construction aimed therefore at covering in maximum depth questions of variation among the dialects that could inform the above domains of investigation. This involved, in the first phase, in-depth comparative research into the dialects, drawing on all available published descriptions (using, in practice, around 40 monograph-length and some 20–30 article length publications on individual dialects). For each grammatical domain, lists of variants were plotted, giving a general inventory of possible forms. These would constitute the backbone of the *form* slots, eliciting the formal representation of morphemes. The tool used in this initial phase was FileMakerPro, a user-friendly database application; this tool was subsequently abandoned, however. We return to the technical side of the database construction below.

The compilation of variants from the literature led at the same time to a comparative analysis, and a historical analysis, of the emergence of certain categories, leading in turn to the plotting of *form-to-form* fields – those representing the shape, for a particular dialect, of an inherited form – as well as the *form-to-function* fields – those representing the dialect-specific function of an inherited form. Thus, a slot was devoted to the hypothesised Early Romani indefinite form **khajek*, asking a) whether it is continued in the dialect (i.e. presence of the *form*), b) its shape in the dialect, e.g. *kajek* or possibly *kek* (i.e. *form to form*), and c) its function in the dialect, e.g. general determiner 'some-' or 'no-', or person indefinite 'somebody' or 'nobody' (i.e. *form to function*).

Consider in more detail an example of the form-to-function perspective. Romani dialects inherit two forms of the present stem: A short form, in which the final morpheme indicates person concord (1SG *-av* etc.), and a long form, where the suffix *-a* attaches to the person concord morpheme (1SG *-av-a* etc.). It appears that the long form served as a present-future in Early Romani, while the short form was the subjunctive. The dialects continue both forms, but alter their functions, often in connection with the introduction of an analytical future category. Figure 1 shows the distribution in some dialects.

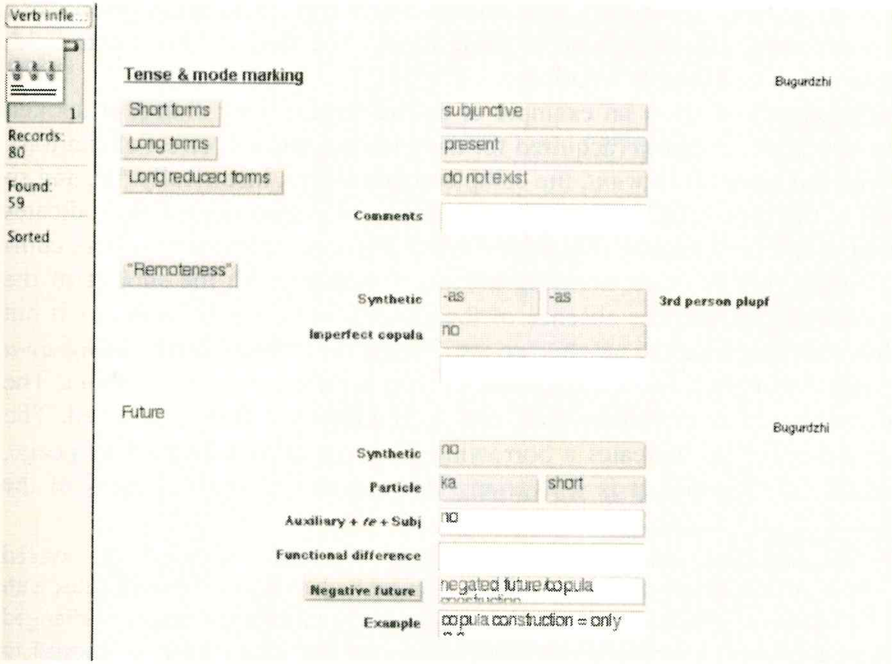
Noteworthy is the geographical distribution of the developments: In the Balkans (Sepečides, Rumelian Romani, Kosovo Bugurdži, Florina Arli), the long forms are confined to the present indicative, and the future is expressed by a future particle (followed by the subjunctive). In central Europe (Lovari, Rumungro, and Roman), the short forms take over also a present indicative meaning, while the long form specialises for future. Serbian Kalderaš shows contamination of the central European pattern with the Balkan pattern. The original state of affairs is preserved in the western, German-French and Scandinavian dialects. Elsewhere, combinations are found: an ongoing shift in the expression of the present indicative from long to short forms, combined with a loss of the future meaning of the long forms only through the introduction of an analytic future in Russian Romani.

DIALECT	SHORT FORM	LONG FORM	FUTURE PARTICLE
Sinti, Manuš	subjunctive	present-future	–
Finnish R	subjunctive	present-future	–
Latvian R	present-subjunctive	present-future	–
Welsh R	present-subjunctive	present-future	–
Rumungro	present-subjunctive	future	–
Roman	present-subjunctive	future	–
Lovari	present-subjunctive	future	–
Serbian Kalderaš	present-subjunctive	future	*ka*
Sepečides	subjunctive	present	*ka*
Rumelian R	subjunctive	present	*ka(m)*
Kosovo Bugurdži	subjunctive	present	*ka(m)*
Florina Arli	subjunctive	present	*ka*
Russian R	present-subjunctive	present	*l-*

Figure 1. Inherited present-stem forms and their TAM function in some dialects

The database organisation in the original FileMakerPro format captures such data by allocating fields in a layout devoted to 'Verb inflection' to 'Tense and mode marking', asking for the function of each of the anticipated Present-tense forms (short form, long form), and continuing to elicit the strategies used to mark the Future tense. Each field carries a value list, comprising all variants that have been collected during the pilot study, and so all anticipated variants. The list is open, and new forms can be added to it, if encountered in the data. Thus a query can select any of the attested forms and search for particular data, or else simply look up the data that has been entered into the relevant field for a respective dialect record or set of records (see Figure 2). The organisation of questions and content of the data fields displayed in Figure 2 are typical of the form-to-function approach.

The second approach is the *function-to-form* procedure. Here, state-of-the-art typological descriptions and questionnaires (e.g. those emerging from the EUROTYP project, and other recent typological investigations) were taken into account in order to plot representation grids for the respective functions. One example is the continuum of semantic integration of complement clauses (cf. Matras 2004a). This is captured, following typological work on complementation such as that by Wierzbicka (1988), Givón (1990), Frajzyngier (1991), Frajzyngier and Jasperson (1991), and Dixon (1995), by a range of main clause predicates representing tighter and less tight event integration (such as *can*, *want*, *begin*, *try*, *fear* etc.), as well as the contrast between modality (*can*, *begin*, etc.) and epistemic complementation (*see*, *know*, *hear* etc.), and between identical subject and different-subject constructions (so-called manipulative predicates such as *demand*, *ask*, etc.).



Tense & mode marking

Bugurdzhi

Short forms: subjunctive

Long forms: present

Long reduced forms: do not exist

Comments:

"Remoteness"

Synthetic: -as -as 3rd person plupf

Imperfect copula: no

Future

Bugurdzhi

Synthetic: no

Particle: ka short

Auxiliary + *te* + Subj: no

Functional difference:

Negative future: negated future copula construction

Example: copula construction = only

Figure 2. Database excerpt 'Tense marking' in FileMakerPro 6 format

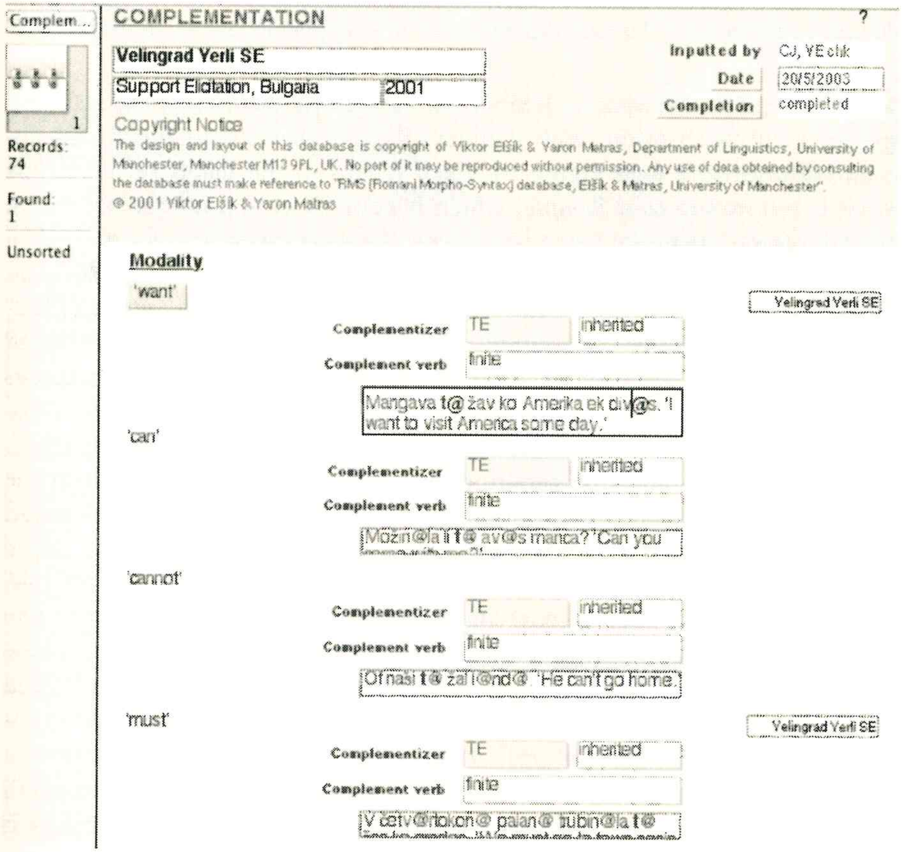
For each predicate, three value lists appear. The first contains a statement about the presence or absence of a complementiser conjoining the two clauses. The value options are 'none', or a choice of a complementiser type. This latter value is a Romani-specific form. Modal complements tend to take a non-factual complementiser of the type *TE* (realised in the individual dialects as *te*, *tə* or *ti*). Epistemic complements tend to take a complementiser of the type *KAJ* (usually realised as *kaj*), though this latter is often substituted by a borrowed particle. The next field identifies the origin of the complementiser, the value options being 'non-applicable' (in case a complementiser is absent), 'inherited', or a choice between several layers of borrowing: those from an Old contact language (no longer spoken in the community), a Recent contact language (still spoken by the older generation), or a Current contact language (spoken regularly by all members of the community).[1]

The final field characterises the inflection of the complement verb. The value options are 'finite' and 'non-finite'. Clause combining in Romani is overwhelmingly finite. However, in modal complements with identical subject constructions ('infinitive clauses'), some (mainly central European) dialects tend to generalise one of the person-inflected forms, thereby abandoning subject agreement, and introducing instead a kind of 'infinitive', based historically on one of the finite forms. The final field is a data field, into which an example is inserted.

Figures 3–4 show an example of entries for the *Yerli* dialect as spoken in Velingrad, Bulgaria (acquired for the database through direct elicitation). With the modal verb *want*, the complementiser is *tə*, historically **te*, and so *TE* is the type selected from the value list. The etymology field indicates that it is inherited (and so part of the pre-European component). The complement verb is finite, showing person agreement with the subject of the matrix clause, and the absence of the present/future suffix *-a* marks it out for the subjunctive: *dža-v* 'go-1SG'; cf. the matrix verb *mang-av-a* 'want-1SG-PRES'. For the verb *see* we find a different state of affairs. The complementiser *či* is borrowed, and so the concrete form is entered. The etymology field indicates a borrowing from the current contact language, which for this dialect is Bulgarian. The question of the finiteness of the verb is redundant in epistemic constructions, where no Romani dialect uses non-finite forms, and therefore it does not appear in the entry.

[1] The distinction, introduced in Matras (1998), is intended to capture the layered history of contact influences, which is relevant both to Romani communities with a history of migrations, and to those whose external prestige language changed as a result of historical circumstances (e.g. the shift from Ottoman Turkish to Bulgarian and Greek in the Balkans, or from Hungarian to German in some territories of the former Austro-Hungarian Empire).

To summarise, then, the initial database sketch consisted of an outline of likely variation and inventories of possible variants in the shape of forms, the semantic functions and the distribution of *inherited* forms, and the structural representation of semantic functions, including both the composition and etymology of the participating forms. These are displayed through two types of fields: those presenting actual linguistic data for exemplification, and those presenting questions about the data (e.g. "is a definite article retained?", "what is the function of short forms of the present tense?"). The primary purpose of the database is to allow the user to query the data by looking up the contents of any individual field or combination of fields, for any dialect or combination of dialects.



COMPLEMENTATION

Velingrad Yerli SE

Support Elicitation, Bulgaria 2001

Inputted by CJ, VE chk

Date 20/5/2003

Completion completed

Copyright Notice

The design and layout of this database is copyright of Viktor Elšík & Yaron Matras, Department of Linguistics, University of Manchester, Manchester M13 9PL, UK. No part of it may be reproduced without permission. Any use of data obtained by consulting the database must make reference to "RMS (Romani Morpho-Syntax) database, Elšík & Matras, University of Manchester".
© 2001 Viktor Elšík & Yaron Matras

Modality

'want' Velingrad Yerli SE

Complementizer TE inherited
Complement verb finite
Mangava t@ žav ko Amerika ek div@s. 'I want to visit America some day.'

'can'

Complementizer TE inherited
Complement verb finite
Možin@la i t@ av@s manca? 'Can you

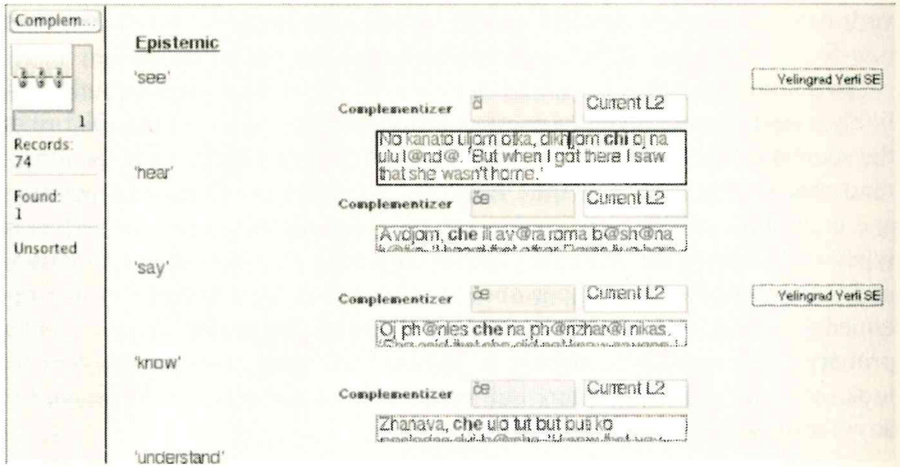
'cannot'

Complementizer TE inherited
Complement verb finite
Ofnasi t@ zal i@nd@. 'He can't go home.'

'must' Velingrad Yerli SE

Complementizer TE inherited
Complement verb finite
V četv@rtokon@ paian@ trubin@la t@

Figure 3. Extract in FileMakerPro 6 format on 'Complementation'/'Modality'



Epistemic

'see'

Complementizer č Current L2

No kanato uljom olka, dikhjom chi oj na ulu l@nd@. 'But when I got there I saw that she wasn't home.'

'hear'

Complementizer če Current L2

Avdjom, che ili av@ra roma b@sh@na

'say'

Complementizer če Current L2

Oj ph@nies che na ph@nzhar@i nikas.

'know'

Complementizer če Current L2

Zhanava, che ulo tut but buti ko

'understand'

Figure 4. Extract in FileMakerPro 6 format on 'Complementation'/'Epistemic'

From the user's viewpoint, RMS is structured in the form of a standard grammatical description, with distinct chapters devoted to functional domains of structure (see Figure 5). Each record in the database represents what is referred to as a *Sample*, which is equivalent to a unique source on the language. The initial batch of sources that were taken into account when first plotting the database fields were published descriptions of Romani dialects. The 'source' in these cases is the author, drawing on a corpus of material from a particular community, which quite often contains data elicited from a variety of speakers (though the type of grammatical sketch that is based exclusively or almost exclusively on one speaker is not rare in Romani dialectology). In the second phase, a questionnaire was constructed, covering all main areas of variation, and most data now contained in the database and RMS archive are the product of systematic fieldwork carried out throughout eastern, central, and southeastern Europe. Here, a *Sample* corresponds to a speaker as a source of data. Several speakers (as well as, where relevant, printed sources) may be grouped together to represent one *Dialect*. The degree of uniformity among unique samples representing one single Dialect thus becomes in itself subject to investigation, and indeed part of the future agenda and prospect of further development of the database tools (see below).

The questionnaire was composed through a careful consideration of all data fields, and inspired by the need to elicit data to be able to fill them. It is thus tailored to the database structure, which itself is the product of a prolonged investigation into variation and structural composition in Romani.

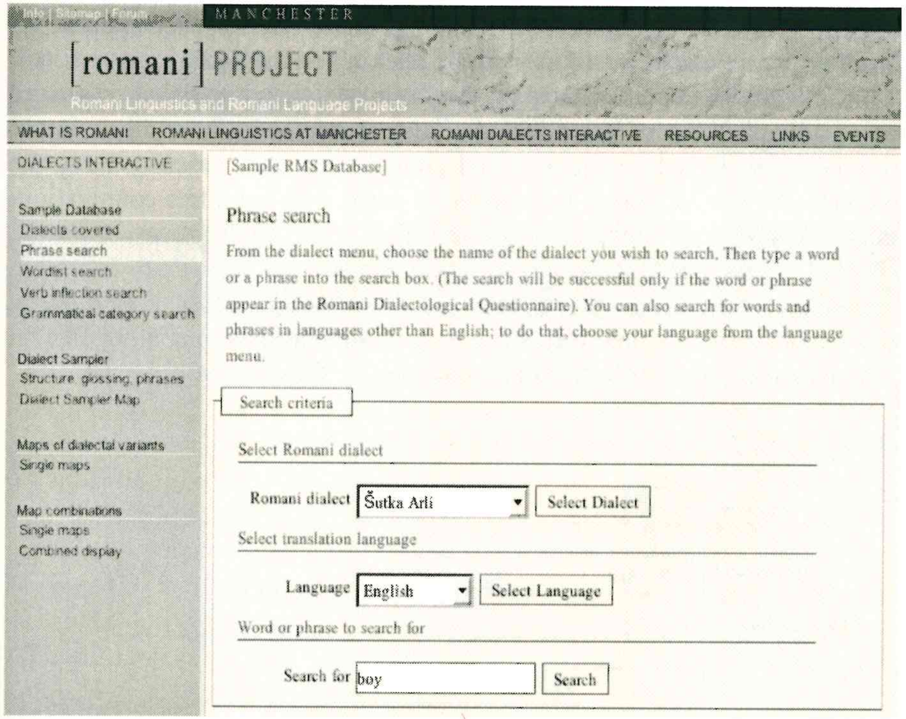
Like the database, the questionnaire addresses form-to-form, form-to-function, and function-to-form questions. All issues are built into either a set of some 850 short sentences, which constitute the bulk of the questionnaire, or a wordlist or a list of verbs to be inflected. The elicitation technique exploits the fact that all Roma are bilingual, and uses the majority language to elicit translations from the speakers of words, verb conjugations, and phrases. For this purpose, the questionnaire in its first version from 2001 has so far been translated into some 14 different languages. Much of the fieldwork has been carried out by graduate students specialising in Romani linguistics, and for this purpose networking workshops were set up, bringing together students from different institutions and different countries to discuss fieldwork methodology, transcription conventions, and so on. Additional fieldwork assistants were recruited among students of Romani background, who were equally invited to participate in training and instruction workshops.

General profile of the source
Noun inflection
Noun derivation
Adjective inflection
Adjective derivation
Adjectives
Numerals
Personal/reflexive pronouns
Demonstratives
Interrogatives and Relatives
Indefinites
Article inflection
Lexicon
Lexicophonetic features
Phonology
Verb inflection
Verb derivation
Verb adaptation
Copula inflection
Modals
Prepositions
Case Representation
Local relations
Temporal relations
Complementation
Embeddings and relative clauses
Adverbial clauses
Word order
Utterance modifiers

Figure 5. Chapters in the RMS database

The advantages of the questionnaire are obvious: It allows systematic coverage of structures in a way that cannot otherwise be guaranteed, and it makes data available for direct comparison between the dialects. For this, fieldworkers follow a uniform procedure. All interviews with speakers – of average duration of some four hours – are recorded, and the recordings digitised and archived, normally both as complete files, as well as cut into individual phrases. The informant's answers are transcribed (using Unicode fonts) onto a spreadsheet. Each phrase in the original questionnaire is tagged for the grammatical categories that it is intended to elicit. The tags range from exemplification of individual phonemes or particular inflection endings in words, through to word classes and entire semantic constructions such as types of clause combinations or case relations. Naturally, not each and very sentence is translated by the speakers as intended, and so there is an error margin in the actual ability of each recorded questionnaire to retrieve the intended constructions through the pre-built tags. But since each semantic function, construction and structure appear in several different positions through the questionnaire, retrieval is generally guaranteed, even if not via each and every intended phrase.

The tags are designed to answer the analytical questions that are dealt with in the database, and so they match chapters, sections, and indeed individual cells in the RMS database. The spreadsheets can thus feed directly into the database: In the earlier working phase, the link between spreadsheet data and the database in FileMakerPro 6 was based on manual retrieval of the data by sorting the spreadsheet rows according to the tags, and entering the relevant data into the database fields. During the recent development stage, a new database has been created, allowing the transcriptions to be fed directly into the database, which then retrieves them automatically as exemplification for individual data fields. Each word and phrase are also linked to the digital sound files, thus making all raw data – in transliteration and in original sound – directly accessible to the user. Figures 6 and onwards show extracts from the recent development of a Sample Database, modelling some of the functions of the new, upgraded RMS. This Sample Database has been freely accessible via the project website since January 2006. Note that users may choose one of several functions: Phrase search, Wordlist search, Verb inflection search, and Grammatical category search. The user can select a dialect (representing a particular Source Sample, i.e. the transcription and recording of an interview with an individual speaker). It is also possible to select languages other than English as input languages for phrases or pre-defined word searches. The Phrase search function retrieves any corresponding string within the transcription, thus covering words, affixes, and short phrases. Figure 6 illustrates the query input for the word 'boy' in English, in the Šutka Arli Romani dialect of Macedonia, while Figure 7 shows the query output.



[Sample RMS Database]

Phrase search

From the dialect menu, choose the name of the dialect you wish to search. Then type a word or a phrase into the search box. (The search will be successful only if the word or phrase appear in the Romani Dialectological Questionnaire). You can also search for words and phrases in languages other than English; to do that, choose your language from the language menu.

Search criteria

Select Romani dialect

Romani dialect: Šutka Arli

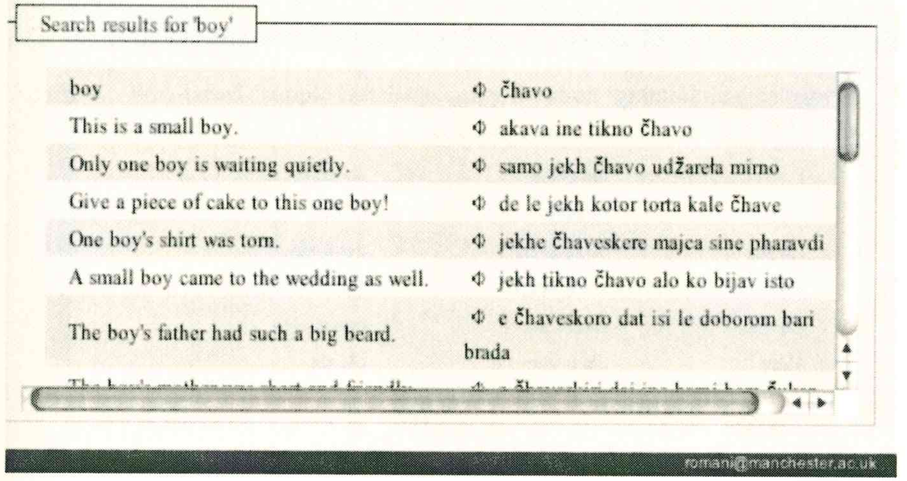
Select translation language

Language: English

Word or phrase to search for

Search for: boy

Figure 6. Web-based Sample Database query function 'Phrase search'
▷ http://romani.humanities.manchester.ac.uk



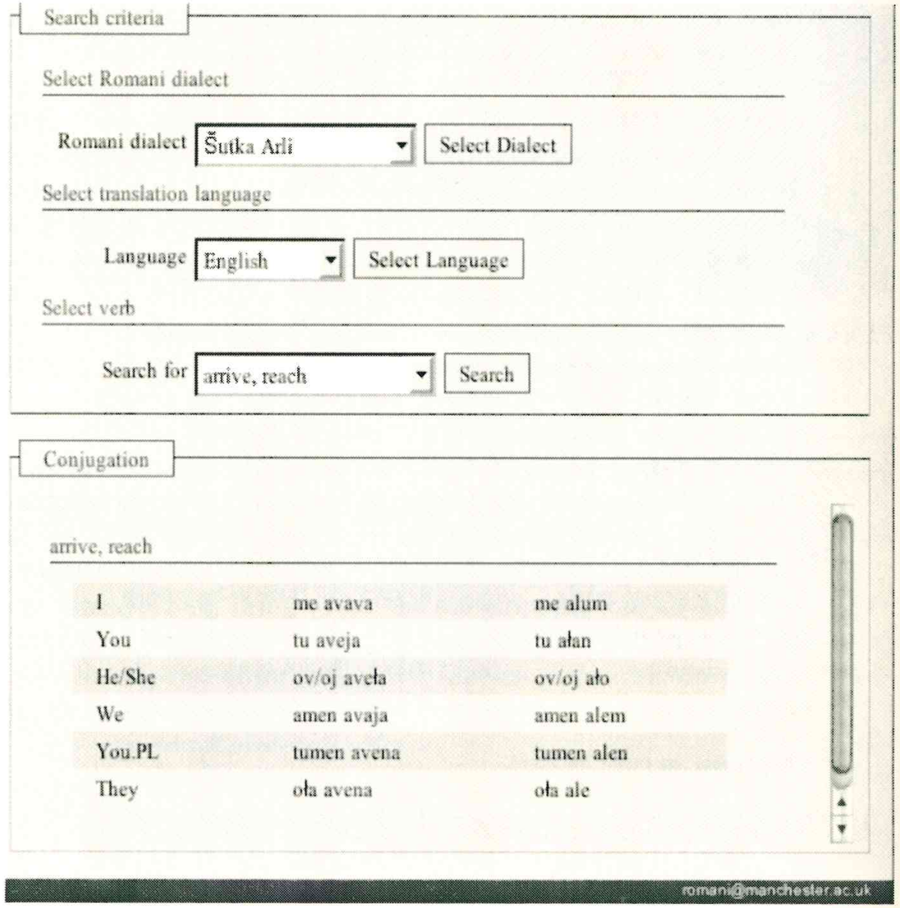
Search results for 'boy'

boy	◁) čhavo
This is a small boy.	◁) akava ine tikno čhavo
Only one boy is waiting quietly.	◁) samo jekh čhavo udžarela mirno
Give a piece of cake to this one boy!	◁) de le jekh kotor torta kale čhave
One boy's shirt was torn.	◁) jekhe čhaveskere majca sine pharavdi
A small boy came to the wedding as well.	◁) jekh tikno čhavo alo ko bijav isto
The boy's father had such a big beard.	◁) e čhaveskoro dat isi le doborom bari brada

Figure 7. Output menu of query function 'Phrase search'

Note that in the output, the corresponding audio file for each transcribed word or phrase can be heard as well, by clicking on the audio icon.

The wordlist query operates on the basis of a direct retrieval of a word or paradigm. A list of some 250 everyday words is included in the questionnaire for the sake of lexical and phonological comparison among the dialects. A list of a chosen set of over 50 verbs with complete conjugations documents all relevant verb inflection classes in the language. Figure 8 shows the query output for the selection of the verb 'arrive': the user can view the complete present-tense and past-tense conjugations, those namely, that cannot be easily predicted since they involve inflection and not an analytical marker. Verb inflections are thus typical 'form-to-form' queries.



Search criteria

Select Romani dialect

Romani dialect: Šutka Arli

Select translation language

Language: English

Select verb

Search for: arrive, reach

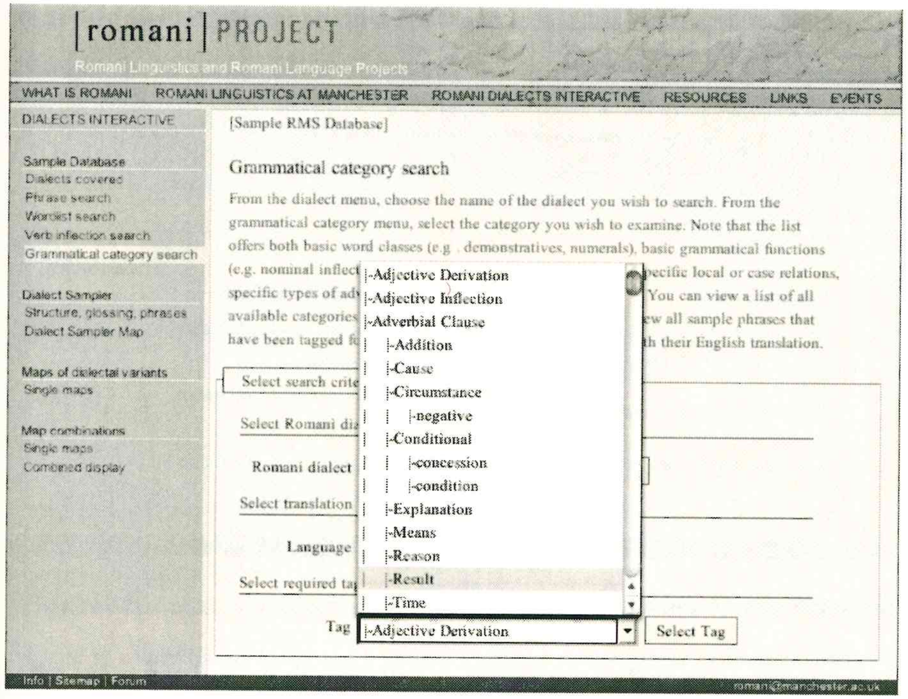
Conjugation

arrive, reach

I	me avava	me alum
You	tu aveja	tu ałan
He/She	ov/oj aveła	ov/oj ało
We	amen avaja	amen alem
You.PL	tumen avena	tumen alen
They	oła avena	oła ale

Figure 8. Query 'verb inflection' output menu

On the other hand, 'function-to-form' queries exploit the tagging system that is applied to phrases. The user is able to open a window and within it select a particular tag, representing a grammatical-semantic or category function (Figure 9). In the output, all phrases are shown which contain the relevant tag (Figure 10).



[Sample RMS Database]

Grammatical category search

From the dialect menu, choose the name of the dialect you wish to search. From the grammatical category menu, select the category you wish to examine. Note that the list offers both basic word classes (e.g . demonstratives, numerals), basic grammatical functions (e.g. nominal inflect
specific types of adv
available categories
have been tagged fo

pecific local or case relations,
You can view a list of all
ew all sample phrases that
th their English translation.

Select search crite

Select Romani dia

Romani dialect

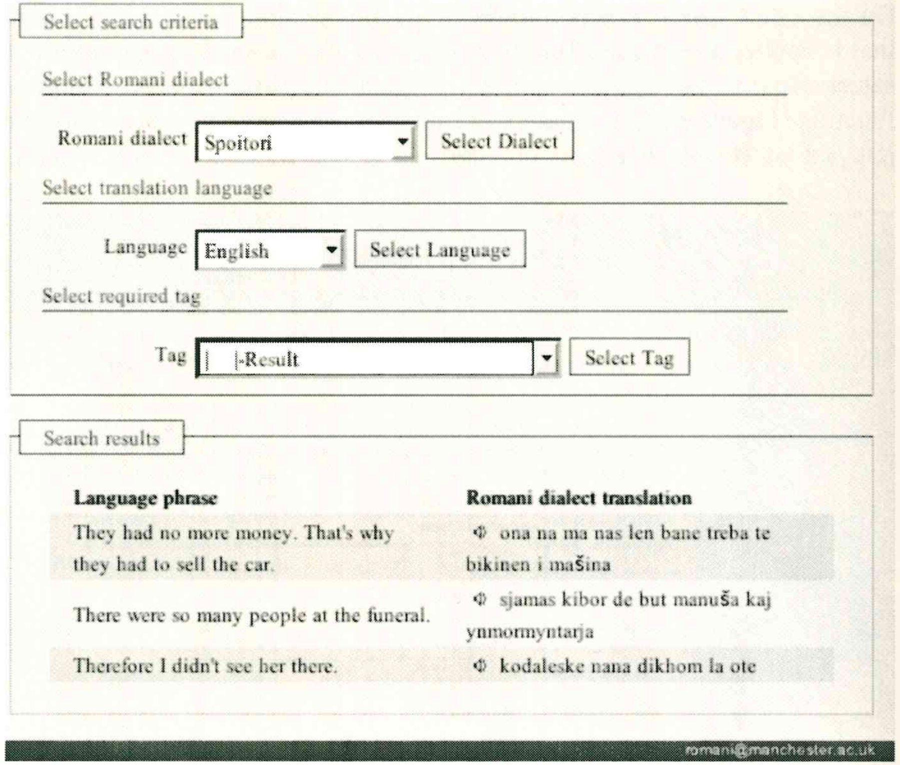
Select translation

Language

Select required ta

Tag: |-Adjective Derivation

Figure 9. Web-based Sample Database query function 'grammatical category search'



Select search criteria

Select Romani dialect

Romani dialect: Spoitori ▾ [Select Dialect]

Select translation language

Language: English ▾ [Select Language]

Select required tag

Tag: | |-Result ▾ [Select Tag]

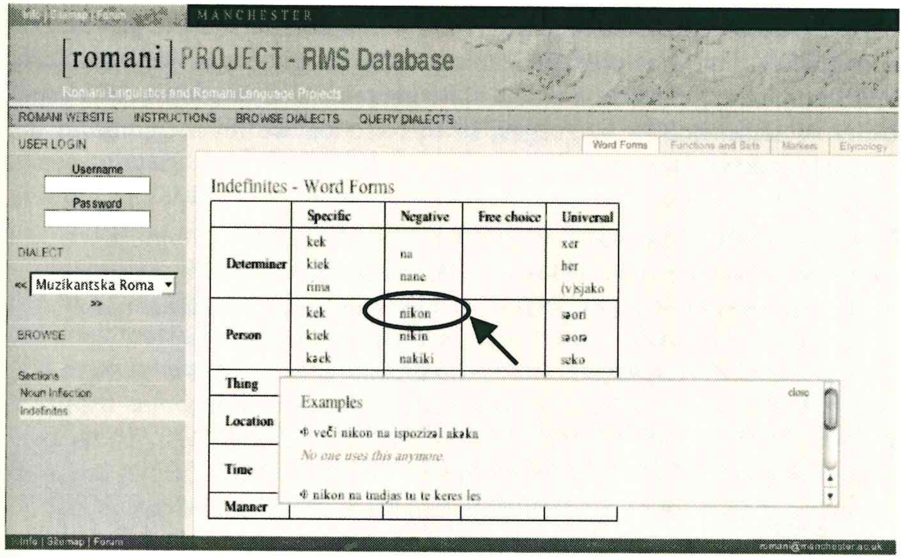
Search results

Language phrase	Romani dialect translation
They had no more money. That's why they had to sell the car.	◂) ona na ma nas len bane treba te bikinen i mašina
There were so many people at the funeral.	◂) sjamas kibor de but manuša kaj ynmormyntarja
Therefore I didn't see her there.	◂) kodaleske nana dikhom la ote

Figure 10. Output 'grammatical category search'

The queries illustrated above (Figure 6–10) constitute an innovation compared to the functions covered by the older database sketch in FileMakerPro 6. From the technical side, they represent, in fact, an entirely new database, a custom-designed application with a web-interface, which replaces the old sketch in FileMakerPro 6 (see below). While the Sample Database – the first part of the new RMS to be available online – is still limited to direct linkage to phrases via the tagging structure, the new RMS 'proper' combines the strength of the analytical RMS database with the functionality of the new application, in that it integrates complete datasets (supplied to it through spreadsheets of transcriptions and corresponding sets of sound files) into the tables that hold data and metadata on grammatical structure. This new application is currently, at the time of writing, under development, and is planned to be freely accessible via the project's website from 2008.

Figure 11 illustrates the presentation of a typical function-to-form section, here the table of indefinite pronouns. Note that by clicking on the respective field within the table, the user is able to retrieve relevant phrases via the tagging system from the questionnaires, in both transcription and sound. The same procedure is followed in order to input data into the tables, for each record. The database is thus enriched by an interactive dimension which allows, for each and every item of data, sentential exemplification, in transcription and sound – something that is impossible to deliver in a conventional written grammatical description.



Dialect: << Muzikantska Roma ▾ >>

Indefinites - Word Forms

	Specific	Negative	Free choice	Universal
Determiner	kek kiek rima	na nane		xer her (v)sjako
Person	kek kiek kaek	nikon nikin nakiki		saori saora seko
Thing				
Location				
Time				
Manner				

Examples

◂) veči nikon na ispozizal akaka

No one uses this anymore.

◂) nikon na tradjas tu te keres les

Figure 11. Web-based Database layout 'Indefinite forms', with exemplification for Person-Negative form (*nikon* 'nobody')

4. Management and organisation

From the above it has become clear that RMS is not just a database, but also a strategy for data collection, processing, and evaluation: It inspires, and is dependent on, a certain method of data collection and archiving, and in its design it subscribes to certain notions prevalent in functionally-oriented typology in respect of categorisation and structural representation of semantic functions, and to certain assumptions about the diachronic development of Romani. Following the database outline it is possible to compose basic grammatical descriptions of the language that are informed by both the functional-typological and the particular diachronic assumptions about

Romani (see e.g. Matras 2004b; Tenser 2005; Chileva 2005; Chashchikhina 2006). More than just a tool to store data, RMS is thus an integrated approach to language documentation and evaluation. Despite its anchoring in certain assumptions about language function and the development of Romani, however, it leaves ample scope for analysts to retrieve data and evaluate them in entirely different directions. It is thus an open resource, one that is theoretically informed but not theoretically prejudiced. Although RMS was constructed as a tool specifically for the investigation of Romani, the procedure behind the management of the resource and the project that supports it is, in principle, applicable to other languages as well. As such, RMS may be regarded as a model for comprehensive documentation especially of lesser-known languages. In this section we review some of the general aspects of the project.

1 Research into dialectal variation
2 Postulation of historical developments/ background analysis
3 Drafting of form, form-to-form, and form-to-function data presentation layout
4 Integration of typological description grids, drafting of function-to-form layouts
5 Creation of a questionnaire to elicit all aspects of structural variation
6 Tagging of questionnaire data with reference to database fields
7 Fieldwork using the questionnaire: training of fieldworkers, audio recording and transcription of interviews, archiving
8 Database upgrade: Custom-made application with web interface, re-design of data tables, integration of questionnaire data with transcription and sound
9 Release of Sample Database and gradual upgrade (adding of samples)
10 Release of complete database model
11 Development of elaborate query structure
12 Replication of model for other languages

Figure 12. Summary of RMS implementation and management strategies, by stage

The steps outlined in Figure 12 represent successive (though sometimes also parallel and intertwined) stages in the project's development. As discussed in the previous section, the preliminary sketch for RMS consisted of an inventory, by grammatical category, of variants, derived from existing literature. To this, information deemed essential for a language description was added – inspired and informed by typological works. Note that most grammatical descriptions of Romani had not, by that stage, been typologically oriented, and few contained any information at all about syntactic typology. Comments and data on syntax in the relevant literature were quite often limited to loose exemplification, rather than systematic remarks. Thus, complementation might be illustrated with one or two examples, but those would not enable to retrieve many insights into the continuum of modality vs. epistemic complementation, for instance. Based on the survey of morphological variants, combined with a typological survey, a preliminary design was produced, outlining the information that was of interest.

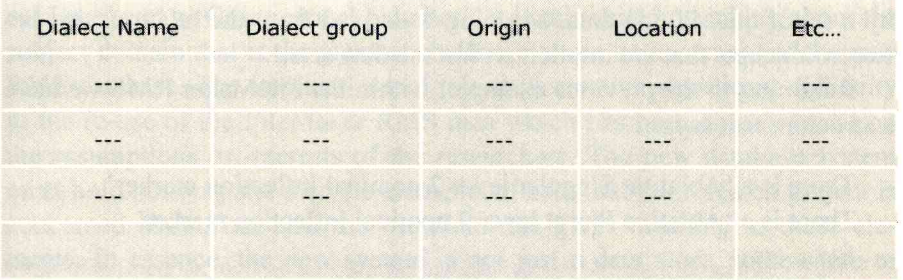
The approach at this stage was not a technical one, and was completely uninformed by any technical approach to database design. Rather, it was based on a purely *linguistic* appreciation of relations between values as representing linguistic functions and paradigm values, with no distinction between primary data, derived data, and meta-data commenting. The availability of FileMakerPro as an application that allowed amateur plotting and easy retrieval of data, created a temptation to focus the project's resources on recruiting linguistic, rather than technical skills. In effect, the resulting file in FileMakerPro was nothing but a single, huge table, with over 5000 columns representing content-defined data fields, interacting with a mere 100 or so rows representing individual dialect records. In hindsight, a more informed approach would have quite possibly enabled a quicker production of a proper application with a relational structure, able to store complex data and allowing the necessary flexibility in designing a query structure. An impeding factor, however, is the structure of the grant scheme and the need to complete phases within a relatively short funding period. Prior to the successive development of questionnaires and the procedures to tag phrases, the full requirements and opportunities of the database would not have been envisaged; and these in turn could only emerge once a database sketch was in place, storing a preliminary set of data and allowing cross-dialectal comparison.

Previous fieldwork on Romani, much like fieldwork on other languages, or on cross-linguistic samples for typological purposes, relied on just a limited set of questions aiming to elicit a modest set of variables. The inconveniences of a comprehensive questionnaire aiming to document a variety of structures to allow a complete descriptive sketch of a dialect are obvious: the time constraints limits access to informants, the amount of material takes time to process, check, archive, and evaluate. The RMS questionnaire is one of few enterprises known to us that aimed at a comprehensive description of dialectal varieties of a language. The work of administering the questionnaire, archiving and processing the data was only possible through the creation of an entire network (in the case of Romani, an international network was required), within which several dozen individuals carried out a series of specialised tasks, from interviewing in particular languages, to transcribing particular dialects, to archiving the material (checking transcriptions, digitising and editing sound files) and inputting the data into the actual database. The network allowed a kind of production-line management of some of the tasks. Thus interviewers are able to pass on their recordings to an archive manager, who delegates various tasks to transcribers, sound technicians, and later to analysts for input; not only are these different individuals, but quite often they work at different institutions and reside in different countries.

The need to upgrade the database to a custom-made application arose when it became clear that the available dataset could only support a very limited query structure, which could not be integrated with other applications or extended to cover new functions. The major problem encountered in this phase was the need to re-define categorisations and create relations among data sets in different tables. This lengthy process, still ongoing at the time of writing, involves a productive re-assessment of the possible relations among instances of data which are not self-evident from the purely linguistic-paradigmatic perspective. The very first significant upgrade was the creation of a database that could accommodate the actual transcriptions and sounds, with their tags, thus allowing the direct query structure described above. The fact that relations between phrases, tags, and sound samples had already been set in advance allowed a rather quick design and early sharing of the so-called Sample Database with a wide audience on the web. Following from this is the gradual convesion of the original sketch into a relational database (see below), the import of data already stored in the FileMaker format, and the development of query structures. A middle-term aim is then to view RMS as a model for potential applications documenting other languages, and to pilot its adaptation to another group of closely related languages.

5. The database structure: technical aspects

The original RMS database, built using FileMaker version 5, cannot be considered a relational database. FileMaker version 5 encourages the development of single table databases; in a sense, a simple spreadsheet with each row being a 'record' and each column holding a certain piece of information for each record.



Dialect Name	Dialect group	Origin	Location	Etc...
---	---	---	---	---
---	---	---	---	---
---	---	---	---	---

Figure 13. FileMaker database structure

This design resulted in the original RMS database containing in excess of 5,000 columns, each holding a discrete piece of information about the dialect. However, while this method is capable of holding any arbitrary data, it is not capable of holding any information about the data. In essence the data, in itself, is meaningless. Any meaning is only derivable from outside the system; meanings imposed upon it by the user.

In fact, the data does have inherent meaning, it is simply that the FileMaker structure cannot represent this. For example the FileMaker database has several columns containing data concerning 'Layer 2 nominal inflection' markers. The data held in the column is the actual marker for the specific dialect, but which inflection it is (for which case/number combination) is not identifiable from the data. For example, the FileMaker database has two columns for 'Ablative' nominal inflection markers; one for 'Plural' and one for 'Singular'. There is nothing within the data that indicates that either of these columns has anything to do with 'Ablative' or to do with 'Singular' or 'Plural' (or in fact that it has anything to do with nominal inflection). This can only be discerned by the user reading the arbitrary label given as the column name. Further, there is nothing inherent within the FileMaker database that indicates that the 'Ablative singular' marker and the 'Ablative plural' marker have anything in common (ie, that they are both 'Ablative' markers).

The new database development faces the challenge to correct this. A relational database attempts to represent data by its relationships to other data, thus building a network of 'links' between subject domain concepts, having the data as a quantification of these relationships for a specific 'record'. In fact the relational model does away with the 'traditional' concept of 'Records' (a single, linear, collection of data referring to a single subject concept) and, instead, breaks the subject of the database into its component concepts. Each concept, or possible variation of that concept, is represented by a set of quantifying data. Meaning is derived from the relationships between concepts that are implicit within the data sets.

Returning to the previous example; within the FileMaker RMS we have a structure that says…

- There is a 'Ablative Singular layer 2 nominal inflection marker'
- There is a 'Ablative Plural layer 2 nominal inflection marker'
- Etc…etc..

In the relational model we say…

- There are *'Samples'*[2]
- There are *'Nominal Inflection Markers'*
- There are *'Grammatical Cases'*
- There are *'Grammatical Numbers'*
- *'Samples'* have *'Nominal Inflection Markers'*
- *'Nominal Inflection markers'* have a *'Grammatical Case'*
- *'Nominal Inflection markers'* have a *'Grammatical Number'*
- *'Nominal Inflection markers'* have a shape

'Samples', 'Nominal Inflection Marker', 'Grammatical Case' and 'Grammatical Number' are component concepts that allow us to build a representation of the subject domain, in this case dialects of Romani. Each of these concepts could have a number of attributes that allow the quantification of each instance of the concept. For example 'Ablative' is an instance of the concept 'Grammatical Case', or put another way one possible instance of 'Grammatical Case' has a 'name' of 'Ablative'. In the same way, the concept of 'Grammatical Number' has two instances, one with the 'name' of 'Singular' the other with the 'name' of 'Plural'. So, conceptually, as a 'Car' has an 'Engine' and a 'Gear Box', a 'Nominal Inflection' has a 'Grammatical Case' and a 'Grammatical Number'. It is this shift in thinking about the subject domain that is the greatest challenge to the development of the new RMS database. The implications of this shift create a significant difficulty in the re-use of the FileMaker RMS data which has been chosen based on the assumptions or interests of the researchers. The new database system must hold not only the discrete elements of data that the research project is concerned about, but also the metadata that gives meaning to those elements. In essence, the new system is not just a data store, somewhere to hold the results of analysis, but it is a model of the real world.

It is often tempting to think of a relational database in terms of 'records' and 'fields'. For many data sets this can be a useful conceptualisation of the data, however, in designing a database it lends itself to over simplification along with the combining of concepts and if care is not taken can result in a lack of adequate normalisation. It also implies formal structure and order where there is none. For example, in the RMS database a 'Record' could be considered to comprise all the information held about a dialect. However, this data will span many component concepts and multiple instances of a concept may relate to the same dialect. If one considers these concepts as represented as tables and the attributes as columns within these tables, then the 'Record' for a dialect will comprise many rows from many tables (and quite possibly multiple rows from the same table), thus the connotations of the concept 'Record' looses validity. In the strictest sense a relational database is constructed of 'Relations', 'Tuples' and 'Attributes'; where a 'Tuple' is a collection of 'Attributes' and a 'Relation' is a set of 'Tuples' that all have the same 'Attributes', no structure nor order is implied. To ease the interaction with the dataset most relational database client interfaces utilise the more familiar representation of the data as 'Tables' with named 'Columns', each 'Tuple' being presented as a row in the table. These concepts are derived from SQL (Structured Query Language), almost exclusively used as the interface to relational database management systems.

One of the more significant changes to the structure of the information for the new RMS database is the way in which a Dialect is defined. Within the new system each interview with an individual constitutes a 'Sample' of the dialect. It is from this sample that data is extracted and entered into the database system. In this way each Dialect can have more than one Sample and thus, more than one set of data defining it. This allows for the interesting possibility of analysing the differences within Dialects as well as between Dialects, measuring similarities and differences between Samples, and ultimately being able to represent the transition between dialects in a more realistic, gradual morphing rather than a set of discrete boundaries.

[2] In earlier phases, the unit explored was considered a 'Dialect' of the language. In later discussions, especially in connection with the technical compilation of the data, it was decided that there was no obvious procedure through which to distinguish 'dialects' from individual 'samples', each of which represents a speaker. Several speakers may be grouped together on the basis of their origin, or residence in, the same location, or on the basis of (any sets of) similarities among them. The entity 'dialect' is thus a secondary classification of samples. It was therefore decided that the database should operate on the basis of assigning data to individual 'samples', each representing a speaker (or a published source, in the case of secondary source compilation).

In order to achieve a platform independence for the new database system it was decided to use web based technologies for the user interface. The user is able to access the data from any computer platform that has a standards compliant web browser and, obviously, is connected to the internet. With such a detailed system, with complex functional requirements this presents a further challenge to the development process. This challenge has been met by the implementation of a multi-tiered MVC (model-view-control) design for the application. Each tier is functionally independent of each other with inter-tier communication achieved through a specified and consistent interface.

Tier 1 is the User Interface. This is the actual web page that the user sees and interacts with. This is built with standard web technologies; HTML for layout definition and Javascript to control the functionality of the individual layouts. Tier 1 follows a loose Model-View-Control architecture. 'Model' being the data that is presented on the layout. 'View' being the HTML code. 'Control' being the Javascript code that manipulates the 'Model', presenting the data on the layout, registers and responds to user activity and communicates with Tier 2.

Tier 2 is the application code that runs on the server. This code performs the tasks requested of it by Tier 1 (eg. acquire data, update data, user login etc...). Tier 2 exposes its functionality to Tier 1 though a simple API (Application Programming Interface). This allows the code running within each user's browser to perform the tasks requested of it by the user. Tier 2 follows a standard Model-View-Control architecture. 'Model' being a set of object classes that represent the key concepts within the database model (each class roughly equating to a 'Table' within the database) and give functionality to the instances of those concepts. The object classes that make up the 'Model' act as a wrapper around the underlying database which comprises Tier 3. 'View' being the application code that build the visual layouts that are presented within the user's browser. 'Control' being the application code that performs the 'business' logic; manipulates the 'Model' in order to select, insert or modify data, decides which 'View' to trigger the building of and performs other tasks such as authentication and authorisation. This separation of Model, View and Control allows for easy modification of the data structures, the visual appearance or the underlying business logic having any effect on the others.

Tier 3 is the actual database itself. This is a full Relational Database Management System (RDBMS). This tier holds and manages all the data and the structures that define the data. Tier 2 has access to the data, and the functionality of the RDBMS through the SQL (= Structured Query Language) based interface that the RDBMS exposes.

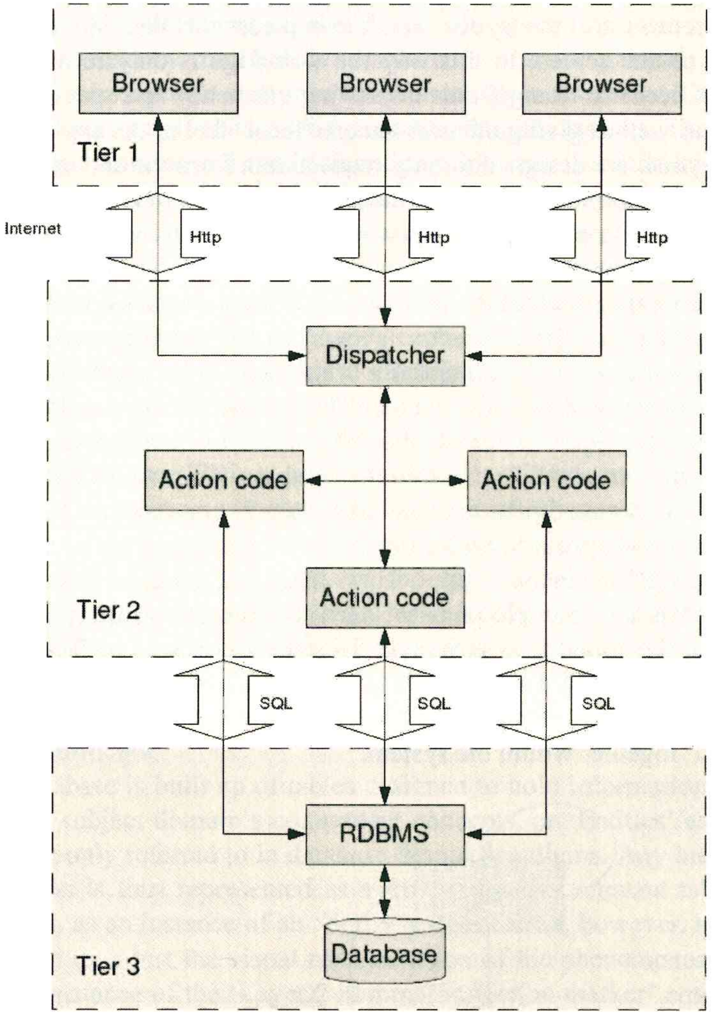


Figure 14. The three-tier structure of the relational RMS database

For data to be presented on the user's screen, the user's browser first sends a request across the internet to the server asking for the specific layout. The server receives this request, processes it, building and returning the layout to the browser. The layout comes in two parts: the formatting code that defines how the page looks on the screen and some control code that defines the functionality that the layout has. At this point the layout has no 'data' on it. Once the layout has loaded, it contacts the server, requesting all the data to be displayed. The server gathers the data via a request to the database and returns it to the layout, which then places that data into the relevant locations on the screen. In this way the web page is only refreshed when the layout needs to change, thus improving efficiency and speed of the application as well as giving the user a more 'local' feel to the application.

The layouts are designed to only request data from the server when they need it. This is demonstrable with the exemplification of data on the layout. With a click on a piece of data a window appears showing example sentences that demonstrate the use of the specific data. These examples are 'loaded' from the server at the point at which the user clicks on the data.

In a similar way the data entry layouts request from the server a list of suggested values to be presented as a 'value list' to the user when they try to enter a piece of data for a dialect. This 'value list' is generated on the server from a unique list of all the values that are entered for that 'data point' for any sample/dialect within the database. This makes the value lists dynamic and always up to date (as new data is entered into the database these values will appear in their respective value lists).

The new RMS database application has, conceptually, three levels of data. There is the 'Sample phrases' derived from the transcriptions of the interviews, the sound files generated from the recordings of the interviews and the 'dialect definition data' which is extracted from the transcribed sentences (synonymous with the FileMaker RMS data). All this data needs to be 'linked' together within the system.

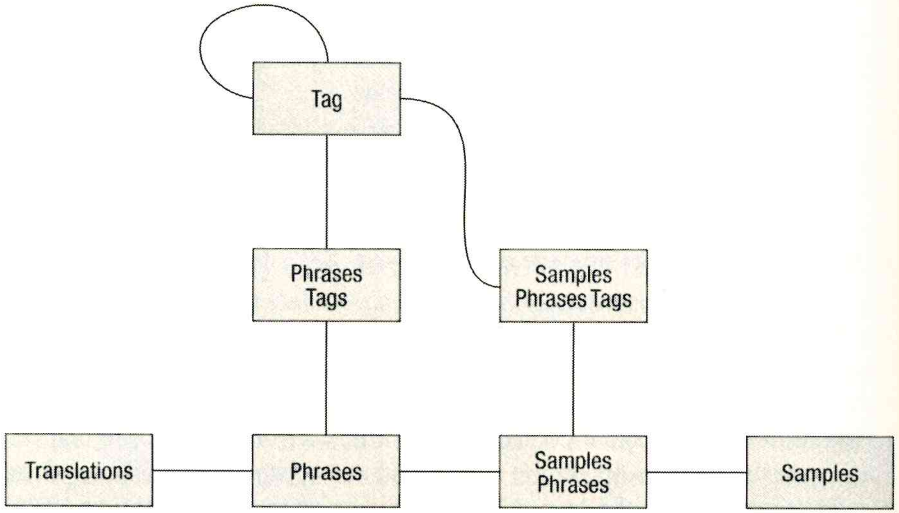


Figure 15. Overview structure of the 'Sample phrases' data

The Sound files are stored on the server file system and are referenced within the system by the Dialect/Sample code and the phrase's reference number, which together comprise the sound file's filename. This allows sound files and Sample Phrases to be combined on the layouts. The 'linking' of the Sample Phrases and the Dialect definition data is a little tricky. Each piece of data refers to some phenomenon identified within the sample's recorded phrases. These phrases are used as the source for data entry as well as exemplification of data once entered. Thus the system must identify which phrases are likely to present each required phenomenon. To achieve this the system holds a set of 'Tags', each referring to a particular phenomenon. Each phrase is 'Tagged' to identify the likelihood of it containing examples of each of the relevant phenomena. Given a specific phenomenon (tag) the system can then present all of the phrases, or all of any one sample's phrases, that are likely to present examples of that phenomenon.

However, the system also needs to know which phenomenon the user is looking for. This can be achieved in two ways; by the user selecting the phenomenon from a list (i.e. selecting the 'tag') or by identifying the phenomenon via the data point that the user is looking at. The first option is seen within the 'Sample Database' system that is currently available from the project website and constitutes phase 1 of the development project. Phase 2 development, found within the RMS database application, presents the second principle.

The database is built up of tables designed to hold information that represents the subject domain's component concepts, or 'Entities' as they are more commonly referred to in database design paradigms. Any one specific phenomenon is thus represented as a row within the relevant table (or in other terms, as an instance of an 'Entity'). These rows, however, hold more information than just the visual representation of the phenomenon. For example, an instance of the 'Layer 2 nominal inflection marker' entity is represented by a specific string of characters, its form. However there is more information needed in order to get meaning from this representation; the 'Sample' it is from, the 'Grammatical Case' that the marker represents, the 'Grammatical Number' that it represents. So the row within the table must hold information on 'Sample', 'Grammatical Case' and 'Grammatical Number' and the form (or shape) of the inflection marker. This 'extra' information of Case and Number could be considered metadata (it describes the data, giving meaning to it, rather than being the data itself). It is also known data, there is a small, fixed list of alternatives that can be used, and it is this information that defines what the user is looking for in the data itself.

For example, a user may look for an 'Ablative', 'Plural' marker. Using all combinations of this known metadata 'proto entities' can be constructed. In this case there would a 'Layer 2 nominal inflection marker' proto entity for each possible combination of case and number and stored in a table in the database that mirrors the table that holds the actual data. This table has two differences to the 'data table': There is no reference to 'Sample' nor any form data (these would be meaningless in the context of the 'proto entity'). There is a reference to the 'Tag' used to indicate the specific phenomenon within the sample phrases. In this way, given specific instance data of a 'Layer 2 nominal inflection marker', or given that the user is requested to enter the form of a specific 'Layer 2 nominal inflection marker', the system will know both Case and Number. This metadata is then used to lookup the Tag using the table that holds the 'proto entities' and can then present all the sample's phrases that are 'linked' to the returned 'Tags'.

Querying the new RMS database system will be handled in three different ways within the application. The most basic way of querying will be similar to the way queries can be made on the current RMS database system. The user, presented with an empty set of layouts can enter certain criteria into the individual 'cells' and then request the application find all the samples that match those entered criteria. This interface will look, fundamentally, like the normal data input interface. Once the matching samples have been returned the interface will allow the user to view any layout for each of these samples. This is a basic 'filter' query. The user, given the correct permissions, will be able to download the results in various formats.

In addition to the simple 'filter' query, the user will be able to select to have certain values plotted on a map of Europe, giving a graphical representation of the dispersal of the particular phenomena. The intention is for this to be ultimately flexible. A more sophisticated mechanism for querying the data, based on the abilities of the SQL database that underpins the new RMS development, will also be implemented. This will involve a complex interface for building unique queries that can analyse the data held within the RMS database rather than just present data per sample. To accompany this interface will be several 'standard' queries that can be used to compare samples with each other within or between dialects.

What has been described so far is a rather simplistic overview of the main technical features of the database application under development, but it does allude to a very complex system with an intricate lattice of data with differing requirements. When one considers the development of such a system, there is a great need to consider many other implications that may not be immediately obvious.

The platform on which the application is developed is rather critical. It is all too often that such projects can take what is considered an 'easy option'. However, easy options are rarely the best option in hindsight. There are issues concerning proprietary lock-in that are often not considered. The original RMS database suffered from this, being built upon FileMaker. FileMaker is a proprietary database system that requires licensing of the product for both client and server use. In this way, anyone who wanted access to the data would need to purchase a license. Although licenses are often perpetual in nature, they only relate to the version of the software that the license was purchased for. As time progresses the software vendor releases new versions and stops support for older versions. Often, for many reasons, new versions are not compatible with the older version. This is a situation that the FileMaker RMS database finds itself in. There are also vendor lock-in risks with bespoke developments, should they be developed on top of proprietary development environments. Like many establishments the development support team within our institution utilises Microsoft.NET development tools and servers for developing applications. Again this leads to vendor lock-in as the development tools and the servers that serve the application must be licensed, and the developments made using those tools can only be 'run' on the same vendors server software thus tying the application to that particular vendor, in this case Microsoft.

This project aimed to eradicate this issue, giving freedom to application and removing all licensing costs, by using only open source or free and standards based software. Due to constraints laid down by the institution the platform technologies that were used for the development were PHP for the server side programming, MySQL for the database engine and standards based web technologies for the client side development. These are not the best solutions for this application and require a degree of extra development work to overcome their shortcomings.

PHP has limited UNICODE support so care is needed when working with text within the application so not to mangle any UNICODE characters or produce specious results. PHP does provide a set of multi-byte functions that duplicate most of the core string functions and provides enough functional coverage for general multi-byte string manipulation. However, there are no specific multi-byte alternatives for non-string functions so care must be taken to ensure that multi-byte characters will not adversely affect the result of such function. For example, array sorting functions are not multibyte safe resulting in the sort order not being semantically correct when the array contains multi-byte characters. The array would be sorted by byte value rather than alphanumeric order, resulting in single byte characters being first (and in alphanumeric order) followed by all two-byte characters, then three-byte characters and then four-byte characters.

MySQL does now support the majority of the SQL standards, however there are certain features omitted, or have limitations (such as requiring 'root' privileges to implement), which can hinder advanced developments. The main concern for the RMS development is UNICODE compatibility. As much of the data is character based the logical choice is to use Regular Expressions for pattern matching in queries. However, MySQL's Regular Expression engine is not multi-byte safe. In most cases this has little impact as both Regular Expression and String To Match will be both UNICODE and any multi-byte characters would be considered as multiple characters in both and thus, relatively speaking, the pattern is maintained. However, this does cause an issue when using multi-byte characters within a Regular Expression character class, for example; [āēīōū]. In theory, this character class should match any one of the 5 long vowel characters presented. However, since each of these characters is multi-byte (in fact 2 bytes long each), the Regular Expression engine in MySQL seems to interpret this character class as containing 10 single byte characters and will try and match to any one of those 10 'characters'. Consequently, in MySQL the Regular Expression /d[āēī]d/ will not match the strings 'dād', 'dēd' or 'dīd' as it would be expected to. MySQL interprets the regular expression as trying to match a string that has 3 characters; the letter 'd' followed by any one of the 6 single byte 'characters' in the character class followed by a 'd' character. The string 'dād' is interpreted by MySQL as a 4 character string; 'd' followed by two single byte characters followed by another 'd'. Since the string has two characters between the 'd's and the Regular Expression requires only one MySQL's Regular Expression engine will not register a match. The necessary work around is to use grouping and alternation, so [āēī] becomes (ā|ē|ī) and MySQL, instead of trying to match with any one of the 6 single byte 'characters', is now trying to match with any one of the 3 'character pairs'.

There is also a considerable data storage requirement for this application. With each sample that is entered into the system there are potentially a few hundred megabytes of data to be stored. The bulk of this is the approximately 800 or so sound files per sample ranging between 4 and 200 kilobytes each. When one considers that the current FileMaker RMS consists of in excess of 100 dialects and the project is continuing its data collection, the application can easily require many gigabytes of storage space. This also needs to be backed up in case of system failure, and adequate backup facilities therefore need to be considered.

6. Conclusion: New prospects in descriptive linguistics

New technologies are by definition revolutionary: They allow us to do things that we were unable to do before, in relation to transfer and processing of information, but also in re-evaluating the meaning of information. RMS has had an institutional impact on Romani linguistics, by creating an international collaboration network needed to collect and process data on Romani varieties. It has also already inspired new analyses of Romani, and beyond – using the Romani sample as a basis for theoretical discussion (see Elšík and Matras 2006 on 'Markedness').

One indisputable accomplishment of RMS is its function as a resource of raw, yet catalogued data. Supplying the armchair user with both transcriptions and sound of hundreds of phrases from dozens of speakers, it brings fieldwork to the home. Moreover, it enables the user to control and verify every instance of analytical judgement and assessment taken by the input team, by retrieving the original exemplification. This sets a new standard in descriptive work in linguistics, which, once noted, is likely to prove difficult for linguists to fall behind. The availability of data in this way on the web also engages wider audiences of users, increasing the relevance of descriptive linguistics. The planned query structure involving dynamic generation of maps from within the database might be regarded as a new step in the understanding of dialectology and dialectological surveys, one which de-constructs, to a certain extent, the notion of dialect boundaries and 'genetic' groupings, and allows the user instead to consider a plethora of classification options with minimal effort. A key function here is carried by the planned query structure to measure distance among samples and sets of samples, as described above.

In the above we have not elaborated on our treatment of speaker metadata. However, a second phase of data collection began in January 2006, using a supplementary questionnaire on details of personal biography as well as community customs. So far, data were gathered in this way, along with data from the primary questionnaire, in communities in Ukraine, Moldova, Serbia, Croatia, Montenegro, Greece, Romania, Hungary, Italy and Poland. One of the tasks on the project's future agenda is to design opportunities to link grammatical data with ethnographic data and with biographical data of speakers, to explore the extent of variation and the existence of boundaries within communities as well as among them.

The latter functions are of great potential importance to educational and language policy in Romani. In the absence of either a standard language, or a central government with responsibility to safeguard language and promote language teaching throughout the Romani-speaking community, it is vital to gain a more thorough understanding about the prospects of cross-dialect communication and mutual intelligibility of the dialects, as well as to develop tools that would facilitate the transfer of text materials from one dialect to another. Combining RMS – the inventory of grammatical variants – with a lexical resource such as Romlex[3] – a lexical database of Romani dialects – could allow for the development of such a tool, which in turn would facilitate the pooling and sharing of linguistic resources for teaching and other purposes.

References

Bakker, Peter
1999 The Northern branch of Romani: mixed and non-mixed varieties. In *Die Sprache der Roma: Perspektiven der Romani-Forschung in Österreich im interdisziplinären und internationalen Kontext*, Dieter W. Halwachs and Florian Menz (eds.), 172–209. Klagenfurt: Drava.

Boretzky, Norbert
1999a Die Gliederung der Zentralen Dialekte und die Beziehungen zwischen Südlichen Zentralen Dialekten (Romungro) und Südbalkanischen Romani-Dialekten. In *Die Sprache der Roma: Perspektiven der Romani-Forschung in Österreich im interdisziplinären und internationalen Kontext*, Dieter W. Halwachs and Florian Menz (eds.), 210–276. Klagenfurt: Drava.
1999b *Die Verwandtschaftsbeziehungen zwischen den Südbalkanischen Romani-Dialekten: Mit einem Kartenanhang*. Frankfurt am Main: Peter Lang.

Boretzky, Norbert and Birgit Igla
2004 *Dialektatlas des Romani*. Wiesbaden: Harrassowitz.

Chashchikhina, Olga
2006 A grammatical sketch of Ukrainian (Servi) Romani. MA diss., University of Manchester.

Chileva, Veliyana
2005 The morphosyntax of Velingrad Yerli Romani. MA diss., University of Manchester.

Dixon, Robert M.W.
1995 Complement clauses and complementation strategies. In *Grammar and Meaning: Essays in Honour of Sir John Lyons*, Frank R. Palmer (ed.), 175–220. Cambridge: Cambridge University Press.

[3] http://romani.uni-graz.at/romlex/

Elšík, Viktor and Yaron Matras
2006 *Markedness and Language Change: The Romani Sample*. Berlin/New York: Mouton de Gruyter.

Frajzyngier, Zygmunt
1991 The de dicto domain in language. In *Approaches to Grammaticalisation*, Vol. 1, Elizabeth Closs Traugott and Bernd Heine (eds.), 219–251. Amsterdam: John Benjamins.

Frajzyngier, Zygmunt, and Robert Jasperson
1991 *That*-clauses and other complements. *Lingua* 83: 133–153.

Gilliat-Smith, Bernard J.
1915 A report on the Gypsy tribes of North East Bulgaria. *Journal of the Gypsy Lore Society*, new series, 9: 1–54, 65–109.

Givón, Talmy
1990 *Syntax: A Functional-Typological Introduction*. Vol. 2. Amsterdam: John Benjamins.

Matras, Yaron
1994 *Untersuchungen zu Grammatik und Diskurs des Romanes: Dialekt der Kelderaša/Lovara*. Wiesbaden: Harrassowitz.
1998 Utterance modifiers and universals of grammatical borrowing. *Linguistics* 36 (2): 281–331.
2002 *Romani: A linguistic introduction*. Cambridge: Cambridge University Press.
2004a Typology, dialectology and the structure of complementation in Romani. In *Dialectology meets typology*, Bernd Kortmann (ed.), 227–304. Berlin/New York: Mouton de Gruyter.
2004b Romacilikanes: The Romani dialect of Parakalamos. *Romani Studies* 14 (1): 59–109.
2005 The classification of Romani dialects: A geographic-historical perspective. In *General and Applied Romani Linguistics*, Dieter W. Halwachs, Barbara Schrammel and Gerd Ambrosch (eds.), 7–26. Munich: Lincom Europa.

Miklosich, Franz
1872–80 *Über die Mundarten und Wanderungen der Zigeuner Europas* X–XII. Vienna: Karl Gerold's Sohn.

Pott, August
1844–45 *Die Zigeuner in Europa und Asien: Ethnographisch-linguistische Untersuchung vornehmlich ihrer Herkunft und Sprache*. Halle: Heynemann.

Tenser, Anton
2005 *Lithuanian Romani*. Munich: Lincom Europa.

Turner, Ralph L.
1926 The position of Romani in Indo-Aryan. *Journal of the Gypsy Lore Society*, third series, 5: 145–189.

Wierzbicka, Anna
1988 *The Semantics of Grammar*. (Studies in Language Companion Series 18.) Amsterdam/Philadelphia: John Benjamins.