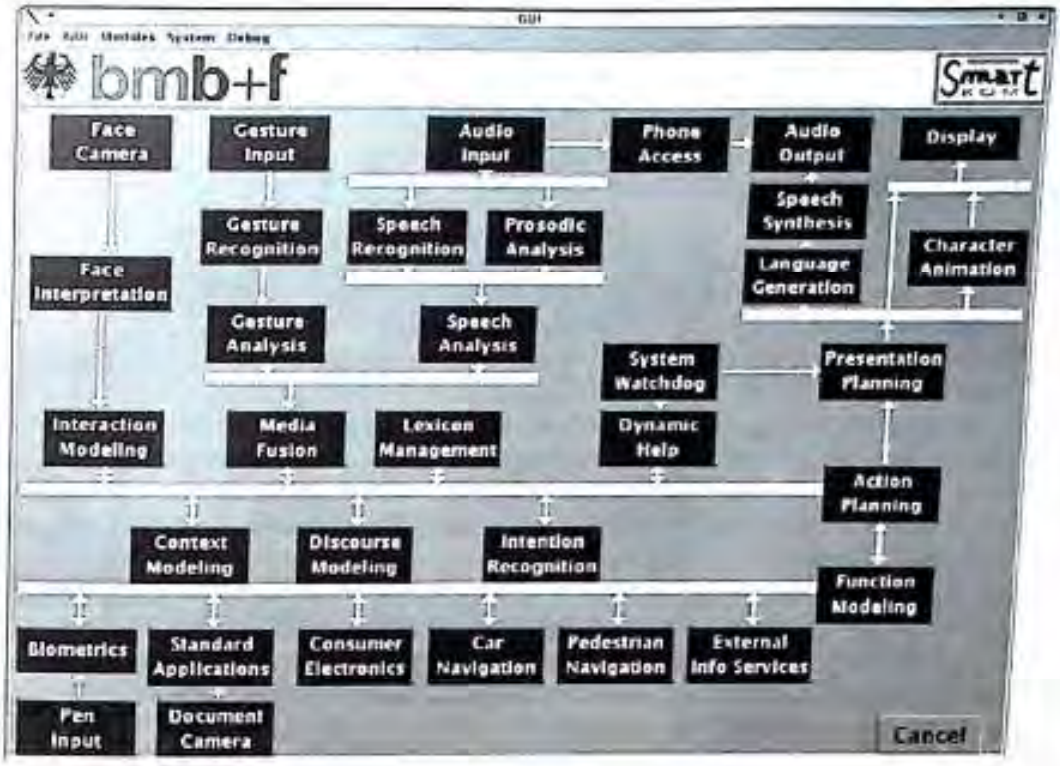


Fig. 3. Graphical user interface for testbed administration

3.3 M3L Support

The SMARTKOM system employs *Multimodal Markup Language* (M3L) as external data format for information exchange between functional components. M3L, an XML-based language, was developed within the SMARTKOM project to cover all data interfaces inside this complex multimodal dialogue system.

Parts of the language specification are generated automatically from an underlying conceptual taxonomy that provides the foundation for the representation of domain knowledge inside the dialogue system. The offline tool described in Gurevych et al. (2003) transforms an ontology written in OIL format (Fensel et al., 2001) into an M3L-compatible XML schema definition. Our tool simplifies updates of the M3L specification as the ontology evolves and ensures a consistent mapping from structural and semantic knowledge encoded in the ontology to directly interpretable communication interfaces.

From the point of view of component development, XML offers various techniques for the processing of transferred content structures. The standard document object model (DOM) API makes the M3L data available as a generic tree structure—the *document object model*—in terms of elements and attributes, whereas *simple API for XML* (SAX) provides an event-based API for XML parsers.

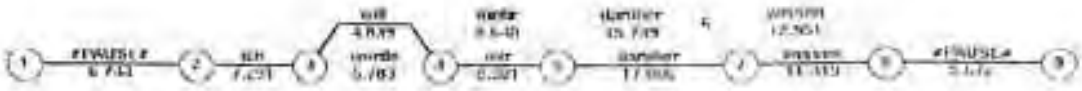


Fig. 4. Word hypothesis graph for "I would like to know more about this"

Another interesting option is to employ XSLT stylesheets to flexibly transform between the external XML format used for communication and a given internal markup language of the specific component. The use of XSLT makes it easier to adapt a component to interface modifications and simplifies its reuse in another dialogue system.

Instead of working on basic XML structures like elements and attributes, XML data binding can be used for a direct mapping between program internal data structures and application-specific M3L markup. In this approach, the M3L language specification is exploited to automatically generate a corresponding object model in a given programming language.

All the techniques decribed here have been applied in different implementations of the various functional components of the SMARTKOM system. In addition, a specific M3L API has been developed and included into the SMARTKOM testbed. This M3L API offers for all target platforms a light-weight programming interface to simplify the processing of XML structures within the implementation of a component. It incorporates a standard XML parser and uses the DOM interface to implement the M3L-specific API layer. Furthermore, the SMARTKOM testbed also provides a generic, XML-enabled toolbox and tailored XSLT stylesheets for the offline and online inspection of M3L data.

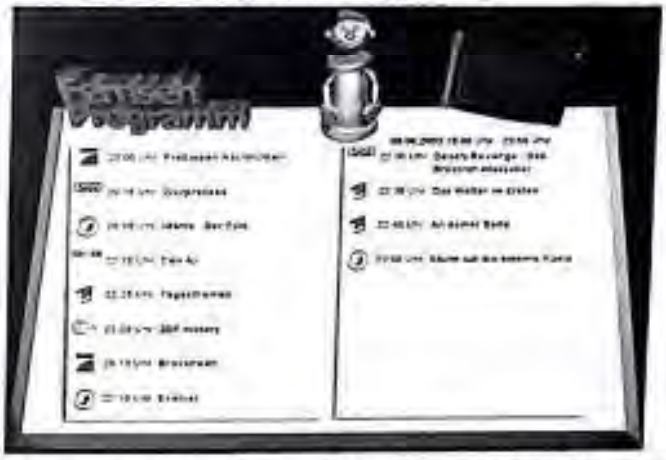
3.4 Additional Testbed Tools

A rich set of utilities can be used to manipulate log data easily, e.g., to select and identify relevant subsets from the complete data store resulting from a system run. Flexible replay of selected pool data provides a simple, yet elegant and powerful mechanism for the simulation of small or complex parts of the dialogue system in order to test and debug components during the development process. Using a simple tool, a system tester is also able to generate trace messages on the fly to inject annotations into the message log.

Another important development tool is a generic XML-enabled data viewer for the online and offline inspection of pool data. Figure 4 presents an example of a word hypothesis graph corresponding to the natural language part of a multimodal user input. Such a graphical visualization of speech recognition results during system execution aids the system integrator while testing the dialogue system.

The flexible data viewer toolbox can also be used in offline mode. A practical example is given in Fig. 5, which shows a comprehensible display format that documents an interaction with the SMARTKOM system.

02429 User comment: SmartKom displays the TV program for the evening
03699 #PAUSE# was kommt Sonntag Abend im Fernsehen #PAUSE#
03779 was kommt Sonntag Abend im Fernsehen (1.0)
was Sonntag Abend im Fernsehen.. (0.8)
03949 informationSearch InformationType=tvProgram weekday='sunday' daytime='evening' from='2003-06-08T18:00:00' to='2003-06-08T23:59:59'
04408 hier sehen Sie die gewünschten Sendungen



05182
05254 User comment: SmartKom provides information about the movie *Evolver*
06789 #PAUSE# ich will mehr darüber wissen #PAUSE#
06821 ich will mehr darüber wissen.. (1.0)
06989 informationSearch InformationType=broadcast at='2003-06-08T22:10:00' at='2003-06-08T23:50:00' avType='scienceFiction' title='Evolver' name='Kabel1'
07278 ich habe diese Informationen zu Evolver

07809
...

Fig. 5. Excerpt from an automatically generated dialogue protocol

This kind of dialogue protocol provides a compact summary of important processing results and is generated fully automatically from the message log. The individual entries consist of the specific message number followed by the summarized

content of selected data pools. Each item is prepended with the corresponding message number in order to enable more detailed inspection of relevant data if potential processing errors need to be resolved. Further offline tools include a standardized build and installation procedure for components as well as utilities for the preparation of software distributions and incremental updates during system integration.

4 System Integration Process

In addition to the software infrastructure, the practical organization of the project constitutes a key factor for the successful realization of an integrated multimodal dialogue system. Distributed teams and heterogeneous software environments within a large research project add complexity that must be resolved. In order to cope with system integration challenges, it is beneficial to clearly distinguish two major roles in the development process. A system integrator takes care of the overall software infrastructure and assembles executable components into an integrated application system, whereas a module developer should concentrate on a specific functional component.

For the SMARTKOM project, a dedicated system integration group composed of professional software engineers has been established to ensure that the resulting software is robust and maintainable, that an adequate architectural framework and testbed are provided to the participating researchers, that software engineering standards are respected, and that modules developed in different programming languages by a distributed team fit together properly. These aspects are too important to leave them to the individual scientists, who regard them as a side issue because they have to focus on their own research topics. An important achievement of the SMARTKOM project is the consistent integration of a very large number of components created by diverse groups of researchers from disparate disciplines.

Given the underlying architecture framework, the approach taken in SMARTKOM bears many similarities to component-based software engineering, where the notion of building an application by writing code has been replaced with building a system by assembling and integrating supplied software components (Bass et al., 2003). In contrast to conventional software development, where system integration is often the tail end of an implementation effort, component integration is the centrepiece of the approach.

Figure 6 provides a sketch of our development process, which is integration-centric as opposed to development-centric. In this kind of view, implementation has given way to integration as the focus of system construction.

After the design phase, which leads to a specification of the concrete component architecture and functional interfaces, the realization starts with the provision of a tailored testbed software package. The SMARTKOM testbed follows the *breadboard paradigm*, which provides an initial instantiation of the overall architecture using placeholders that represent the functional components (Bub and Schwinn, 1999; Klüter et al., 2000). During system integration, the dummy components are replaced

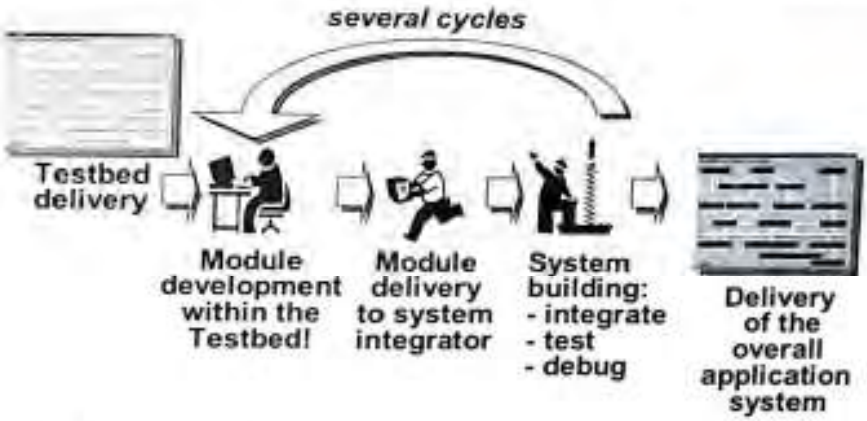


Fig. 6. Stages of system integration

successively by more substantial modules with increased functionality, i.e., components are like black boxes which can be plugged into a breadboard. Nonexistent modules can be simulated through the replay of appropriate test data. This technique enables rapid prototyping, given that even at an early stage an environment is available for testing and evaluating individual modules in the context of the entire application.

The one constant aspect of component-based software development is change. Constituent components are constantly evolving, leading to an iterative process where the different phases have to repeated several times until the multimodal dialogue application can be finalized. This includes also the customization of the software infrastructure, which is an integral element of a project life cycle.

Stepwise improvement and implementation of the design of architecture details and interfaces necessitate an intensive discussion process. It has to include all participants involved in the realization of system components in order to reach a common understanding of the intended system behaviour. The moderation and coordination of these activities constituted an important task of the system integration group in SMARTKOM.

The functional architecture of the multimodal dialogue system provides a basis for modularization into distinct software components. Significant parts of the initial system architecture were already predetermined by the project structure. Functional components are mapped to modules, realized as independent software processes, to obtain a specific software architecture. Figure 7 depicts the components and communication links inside the SMARTKOM system, which includes about 40 functional modules and 100 data pools for information exchange.

The specification of the content format for each data pool defines the common language that the dialogue system components use to interoperate. The careful design of information flow and accurate specification of content formats act as essential elements of our approach. We operationalize interface specifications in the form of an XML language. Our experience with M3L shows that the design of such an XML language for the external representation of complex data is not a simple task. All

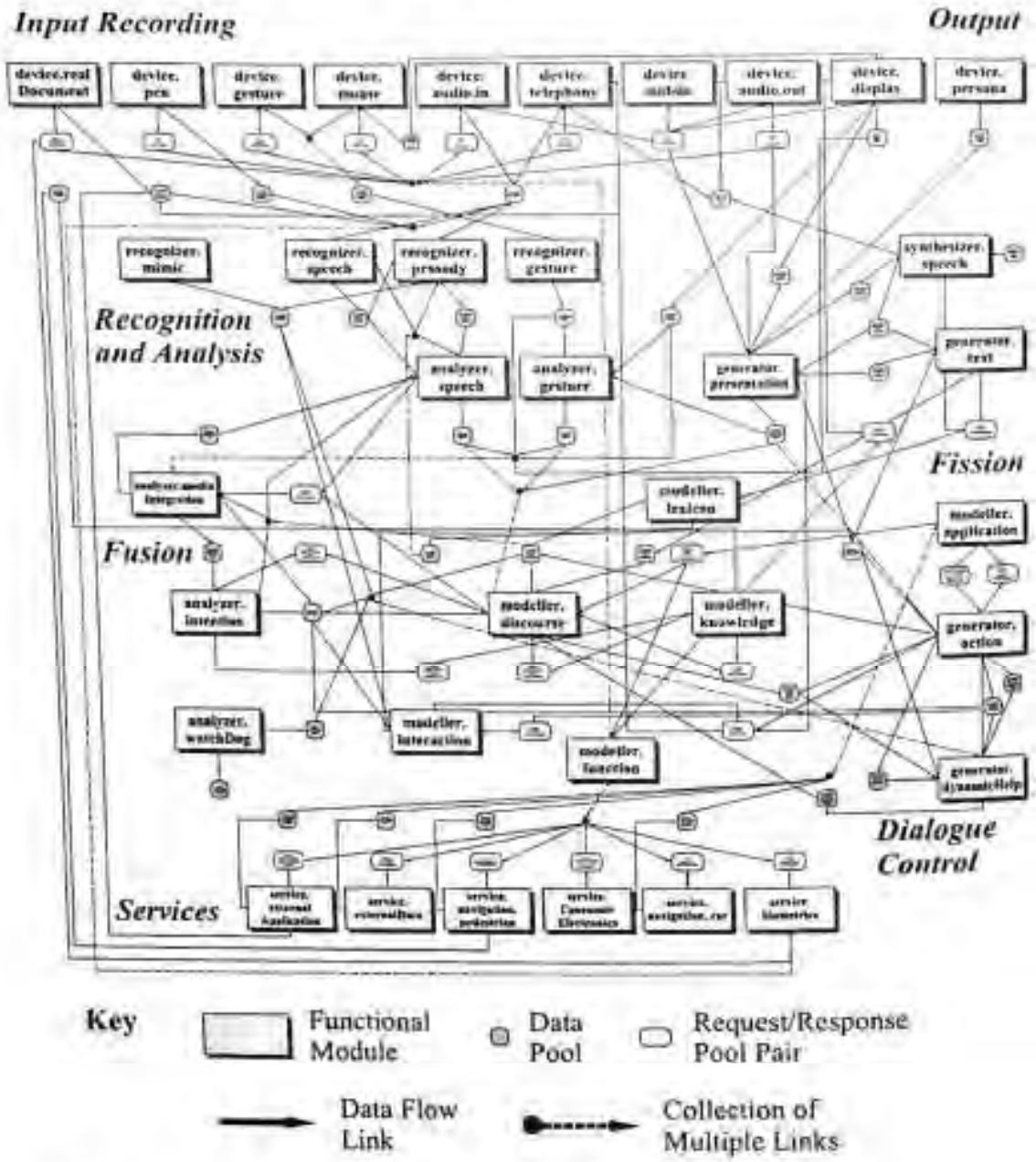


Fig. 7. Detailed software architecture of the SMARTKOM system

design decisions have to be made carefully. For example, it is better to minimize the use of attributes. They are limited to unstructured data and may occur at most once within a single element. Preferring elements over attributes better supports the evolution of a specification since the content model of an element can easily be redefined to be structured and the maximum number of occurrences can simply be increased to more than one. A further principle for a well-designed XML language

requires that the element structure reflects all details of the inherent structure of the represented data, i.e., textual content for an element should be restricted to well-defined elementary types. Another important guideline is to apply strict naming rules so that it becomes easier to grasp the intended meaning of specific XML structures.

The latest version of M3L consists of more than 1,000 different XML elements. In order to make the specification process manageable and to provide a thematic organization, the M3L language definition has been decomposed into about 40 schema specifications. This kind of modularization allowed to distribute responsibility for the different parts of the specification, assigning each XML schema to one selected researcher with specific expertise in that area.

5 Conclusion

This contribution has presented a discussion on system integration in building multimodal dialogue applications with an emphasis on the SMARTKOM integration platform. It has shown the analogy of our approach to component-based software engineering.

The capabilities of the multiplatform software framework are being enhanced on a continuing basis since the first design in Verbmobil, the predecessor of the SMARTKOM project. The SMARTKOM testbed provides a powerful environment and advanced new tools, like, for example, the added support for XML-based data representations. Multiplatform has been made available as open source software and is currently being used at several industrial and academic sites.

The paper also gave a look into the underlying system integration process and underlines the role of nontechnical factors for effective inter- and intrainstitutional collaboration.

Acknowledgments

First of all, we would like to thank our former colleagues in the system integration group. Heinz Kirchmann, Andreas Klüter, Hans-Jörg Kroner, Stefan Merten, Martin Neumann, and Rainer Peukert all contributed as well to the work described here.

Given the central position of our team within the project structure, there have been intensive contacts with nearly every project participant. We enjoyed this joint work and gratefully acknowledge the very successful collaboration in the SMARTKOM consortium.

References

L. Bass, P. Clements, and R. Kazman. *Software Architecture in Practice*. Addison-Wesley, Boston, MA, 2nd edn., 2003.

P.A. Bernstein. Middleware: A Model for Distributed System Services. *Communications of the ACM*, 39(2):86–98, 1989.

T. Bub and J. Schwinn. The Verbmobil Prototype System — A Software Engineering Perspective. *Natural Language Engineering*, 5(1):95–112, 1999.

N. Carriero and D. Gelernter. Linda in Context. *Communications of the ACM*, 32(4): 444–458, 1989.

W. Emmerich. Software Engineering and Middleware: A Roadmap. In: *Proc. Conf. on the Future of Software Engineering*, pp. 117–129, Limerick, Ireland, 2000. ACM.

L.D. Erman, F. Hayes-Roth, V.R. Lesser, and D.R. Reddy. The Hearsay-II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty. *ACM Computing Surveys*, 12(2):213–253, 1980.

D. Fensel, F. van Harmelen, I. Horrocks, D.L. McGuinness, and P.F. Patel-Schneider. OIL: An Ontology Infrastructure for the Semantic Web. *IEEE Intelligent Systems*, 16(2):38–45, 2001.

K. Geihs. Middleware Challenges Ahead. *IEEE Computer*, 34(6):24–31, 2001.

A. Geist, A. Beguelin, J. Dongorra, W. Jiang, R. Manchek, and V. Sunderman. *PVM: Parallel Virtual Machine. A User's Guide and Tutorial for Networked Parallel Computing*. MIT Press, Cambridge, MA, 1994.

I. Gurevych, S. Merten, and R. Porzel. Automatic Creation of Interface Specifications from Ontologies. In: H. Cunningham and J. Patrick (eds.), *Proc. HLT-NAACL 2003 Workshop on Software Engineering and Architecture of Language Technology Systems (SEALTS)*, pp. 59–66, Edmonton, Canada, 2003. Association for Computational Linguistics.

G. Herzog, H. Kirchmann, S. Merten, A. Ndiaye, P. Poller, and T. Becker. MULTIPLATFORM Testbed: An Integration Platform for Multimodal Dialog Systems. In: H. Cunningham and J. Patrick (eds.), *Proc. HLT-NAACL 2003 Workshop on Software Engineering and Architecture of Language Technology Systems (SEALTS)*, pp. 75–82, Edmonton, Canada, 2003. Association for Computational Linguistics.

G. Herzog, A. Ndiaye, S. Merten, H. Kirchmann, T. Becker, and P. Poller. Large-Scale Software Integration for Spoken Language and Multimodal Dialog Systems. *Natural Language Engineering*, 10(3–4):283–305, 2004. Special Issue on Software Architecture for Language Engineering.

J. Hopkins. Component Primer. *Communications of the ACM*, 43(10):27–30, 2000.

A. Klüter, A. Ndiaye, and H. Kirchmann. Verbmobil From a Software Engineering Point of View: System Design and Software Integration. In: W. Wahlster (ed.), *Verbmobil: Foundations of Speech-to-Speech Translation*, pp. 635–658, Berlin Heidelberg New York, 2000. Springer.

N. Reithinger, G. Herzog, and A. Ndiaye. Situated Multimodal Interaction in SmartKom. *Computers & Graphics*, 27(6):899–903, 2003.

W. Wahlster. SmartKom: Symmetric Multimodality in an Adaptive and Reusable Dialogue Shell. In: R. Krahl and D. Günther (eds.), *Proc. Human Computer Interaction Status Conference 2003*, pp. 47–62, Berlin, Germany, June 2003. DLR.

M.W. Wu and Y.D. Lin. Open Source Software Development: An Overview. *IEEE Computer*, 34(6):33–38, 2001.

SmartKom-English: From Robust Recognition to Felicitous Interaction

David Gelbart[1], John Bryant[1], Andreas Stolcke[1], Robert Porzel[2], Manja Baudis[2] and Nelson Morgan[1]

[1] International Computer Science Institute (ICSI), Berkeley, USA
texttt{gelbart,jbryant,stolcke,morgan}@icsi.berkeley.edu
[2] European Media Laboratory GmbH [EML], Heidelberg, Germany
{robert.porzel,manja.baudis}@eml-d.villa-bosch.de

Summary. This chapter describes the English-language SMARTKOM-Mobile system and related research. We explain the work required to support a second language in SMARTKOM and the design of the English speech recognizer. We then discuss research carried out on signal processing methods for robust speech recognition and on language analysis using the Embodied Construction Grammar formalism. Finally, the results of human-subject experiments using a novel *Wizard and Operator* model are analyzed with an eye to creating more felicitous interaction in dialogue systems.

1 Introduction

The SMARTKOM-Mobile application provides navigation and tourism information using either a handheld personal digital assistant (PDA) interface or an in-car interface. Some images from the application's display are shown in Fig. 1. A user communicates with SMARTKOM-Mobile using pointing gestures and natural speech, and the system responds with speech (from an animated agent, displayed on-screen) and the display of images and text. The natural and reliable conversational interaction that this calls for provided motivation for a range of research. In this section and in Sect. 2 we describe the work required to port SMARTKOM-Mobile to the English language and the design of the English speech recognizer. In the following sections, we describe research on robust speech recognition, language analysis, and human–computer interaction carried out by the SMARTKOM-English team.

The development of an English-language SMARTKOM-Mobile verified the language portability of the SMARTKOM architecture and facilitated the demonstration of SMARTKOM at international conferences. Staff and visiting researchers at ICSI and staff at DFKI took the lead roles in the creation of the English-language system, and important contributions came from several other SMARTKOM partner sites.

The English speech recognizer was developed completely independently from the German one; it is a hybrid connectionist system descended from the one used in

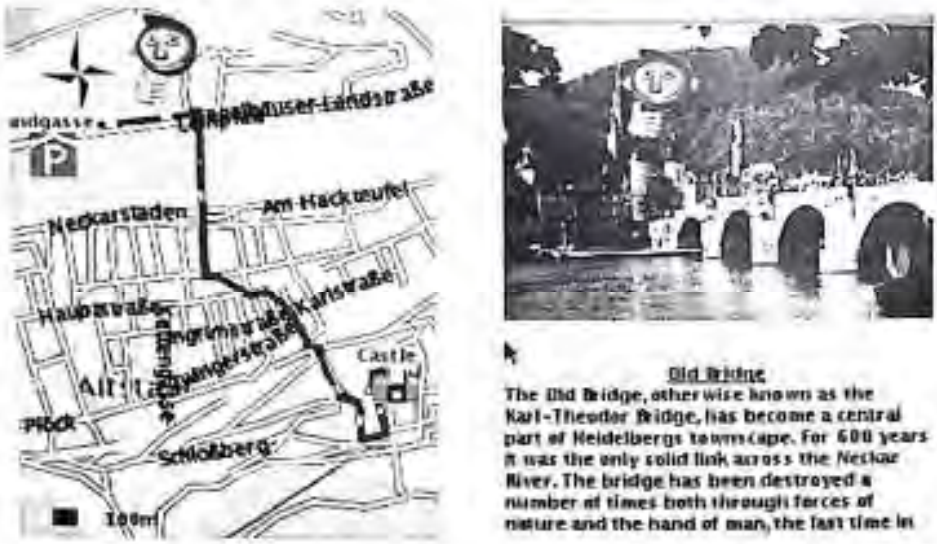


Fig. 1. SMARTKOM-Mobile screen shots showing pedestrian navigation (*left*) and tourist site information (*right*)

the BERP dialogue system project (Jurafsky et al., 1994). All other modules in the English SMARTKOM-Mobile system were based on, or identical to, modules in the German-language SMARTKOM-Mobile.

The modular architecture of SMARTKOM greatly eased porting to English by encapsulating language dependencies in specific modules. Most SMARTKOM modules required no modification to support English. Of the modules requiring modification, only speech recognition and speech synthesis required significant changes to software source code. The speech analyzer (which parses the recognized speech) and text generator (which creates the system's output sentences) required only a change in their template (grammar) files and otherwise used the same software engines for both German and English. The lexicon module had to be aware of the current language in order to provide the correct word pronunciations. Some displayed text provided by the pedestrian and vehicle navigation modules (such as tourist site information and map labels) was translated to English. If dynamic help had been included in the English system, some additional displayed text would have required translation. The speech analyzer outputs a language-independent semantic representation of the user input, and so modules which tracked dialogue state and user intention did not need to be language-aware.

2 The SmartKom-English Speech Recognizer

2.1 Overview

The recognizer uses the hidden Markov model (HMM) approach to speech recognition illustrated in Fig. 2. This approach models speech as a sequence of observations

sampled from different possible probability distributions, with a distinct distribution corresponding to each member of a finite set of possible hidden states. In our recognizer the hidden states represent phones, and the observations are assumed to be acoustic realizations of those phones. Each observation represents a single frame of time. The time step from the start of one frame to the start of the next frame is 16 ms and the length of each frame is 32 ms; the resulting overlap between frames is useful since the frame boundaries are not necessarily aligned with phone boundaries. The observations being modeled are not the original audio but rather are the output of a feature extraction process intended to reduce dimensionality and discard irrelevant variation. We used perceptual linear prediction (PLP) feature extraction (Hermansky, 1990), which captures the envelope of the frame power spectrum but discards some spectral detail. The probability of a particular observation for each possible hidden state is determined by an acoustic modeling stage, which we carry out using a multilayer perceptron (MLP), following the hybrid connectionist approach of Bourlard and Morgan (1993). The probabilities determined by the acoustic modeling stage for all frames are used by a decoding stage that searches for the most likely sentence, taking into account a dictionary of word pronunciations and a language model (tables of word transition probabilities).

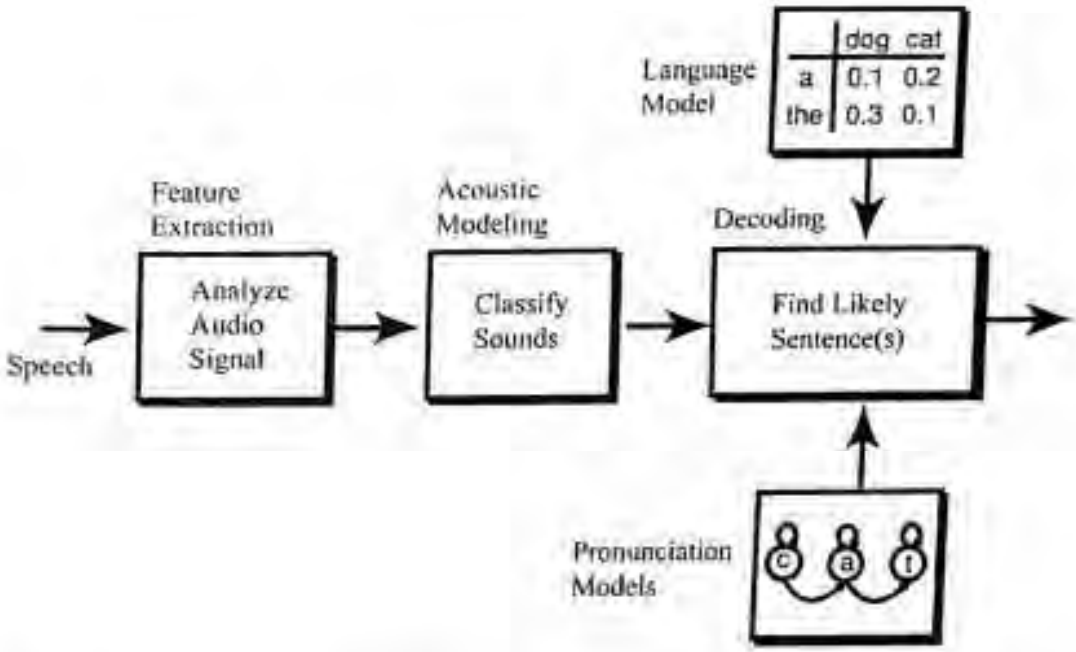


Fig. 2. Speech recognizer architecture

The English speech recognizer was built as a chain of small tools used in a pipeline, communicating with each other using Unix pipes. This simple, modular design makes adding or upgrading speech recognizer components easy. A wrapper program starts the pipeline and handles communication with other SMARTKOM mod-

iles. Feature extraction is performed by ICSI's RASTA tool. Utterance-level mean and variance normalization of the features and MLP output calculation is performed by ICSI's FFWD-NORM tool. Decoding was performed by the NOWAY tool (Renals and Hochberg, 1996); changes made to the NOWAY tool to support the needs of the SMARTKOM project are described below. Source code for all these tools is available free for research use.

2.2 Language Modeling and Decoding

The speech recognizer uses a trigram language model with backoff to bigrams and unigrams. The SRI language modeling toolkit (Stolcke, 2002) was used to estimate the language model from training data that consisted of sample dialogues, partly edited by hand to achieve good coverage of what were perceived to be natural user inputs. Since no naturally collected English data was available, we relied on English translations of German dialogues, taking care to produce idiomatic, rather than literal translations.

To meet the needs of SMARTKOM, we modified the NOWAY decoder to use the C++ libraries associated with the SRI LM toolkit to access language models. This modification allowed NOWAY to be used with class-based language models. Class-based LMs can include class labels as part of N-grams, which are then expanded by a list of class member words. Class-based models have two key advantages for SMARTKOM. First, by using classes the LM generalizes better to novel word sequences, which is especially important given the scarcity of training data. For example, a DIRECTION class was used to stand for possible map directions (e.g., left, right, up, down, east, west, north, and south) in training sentences. To achieve generalization, the word classes are defined by hand based on task knowledge, and the appropriate words are replaced by class labels in the training data.

The second key function of word classes is that they allow new class members to be added on the fly while SMARTKOM is running, without reestimating the entire language model. In the English Mobile application this occurs with parking garage names, which are retrieved from the car navigation module. Parking garage names can occur any time the class name GARAGE occurs in the language model, thereby covering sentences such as

- Can you tell me more about GARAGE
- I would like to know more about GARAGE
- I'd like to know more about GARAGE

3 Signal Processing for Robust Speech Recognition

3.1 Introduction

Compared to recordings made using a close-talking microphone placed near the user's mouth (e.g., a headset microphone), recordings from more distant microphones have higher levels of background noise relative to the speech level (since

the speech level is lower) and are subject to reverberation and other effects due to the longer and sometimes indirect paths taken by the traveling sound waves. While these degradations affect human speech recognition performance as well, current automatic speech recognition systems are much more sensitive to them. However, close-talking microphones are often inconvenient, and improvements in the recognition accuracy that can be achieved without them are very likely to increase adoption of speech recognition technology.

The SMARTKOM-Mobile application can be used inside a car (using an installed display and microphones) or on foot (using the microphone and display on a handheld computer or PDA).[3] Of these two circumstances, ICSI research focused on the in-car case, making use of the SpeechDatCar (Moreno et al., 2000) corpus. The SpeechDatCar corpus is available in several languages; in this article we will only describe results for SpeechDatCar-German. This contains in-car recordings of German connected digit strings made simultaneously with close-talking and hands-free microphones, in various noise conditions: "Stop Motor Running," "Town Traffic," "Low-Speed Rough Road," and "High-Speed Good Road." Our speech recognition experiments with this corpus were performed for three cases: the well-matched case used all microphone types and noise conditions in both training and test data, the medium-matched case used only the hands-free microphone and tests using only the "High-Speed Good Road" noise condition (which is excluded from the training data in this case), and the highly mismatched case used close-talking microphone training data and hands-free microphone test data.

3.2 Noise Reduction and Deconvolution

Figure 3 shows a model of how acoustic degradation is caused by reverberation and background noise. Reverberation and other acoustic effects related to the transmission of speech from talker to microphone, together with the frequency response of the microphone itself, are modeled as a linear time-invariant system with impulse response $c(n)$ and frequency response $C(\omega)$. Background noise (including noise internal to the microphone itself) is additive after this system. We investigated the effectiveness of some signal processing techniques based on this model.

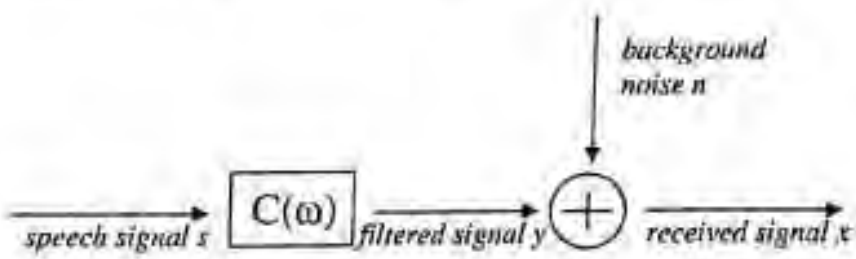


Fig. 3. Model of acoustic degradation

[3] In in-car applications, non-close-talking microphones are sometimes referred to as *hands-free* microphones, as in hands-free mobile phone operation.

3.2.1 Noise Reduction

We used a noise reduction implementation developed for an Aurora (Hirsch and Pearce, 2000) front-end proposal, a joint effort between ICSI, OGI, and Qualcomm engineers, described in Adami et al. (2002). The algorithm performs Wiener filtering with modifications such as a noise overestimation factor, smoothing of the filter response, and a spectral floor. It calculates an instantaneous noise power spectral estimate $|\dot{N}(m,k)|^2$ (where k is the frequency bin and m is the frame index) by averaging the noisy power spectra $|X(m,k)|^2$ over an initial period before speech starts as well as later frames, which are judged to be nonspeech because their energy level falls below a threshold. This estimate is used to calculate a filter

$$|H(m,k)| = max\left(\frac{|X(m,k)|^2 - \alpha|\dot{N}(m,k)|^2}{|X(m,k)|^2}, \beta\right),$$

where α is an SNR-dependent oversubtraction factor and the spectral floor parameter β is used to avoid negative and very small filter values. This filter $|H(m,k)|$ varies from frame to frame. To reduce artifacts in the noise-reduced output, the filter is smoothed over time and frequency. Then the smoothed filter is applied to the noisy power spectra X to obtain an estimate of the noise-free power spectra Y:

$$|\dot{Y}(m,k)|^2 = max(|X(m,k)|^2 * |H(m,k)|^2, \beta_{final} * |\dot{N}(m,k)|^2),$$

where β_{final} is a second spectral floor parameter, which specifies the floor as a fraction of the estimated noise power.

This Wiener filtering approach is based on the assumption that the noise power spectrum is steady. This is probably a fairly good assumption for engine and wind noise during unchanging driving conditions.

Table 1 shows word error rates (WER) on SpeechDatCar-German using the Aurora baseline speech recognizer described in Hirsch and Pearce (2000), with and without noise reduction preprocessing. The noise reduction is very effective in the medium-mismatch condition. In the high mismatch condition it is counterproductive; perhaps the application of noise reduction to the close-talking microphone training data makes that data an even worse match for the hands-free microphone test data. For these experiments, the noise estimation was performed independently for each utterance, and we used overlap-add resynthesis to create noise-reduced output waveforms. This allowed the noise reduction to be used with existing feature extraction code without modifying that code. This noise reduction approach was added to the SMARTKOM-English system as a new pipeline stage preceding the RASTA feature extraction tool.

3.2.2 Deconvolution by Mean Subtraction

Deconvolution by mean subtraction is commonly employed in speech recognition systems, most often via the cepstral mean subtraction (CMS) algorithm. The reasoning behind it is as follows. Consider a discrete-time speech signal $s(n)$ (the origin

Table 1. SpeechDatCar-German word error rates (WER) with and without noise reduction. The well-matched test data contain 5009 words, the medium-matched test data contain 1366 words, and the highly mismatched test data contain 2162 words.

WER (%)	Without noise reduction	With noise reduction
Well-matched	8.0	7.0
Medium-matched	20.6	14.9
Highly mismatched	15.4	17.5

of speech as a continuous-time signal is ignored here for simplicity) sent over a linear time-invariant channel with impulse response $c(n)$, producing a filtered signal $y(n)$. Then the product property holds for the spectra of s and c: $Y(\omega) = S(\omega)C(\omega)$. Assume that the filtered signal is processed using a windowed discrete Fourier transform, giving $Y(m,\omega)$, where m is the frame index determining around which sample the window function was centered. If the window function is long and smooth enough relative to $c(n)$, then the product property approximately still holds (Avendano, 1997): $X(m,\omega) \approx S(m,\omega)C(\omega)$. Note that the channel is assumed not to vary over time. Taking the logs of the magnitudes of both sides of the previous equation, we find that $log|X(m,\omega)| \approx log|S(m,\omega)| + log|C(\omega)|$. Therefore, by subtracting the mean over m (i.e., over time) of $log|X(m,\omega)|$ from $log|X(m,\omega)|$, we remove $log|C(\omega)|$ and the mean over time of $log|S(m,\omega)|$. If $s(n)$ is speech, and the mean is calculated over a large enough number of frames, then we expect the mean of $log|S(m,\omega)|$ to contain little linguistic information, and therefore its removal need not be detrimental to speech recognition performance. Thus the subtraction can be used to compensate for the magnitude response of the channel (for various reasons, there is usually no attempt to compensate the phase response). Speech recognition feature extraction is usually essentially based on windowed discrete Fourier transforms. However, following the transform it is typical to do further processing like Mel/Bark-scale filter bank integration and cepstral transformation, and most often the mean subtraction is done after that processing. The cepstral transformation, being linear, does not affect the reasoning above, but the filter bank integration implies an additional assumption that the channel frequency response is close to constant across the frequency bins that are being integrated.

Table 2 shows WER on SpeechDatCar-German with and without mean subtraction methods added to the Aurora baseline speech recognizer described in Hirsch and Pearce (2000), which uses Mel-frequency cepstral coefficient (MFCC) feature extraction and by default does not perform mean subtraction.

Cepstral mean subtraction (CMS) removes the mean across frames from the MFCCs. The table shows CMS was most helpful in the well-matched and highly mismatched cases, where both close-talking and hands-free recordings are used. This is not surprising because there is a channel mismatch between close-talking and hands-free recordings. The reasoning behind mean subtraction assumes that all frames contain speech, while in fact the data is a mix of speech and pauses. Calculating the mean only over frames judged by a multilayer perceptron classifier to contain speech resulted in a significant performance improvement, which is consistent with the results

for speech recognition over telephone connections in Mokbel et al. (1996). We also tried the log-DFT mean normalization (LDMN) proposed in Neumeyer et al. (1994). In this method the mean subtraction occurs midway through the MFCC computation, before the filter bank integration, so the assumption of channel constancy across bins being integrated is not required. At ICSI, we have sometimes found this to be more effective than CMS, but on this task it does not perform better (in fact, there is a slight, though not statistically significant, drop in performance).

Table 2. SpeechDatCar-German word error rates (WER)

WER (%)	Baseline	Noise reduction alone	Noise red and CMS	Noise red. and CMS; mean taken over speech frames	Noise red. and LDMN; mean taken over speech frames
Well-matched	8.0	7.0	6.7	6.1	6.1
Medium-matched	20.6	14.9	15.7	14.6	15.2
Highly mismatched	15.4	17.5	14.7	11.3	11.0

For further results, including other mean subtraction methods and other data sets, see Gelbart (2004).

3.3 Gabor Filtering

The noise reduction and mean subtraction approaches described above are intended to increase the robustness of existing feature extraction methods by adding additional processing. Another approach, which can be complementary, is to create new feature extraction methods which have desirable properties. We collaborated with Michael Kleinschmidt of the Universität Oldenburg on his project on Gabor filter feature extraction for automatic speech recognition (Kleinschmidt and Gelbart, 2002; Kleinschmidt, 2002). Gabor filters are a family of two-dimensional filters, which Kleinschmidt proposed to use for feature extraction by convolving Gabor filters with a time-frequency representation such as a mel-band spectrogram (see Fig. 4). Depending on what Gabor filters are used, this can behave similarly to short-term spectral envelope-based feature extraction approaches like the popular MFCC and PLP methods, or to the TRAPS (Jain and Hermansky, 2003) approach of long-term analysis in narrow frequency bands, or it can look for patterns at an oblique angle to the time and frequency axes. In Kleinschmidt's approach the Gabor filters used are chosen by a data-driven selection procedure which searches for Gabor filters that appear likely to give good classification performance.

The second column of Table 3 gives WER on SpeechDatCar-German using the back end of the Aurora baseline speech recognizer (Hirsch and Pearce, 2000) with the QIO-NoTRAPS feature extraction module developed by Qualcomm, ICSI, and OGI (Adami et al., 2002), which calculates robust MFCC features using techniques

such as noise reduction by Wiener filtering and mean subtraction. When features derived from Gabor analysis were concatenated to the robust MFCC features to form a longer feature vector, the error rate decreased, as shown in the third column. We found that for good performance with the Gabor filters it was necessary to pass them through a stage of nonlinear discriminant analysis by MLP (the back end used a different acoustic modeling approach). For source code for Gabor feature extraction, and additional information about this approach, please refer to our website.[4]

Table 3. SpeechDatCar-German word error rates (WER)

WER (%)	QIO-NoTRAPS	With Gabor analysis
Well-matched	5.8	5.4
Medium-matched	12.1	11.7
Highly mismatched	12.0	11.6

4 Robust, Semantically Rich Language Analysis

The approach to analysis (parsing and semantic analysis) of recognized speech normally used in the SMARTKOM system was Ralf Engel's SPINmodule (Engel, 2002). The SMARTKOM project also funded investigation into an alternative approach, aimed at robust and semantically rich language analysis. The alternative approach, described here, makes use of a linguistically sophisticated language formalism called Embodied Construction Grammar (ECG (Bergen and Chang, 2002; Chang et al., 2002). ECG is a construction-based grammar formalism (Goldberg, 1995) that uses embodied primitives like frames (Fillmore, 1982), image schemas (Lakoff, 1987) and executing schemas (Narayanan, 1997) as its semantic representation. In addition to supporting these cognitive primitives, ECG is an extension of unification-based formalisms like HPSG (Pollard and Sag, 1994), and as a consequence, it is precise enough for computational models of language analysis.

ECG is an extremely expressive grammar formalism, and as such new algorithms needed to be designed that could take advantage of the wealth of semantic information contained in an ECG grammar, and yet still efficiently and robustly process each utterance. These algorithms are implemented within the so-called constructional analyzer (Bryant, 2003).

The first key innovation employed in the design of the constructional analyzer is how it combines linguistic knowledge with process. Instead of treating each construction as a passive piece of grammatical knowledge, the analyzer compiles each construction into an active unit called a construction recognizer. Each construction recognizer is responsible for applying both the form and the semantic constraints associated with its construction. The recognizer then generates an instance of the construction if an acceptable set of constituents is found.

[4] http://www.icsi.berkeley.edu/Speech/papers/icslp02-gabor/

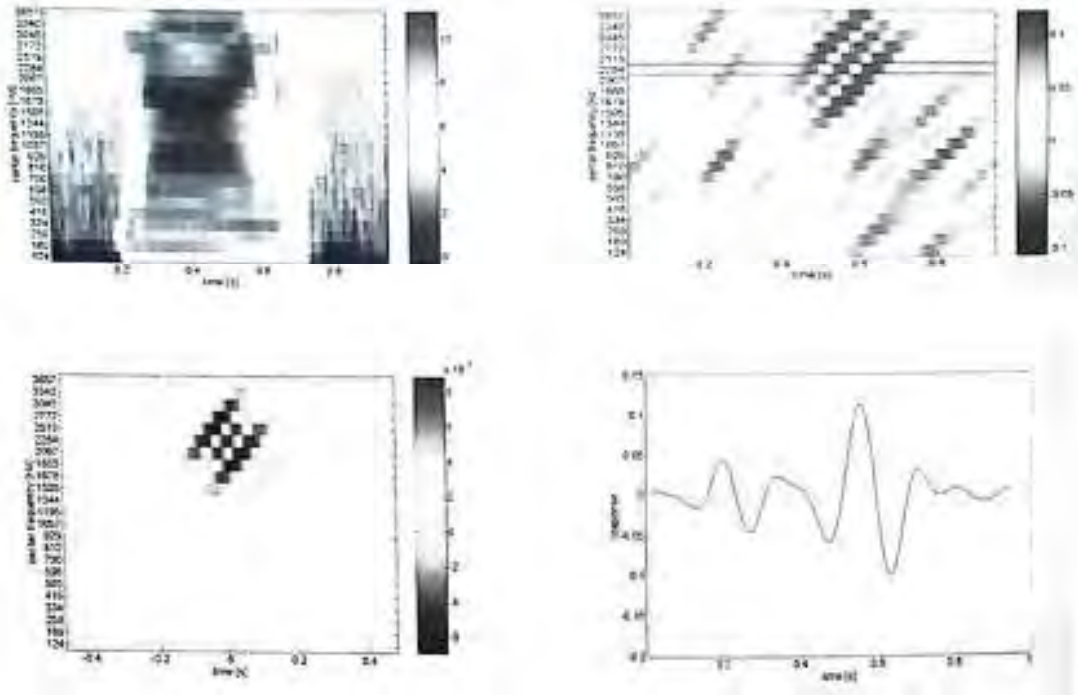


Fig. 4. *Upper left*: A log mel-band spectrogram for the spoken word "nine". *Lower left*: The real part of a Gabor filter. The Gabor filters are complex-valued. Rather than doing acoustic modeling with complex-valued features, Kleinschmidt chose to use the real part, imaginary part, or magnitude of the results of correlation as features. *Upper right*: The real part of the log mel-band spectrogram after correlation with the Gabor filter. *Lower right*: The values of the filtered log mel-band spectrogram at the center frequency of the Gabor filter (it is those that would be used as features for speech recognition)

The analyzer itself must manage the interaction between each of the recognizers, while looking for semantically and formally complete analyses of the whole utterance. The analyzer also has the responsibility for robustly responding when an analysis of the whole utterance cannot be found because of an unforeseen syntactic pattern. This scenario leads us to the second key conceptual innovation employed in the constructional analyzer: leveraging semantics for robust analysis.

Given the rich semantics found in an ECG grammar, the strategy for robust behavior is twofold. First, the analyzer needs a way to infer likely identifications between frames. Such a strategy is specified by the Parsimony Principle (Kay, 1987). It states that the ideal reader unifies compatible frames whenever possible. Applying this principle to a collection of frames and schemas simulates the identifications that were likely to result had a complete analysis been found.

Second, the analyzer needs a mechanism for choosing between the competing analyses that result from application of the Parsimony Principle. The heuristic used for this task is called semantic density (Bryant, 2003). Semantic density defines the completeness of an analysis as the ratio of filled frame roles to total frame roles.

Analyses that are more semantically dense specify more of the frame roles in the utterance than those that are less semantically dense, and the analyzer thus prefers denser analyses.

The constructional analyzer was tested by integrating it with the SMARTKOM domain and context model (Porzel et al., 2006), taking semantic information from that module's ontology. Not only did it successfully analyze questions from the tourist domain, but it also capitalized on the structure found with the ontology to perform linguistic type-coercion known as construal (Porzel and Bryant, 2003).

5 Wizard and Operator Study of Felicitous Human Computer Interaction[5]

5.1 Introduction

End-to-end evaluations of conversational dialogue systems with naive users are currently uncovering severe usability problems that result in low task completion rates. Preliminary analyses suggest that these problems are related to the system's dialogue management and turn-taking behavior. We present the results of experiments designed to take a detailed look at the effects of that behavior. Based on the resulting findings, we spell out a set of criteria which lie orthogonal to dialogue quality, but nevertheless constitute an integral part of a more comprehensive view on dialogue *felicity* as a function of dialogue quality and efficiency.

Research on dialogue systems in the past has focused on engineering the various processing stages involved in dialogical human–computer interaction (HCI), e.g., robust automatic speech recognition, intention recognition, natural language generation, or speech synthesis (Allen et al., 1996; Cox et al., 2000; Bailly et al., 2003). Alongside these efforts, the characteristics of computer-directed language have also been examined as a general phenomenon (Zoeppritz, 1985; Wooffitt et al., 1997; Fraser, 1993; Darves and Oviatt, 2002). The flip side, i.e., computer–human interaction (CHI), has received very little attention as a research question by itself.

The intuitive usability of such conversational dialogue systems can be demonstrated by usability experiments with real users that employ the PROMISE evaluation framework (Beringer et al., 2002), which offers some multimodal extensions over the PARADISE framework (Walker et al., 2000). The work described herein constitutes a starting point for a scientific examination of the "whys" and "wherefores" of the challenging results stemming from such end-to-end evaluations of conversational dialogue systems.

One of the potential reasons for the problems thwarting task completion stems from the problem of *turn overtaking*, which occurs when users rephrase questions or make a second remark to the system while it is still processing the first one. After such occurrences a dialogue becomes asynchronous, meaning that the system responds to the second-last user utterance while in the user's mind that response con-

[5] Robert Porzel and Manja Baudis were the principal authors of this section.

cerns the last. Given the state of the art regarding the dialogue handling capabilities of HCI systems, this inevitably causes dialogues to fail completely.

5.2 Wizard and Operator Study

Here, we describe a new experimental paradigm and the first corresponding experiments tailored toward examining the effects of the computer's communicative behavior on its human partner. More specifically, we will analyze the differences in human–human interaction (HHI) and HCI/CHI turn-taking and dialogue management strategies, which constitutes a promising starting point for an examination of the effects of the computer's communicative behavior on the felicity and intuitiveness of dialogue systems. The overall goal of analyzing these effects is for systems to become usable by exhibiting a more felicitous communicative behavior. After reporting on the results of the experiments in Sect. 5.3, we highlight a set of hypotheses that can be drawn from them and finally point toward future experiments that need to be conducted to verify these hypotheses in Sect. 5.4.

For conducting the experiments we developed a new paradigm for collecting telephone-based dialogue data, called *Wizard and Operator Test* (WOT), which contains elements of both Wizard-of-Oz (WOZ) experiments (Francony et al., 1992) as well as Hidden Operator Tests (HOT (Rapp and Strube, 2002)). This procedure also represents a simplification of classical end-to-end experiments, as it is much like WOZ and HOT experiments conductable without the technically very complex use of a real conversational system. As postexperimental interviews showed, this did not limit the feeling of *authenticity* regarding the simulated conversational system by the human subjects. The WOT setup is described in detail by Porzel and Baudis (2004) and Gurevych and Porzel (2006). It consists of two major phases that begin after the subject has been given a set of tasks to be solved with the telephone-based dialogue system. In the first phase the human assistant is acting as a wizard who is simulating the dialogue system by operating a speech synthesis interface. In the second phase, which starts immediately after a system breakdown has been simulated by means of beeping noises transmitted via the telephone, the human assistant is acting as a *human* operator asking the subject to continue. In our experiments, subjects used the simulated dialogue system to gather information related to tourism in the city of Heidelberg. Simulating a telephone-based dialogue system (rather than a local multimodal dialogue system such as the SMARTKOM-Mobile demonstrator) allowed a natural-seeming switchover from computer–human interaction to human–human interaction.

The experiments were conducted in the English language at ICSI in California. A total of 25 sessions were recorded. At the beginning of the WOT, a person acting as test manager told the subject that they were testing a novel, telephone-based dialogue system that supplies tourist information on the city of Heidelberg. In order to avoid the usual paraphrases of tasks worded too specifically, the manager gave the subjects an overall list of 20 very general tourist activities, such as *visit museum* or *eat out*, from which each subject had to pick 6 tasks that were to be solved in the experiment. The manager then removed the original list, dialed the system's number on

the phone, and exited from the room after handing over the telephone receiver. The subject was always greeted by the system's standard opening ply: *Welcome to the Heidelberger tourist information system. How I can help you?* After three tasks were finished (some successful, some not) the assistant simulated the system's breakdown and came onto the telephone line saying *Excuse me, something seems to have happened with our system, may I assist you from here on*, and finishing the remaining three tasks with the subjects.

5.3 Experimental Results

The PARADISE framework (Walker et al., 1997, 2000) proposes distinct measurements for dialogue quality, dialogue efficiency, and task success metrics. The remaining criterion, i.e., user satisfaction, is based on questionnaires and interviews with subjects and cannot be extracted (sub)automatically from log-files. The measurements described here mainly revolve around dialogue efficency metrics in the sense of Walker et al. (2000). As we will show below, our findings show that a felicitous dialogue is not only a function of dialogue quality, but critically hinges on a minimal threshold of efficiency and overall dialogue management as well. While these criteria lie orthogonal to the Walker et al. (2000) criteria for measuring dialogue quality (such as recognition rates), we regard them to constitute an integral part of an aggregate view on dialogue quality and efficiency, here referred to as *dialogue felicity*. For examining dialogue felicity we will provide detailed analyses of efficiency metrics per se as well as additional metrics for examining the number and effect of pauses, the employment of feedback and turn-taking signals, and the amount of overlaps.

First of all, we apply the classic Walker et al. (2000) metric for measuring dialogue efficiency, by calculating the number of turns over dialogue length on the collected data. (The average length of a dialogue was 6 minutes. The subjects featured approximately uniform distributions of gender, age (12–71), and computer expertise.) As the discrepancy between the dialogue efficiency in phase 1 (HHI) versus phase 2 (HCI) of the experiment might be accounted for by latency times alone, we calculated the same metric with and without pauses. For these analyses, pauses are very conservatively defined as silences during the conversation that exceeded one second.

The overall comparison, given by Porzel and Baudis (2004), shows that naturally latency times severely decrease dialogue efficiency, but also that they alone do not account for the difference in efficiency between human–human and human–computer interaction. This means that even if latency times were to vanish completely, yielding actual real-time performance, we would still observe less efficient dialogues in HCI. While it is obvious that the existing latency times increase the number and length of pauses of the computer interactions as compared to the human operator's interactions, there are no such obvious reasons why the number and length of pauses in the human subjects' interactions should differ in the two phases. However, they do differ substantially.

Next to this *pause effect*, which contributes greatly to dialogue efficiency metrics by increasing dialogue length, we have to take a closer look at the individual turns and their nature. While some turns carry propositional information and constitute utterances proper, a significant number solely consist of specific particles used to exchange signals between the communicative partners, or combinations of such communicative signals with propositional information. We differentiate between dialogue-structuring signals and feedback signals in the sense of Yngve (1970). Dialogue-structuring signals — such as hesitations like *hmm* or *ah* as well as expressions like *well*, *yes*, *so* — mark the intent to begin or end an utterance, or to make corrections or insertions. Feedback signals, while sometimes phonetically alike — such as *right*, *yes* or *hmm* — do not express the intent to take over or give up the speaking role, but serve as a means to stay in contact, which is why they are sometimes referred to as *contact signals*. All dialogues were annotated manually for dialogue structuring and feedback particles.

The data show that feedback particles almost vanish from the human–computer dialogues — a finding that corresponds to those described above. This linguistic behavior, in turn, constitutes an adaptation to the employment of such particles by the respective interlocutor. Striking, however, is that the human subjects still attempted to send dialogue structuring signals to the computer, which — unfortunately — would have been ignored by today's "conversational" dialogue systems. (In the data the subject's employment of dialogue structuring particles in HCI even slightly surpassed that of HHI.)

Most overlaps in human–human conversation occur during turn changes, with the remainder being feedback signals that are uttered during the other interlocutor's turn (Jefferson, 1983). In the collected data the HHI dialogues featured significantly more overlap than the HCI ones, which is partly due to the respective presence and absence of feedback signals as well as to the fact that in HCI turn-taking is accompanied by pauses rather than immediate overlapping handovers.

Lastly, our experiments yielded negative findings concerning the type-token ratio and syntax. This means that there was no statistically significant difference in the linguistic behavior with respect to these factors. We regard this finding to strengthen our conclusions, that emulating human syntactic and semantic behavior does not suffice to guarantee effective and therefore felicitous human–computer interaction.

5.4 Analysis of the Results

The results presented above enable a closer look at dialogue efficiency as one of the key factors influencing overall dialogue felicity. As our experiments show, the difference between the human–human efficiency and that of the human–computer dialogues is not solely due to the computer's response times. There is a significant amount of *white noise*, for example, as users wait after the computer has finished responding. We see these behaviors as a result of a mismanaged dialogue. In many cases users are simple unsure whether the system's turn has ended or not and consequently wait much longer than necessary.

The situation is equally bad at the other end of the turn-taking spectrum, i.e., after a user has handed over the turn to the computer, there is no signal or acknowledgment that the computer has taken the baton and is running with it — regardless of whether the user's utterance is understood or not. Insecurities regarding the main question, i.e., *whose turn is it anyways*, become very notable when users try to establish contact, e.g., by saying *hello* —pause— *hello*. This kind of behavior certainly does not happen in HHI, even when we find long silences.

Examining why silences in human–human interaction are unproblematic, we find that such silences are being announced, e.g., by the human operator employing linguistic signals, such as *just a moment please* or *well, I'll have to have a look in our database* in order to communicate that he is holding on to the turn and finishing his round.

To push the relay analogy even further, we can look at the differences in overlap as another indication of crucial dialogue inefficiency. Since most overlaps occur at the turn boundaries, thereby ensuring a smooth (and fast) handover, their absence constitutes another indication why we are far from having winning systems. As the primary effects of the human-directed language exhibited by today's conversational dialogue systems, the experiments showed that dialogue efficiency decreased significantly even beyond the effects caused by latency times. Additionally, human interlocutors ceased in the production of feedback signals, but still attempted to use his or her turn signals for marking turn boundaries, which, however, go ignored by the system. Last, an increase in pausing is observable, caused by waiting and uncertainty effects, which are also manifested by missing overlaps at turn boundaries.

Generally, we can conclude that a felicitous dialogue needs some amount of extrapropositional exchange between the interlocutors. The complete absence of such dialogue controlling mechanisms by the nonhuman interlocutors alone literally causes the dialogical situation to get out of control, as observable in turn-taking and turn-overtaking phenomena. As evaluations show, this way of behaving does not serve the intended end, i.e., efficient, intuitive, and felicitous human–computer interaction.

Acknowledgments

The English Mobile team at ICSI was Johno Bryant, David Gelbart, Eric Lussier, Bhaskara Marthi, Robert Porzel (visiting from EML), Thilo Pfau, Andreas Stolcke, and Chuck Wooters. Contributions to the development, integration, and testing of the English Mobile system also came from Tilman Becker, Ralf Engel, Gerd Herzog, Norbert Reithinger, Heinz Kirchmann, Markus Loeckelt, Stefan Merten, Christian Pietsch, and Hans-Joerg Kroner at DKFI; Antje Schweizter at IMS; Silke Goronzy, Juergen Schimanowski, and Marion Freese at Sony; Hidir Aras at EML; and Andre Berton at DaimlerChrysler. Funding for ICSI and EML participation in the SMARTKOM project was provided by the German Federal Ministry for Education and Research (BMBF). Some of the work described in this chapter received additional support from other sources, including the Canadian Natural Sciences and Engineering Research Council, the Klaus Tschira Foundation, and Qualcomm.

References

A. Adami, L. Burget, S. Dupont, H. Garudadri, F. Grezl, H. Hermansky, P. Jain, S. Kajarekar, N. Morgan, and S. Sivadas. Qualcomm-ICSI-OGI Features for ASR In: *Proc. ICSLP-2002*, Denver, CO, 2002.

J.F. Allen, B. Miller, E. Ringger, and T. Sikorski. A Robust System for Natural Spoken Dialogue. In: *Proc. 34th Annual Meeting of the Association for Computational Linguistics*, pp. 62–70, Santa Cruz, CA , June 1996.

C. Avendano. *Temporal Processing of Speech in a Time-Feature Space*. PhD thesis, Oregon Graduate Institute, 1997.

G. Bailly, N. Campbell, and B. Mobius. ISCA Special Session: Hot Topics in Speech Synthesis. In: *Proc. EUROSPEECH-03*, pp. 37–40, Geneva, Switzerland, 2003.

B. Bergen and N. Chang. Embodied Construction Grammar in Simulation Based Language Understanding. Technical Report TR-02-004, ICSI, 2002.

N. Beringer, U. Kartal, K. Louka, F. Schiel, and U. Türk. PROMISE: A Procedure for Multimodal Interactive System Evaluation. In: *Proc. Workshop "Multimodal Resources and Multimodal Systems Evaluation"*, pp. 77–80, Las Palmas, Spain, 2002.

H. Bourlard and N. Morgan. *Connectionist Speech Recognition: A Hybrid Approach*. Kluwer Academic, Dordrecht, The Netherlands, 1993.

J. Bryant. Constructional Analysis. Master's thesis, University of California Berkeley, 2003.

N. Chang, S. Narayanan, and M. Petruck. From Frames to Inference. In: *Proc. Scalable Natural Language Understanding(SCANALU)*, Heidelberg, Germany, 2002.

R.V. Cox, C.A. Kamm, L.R. Rabiner, J. Schroeter, and J.G. Wilpon. Speech and Language Processing for Next-Millenium Communications Services. *Proc. IEEE-2000*, 88(8):1314–1337, 2000.

C. Darves and S. Oviatt. Adaptation of Users' Spoken Dialogue Patterns in a Conversational Interface. In: *Proc. ICSLP-2002*, Denver, CO, 2002.

R. Engel. SPIN: Language Understanding for Spoken Dialogue Systems Using a Production System Approach. In: *Proc. ICSLP-2002*, pp. 2717–2720, Denver, CO, 2002.

C. Fillmore. Frame Semantics. In: Linguistics Society of Korea (ed.), *Linguistics in the Morning Calm*, Seoul, Korea, 1982. Hanshin.

J.M. Francony, E. Kuijpers, and Y. Polity. Towards a Methodology for Wizard of Oz Experiments. In: *Proc. 3rd Conf. on Applied Natural Language Processing - ANLP-92*, Trento, Italy, 1992.

N. Fraser. Sublanguage, Register and Natural Language Interfaces. *Interacting with Computers*, 5, 1993.

D. Gelbart. Mean Subtraction for Automatic Speech Recognition in Reverberation. Technical Report TR-04-003, ICSI, 2004.

A. Goldberg. *Constructions: A Construction Grammar Approach to Argument Structure*. University of Chicago Press, Chicago, IL, 1995.

I. Gurevych and R. Porzel. Empirical Studies for Intuitive Interaction, 2006. In this volume.

H. Hermansky. Perceptual Linear Predictive (PLP) Analysis of Speech. *Journal of the Acoustical Society of America*, 87(4), 1990.

H.G. Hirsch and D. Pearce. The AURORA Experimental Framework for the Performance Evaluations of Speech Recognition Systems Under Noisy Conditions. In: *Proc. ISCA ITRW ASR2000*, Paris, France, 2000.

P. Jain and H. Hermansky. Beyond a Single Critical-Band in TRAP Based ASR. In: *Proc. EUROSPEECH-03*, Geneva, Switzerland, 2003.

G. Jefferson. Two Explorations of the Organisation of Overlapping Talk in Conversation. *Tilburg Papers in Language and Literature*, 28, 1983.

D. Jurafsky, C. Wooters, G. Tajchman, J. Segal, A. Stolcke, E. Fosler, and N. Morgan. The Berkeley Restaurant Project. In: *Proc. ICSLP-94*, Yokohama, Japan, 1994.

P. Kay. Three Porperties of the Ideal Reader. In: R.O. Freedle and R.P. Durán (eds.), *Cognitive and Linguistic Analyses of Test Performance*, pp. 208–224, Norwood, NJ, 1987. Ablex.

M. Kleinschmidt. *Robust Speech Recognition Based on Spectro-Temporal Processing*. PhD thesis, Carl von Ossietzky-Universität, Oldenburg, Germany, 2002.

M. Kleinschmidt and D. Gelbart. Improving Word Accuracy With Gabor Feature Extraction. In: *Proc. ICSLP-2002*, Denver, CO, 2002.

G. Lakoff. *Women, Fire, and Dangerous Things*. University of Chicago Press, Chicago, IL, 1987.

C. Mokbel, D. Jouvet, and J. Monné. Deconvolution of Telephone Line Effects for Speech Recognition. *Speech Communication*, 19(3), 1996.

A. Moreno, B. Lindberg, C. Draxler, G. Richard, K. Choukri, S. Euler, and J. Allen. SpeechDat-Car: A Large Speech Database for Automotive Environments. In: *Proc. 2nd Int. Conf. on Language Resources and Evaluation (LREC 2000)*, Athens, Greece, 2000.

S. Narayanan. *Knowledge-Based Action Representations for Metaphor and Aspect*. PhD thesis, University of California, Berkeley, CA, 1997.

L. Neumeyer, V. Digalakis, and M. Weintraub. Training Issues and Channel Equalization Techniques for the Construction of Telephone Acoustic Models Using a High-Quality Speech Corpus. *IEEE Transactions on Speech and Audio Processing*, 2(4), 1994.

C. Pollard and I. Sag. *Head-Driven Phrase Structure Grammar*. University of Chicago Press and CSLI Publications, Chicago, IL, 1994.

R. Porzel and M. Baudis. The Tao of CHI: Towards Felicitous Human-Computer Interaction. In: *Proc. Human Language Technology Conference / North American Chapter of the Association for Computational Linguistics Annual Meeting 2004*, Boston, MA, 2004.

R. Porzel and J. Bryant. Employing the Embodied Construction Grammar Formalism for Knowledge Represenation: The Case of Construal Resolution. In: *Proc. 8th Int. Cognitive Linguistics Conference*, Logrono, Spain, 2003.

R. Porzel, I. Gurevych, and R. Malaka. In Context: Integrating Domain- and Situation-Specific Knowledge, 2006. In this volume.

S. Rapp and M. Strube. An Iterative Data Collection Approach for Multimodal Dialogue Systems. In: *Proc. 3rd Int. Conf. on Language Resources and Evaluation (LREC 2002)*, pp. 661–665, Las Palmas, Spain, 2002.

S. Renals and M. Hochberg. Efficient Evaluation of the LVCSR Search Space Using the NOWAY Decoder. In: *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP-96)*, Atlanta, GA, 1996.

A. Stolcke. SRILM — An Extensible Language Modeling Toolkit. In: *Proc. ICSLP-2002*, Denver, CO, 2002.

M.A. Walker, C.A. Kamm, and D.J. Litman. Towards Developing General Model of Usability with PARADISE. *Natural Language Engineering*, 6, 2000.

M.A. Walker, D.J. Litman, C.A. Kamm, and A. Abella. PARADISE: A Framework for Evaluating Spoken Dialogue Agents. In: *Proc. 35th ACL*, Madrid, Spain, 1997.

R. Wooffitt, N. Gilbert, N. Fraser, and S. McGlashan. *Humans, Computers and Wizards: Conversation Analysis and Human (Simulated) Computer Interaction*. Brunner-Routledge, London, UK, 1997.

V. Yngve. On Getting a Word in Edgewise. In: *Papers From the 6th Regional Meeting of the Chicago Linguistic Society*, Chicago, IL, 1970.

M. Zoeppritz. Computer Talk? Technical Report 85.05, IBM Scientific Center Heidelberg, 1985.

SmartKom-Public

Axel Horndasch[1], Horst Rapp[2], and Hans Röttger[3]

[1] Sympalog Voice Solutions GmbH, Erlangen, Germany
horndasch@sympalog.de
[2] MediaInterface Dresden GmbH, Dresden, Germany
rapp@mediainterface.de
[3] Siemens AG, CT IC 5, München, Germany
hans.roettger@siemens.com

Summary. SMARTKOM-Public is the result of consistent development of traditional public telephone booths for members of a modern information society in the form of a multimodal communications booth for intuitive broad-bandwidth communication.

1 Introduction

The SMARTKOM-Public communications booth is provided with a wide palette of modern communication appliances. SMARTKOM-Public offers a higher degree of privacy, much higher communication bandwidth and optimal quality with regard to input and output results than is available to mobile systems for technical reasons. High rates of data flow and high-resolution screens lead, in contrast to a personal digital assistant (PDA), to increased comfort and better ergonomics. This communication space offers the simultaneous and comfortable use of language, pictorial and data services. A document camera is integrated for the transmission and processing of documents. An interesting further possibility concerns the saving of the user's data in the communications booth. Upon biometric identification, users are able to access their personal data. The idea of a virtual communications workstation that is capable of traveling with the users and adapting to their needs becomes reality. As the product at hand deals with a communication space for the public, along with supplying services, intuitive usability is of primary importance for acceptance. This is particularly so as even people who do not have access to a home computer with similar services will also make use of SMARTKOM-Public. The human–machine interface is designed to replicate normal human means of communication. This includes the broadest range of free speech dialogue possible and the use of natural gestures. Consideration must be made that conventional uses, and not only those based on multimodal communication, can be performed from the communication booth. Therefore our project started with a user study. The results are presented in the next section, followed by a description of the implemented functionality and some detailed information on selected realization topics in subsequent sections.

2 User Study

2.1 Introduction

SMARTKOM-Public demonstrates fundamental techniques of a user-friendly, multimodal human–machine communication in public areas. In order to have a close connection between the project specification and the behavior and the wishes of the potential users, the project was accompanied by a user study from the beginning. This study was to give answers to following questions:

- What are the expectations of potential users concerning a multimodal communication cell?
- Which kinds of human–machine communication does the user accept?
- Which communication services have to be provided?
- Which fears and application thresholds are connected with these new techniques?

2.2 Milestones

The data of the user study were obtained by questionnaires in two phases. In a first phase (9/99–5/00) a questionnaires from the technical point of view was elaborated. The questionnaires were adjusted among the SMARTKOM partners. Around 100 persons took part in the first part – mainly „insiders," persons who were directly involved in SMARTKOM or who had a very good knowledge about the project.

In the second phase (6/00–12/00) a more structured questionnaire was developed in collaboration with the Institute for Communication, Information and Education (KIB) of the University of Applied Sciences Zittau/Görlitz. The questions were asked in more detail, and the answers had to be given on a 5-point scale. It took the participants around 30–60 minutes to fill in the questionnaire. The questionnaire was provided in an Internet version, and owing to different marketing strategies, like mailings actions and a quiz, around 400 persons took part in the second phase.

2.3 Questions and Results

2.3.1 Demographic Data ("Who?")

- Gender: 50 % female, 50% male
- Age: 40% below 30 years
  50% between 30 and 50 years
  10% above 50 years

The mean age was 34 years.

- Education: ca. 33% school
  ca. 33% colleges
  ca. 33% universities

- Concerning their professions, there was a broad variety among the participants with a certain concentration of employees.
- Their computer skills were assessed as follows:
  60% good
  30% average
  10% bad

Thus it can be concluded that the study achieved a well-balanced and representative mixture of potential users concerning sex, age, education and skills.

2.3.2 Application Domains ("What?")

With these questions the preferred application domains of the potential users for SMARTKOM services were investigated. The participants were asked to give their opinions on four domains:

- information (timetables, weather, holiday planning, cultural and sport events, tourist and cultural locations, stock values, navigation and complex information)
- reservations (hotel, cultural events)
- orders and booking (tickets, products, travels)
- financial services (bank and stock business)

The potential users gave the following assessments to the application domains:

- Information concerning timetables and routes found a high degree of acceptance.
- Information about locations, events and complex information was also accepted.
- Information about weather, travel and especially about stock values found only a low acceptance.
- Reservations in cinemas and hotels were accepted.
- From the application domains directly connected with financial transactions, ordering of tickets was the only accepted area.
- Ordering of products and the booking of holiday trips were refused, more or less.
- For financial services the SMARTKOM-Public communication cell was not found to be suitable.

2.3.3 Media Services ("Which?")

Within these questions the technological equipment of a multimodal communication cell had to be estimated. The following media could be chosen:

- telephone
- video telephone
- fax
- Internet
- Access to personal mailboxes

The answers coincided in a high degree:

- All the media offered were accepted.
- The telephone as the most usual and best-known means of communication found the highest acceptance.

2.3.4 User Groups and Places of Installation ("Where?")

From these questions it should be concluded which places of installation would be preferred by different user groups. The participants of the questionnaire were offered the following situations:

- business traveler in an unknown city
- vacationer in an unknown city
- inhabitant in their home city
- employee in an office

2.3.5 Interaction and Technology ("By what?")

With these questions the modalities and possibilities of interaction were assessed. The participants had to give their preferences concerning:

- input options (speech, gestures, free input by pencil, mimic, document input)
- output options (text and pictures on a screen, speech, video telephone, printer)
- others (use of personal devices, chipcards for identification and payment)

The participants gave the following adjustments:

- High acceptance rates for input by speech and hand gestures.
- Input by pencil and document camera was also accepted.
- Mimic as input modality was refused. (Here it can be presumed that the importance of mimic for the dialogue was not understood sufficiently. Further, the mimic input requires a permanent video recording of the face, which was not accepted.)
- Among the output modalities speech and visual output were preferred.
- The printer was estimated as very useful (for faxes, roadmaps, receipts for reservations, etc.)
- The chipcard reader and the plug-in possibility for personal devices were also accepted.

2.3.6 Verification ("How?")

For purposes of verification the following items had to be adjusted:

- personal identification number (PIN)
- signature
- voice
- hand contour

- finger print
- face

The results were as follows:

- The well-known methods with chipcards or PIN were accepted.
- Among the biometrical methods, the fingerprint found the highest acceptance.
- Signature and face recognition were refused in many cases.

2.3.7 General Requirements for a Communication Cell

Different questions about general requirements for a communications cell gave the following views:

- There were high expectations concerning the intelligence of the system. More than 70% of the participants expected a successful dialogue with only a few clearing steps.
- The communication cell and the services were regarded more as an office employee. Only a third of the participants saw the technical system in the foreground.
- Discretion and anonymity were expected by the majority of potential users. Alternatively to the loudspeaker, a telephone was demanded. The video recording for the mimic recognition was refused in most of the cases. Around 75% of the participants would not make use of the services if they were connected with personal identification.
- There was a high consensus about the fact that better technological possibilities should be offered at constant prices. The only additional costs accepted were taxes comparable to information services and other special services like fax.

2.3.8 Correlation Analysis

In order to investigate the relations between the judgments of the potential users, their answer were analyzed in a paired correlation matrix. For example, there were high correlations in the answers for questions like "information for holiday travel" and "booking of holiday travel". This seems logical — if somebody wants to book travel, of course they want to get information about it beforehand. From the variety of such logical relations, two main conclusions can be obtained:

- The correlation matrix gives indications for producers of future devices on which applications are compatible with each other and can be integrated into one product.
- The answers were well chosen, which underlines the fact of representative nature of the study.

2.3.9 User Group Analysis

From the analysis according to the gender it can be concluded that female and male participants tended to give the same answers. Women showed less interest in technological details, and their expectations concerning the intelligence of the system were not so high. On the other hand, they were more restrictive in the face recognition problems. The user groups "below 30 years" and "between 30 and 50 years" tended to give the same answers in tendency. The user group "above 50 years" had different opinions on some questions. Elderly people want make less use of the multimodal communication cell in general. They were less interested in the technical possibilities, and they are less restrictive in security problems. The comparison between "insiders" and "others" showed that the answers also were given with the same tendency. The "insiders" however were slightly more optimistic and in agreement with the scenario.

2.3.10 User Suggestions

The questionnaires asked the participants to notify individual suggestions, opinions, apprehension or ideas. Fortunately, this possibility was used in a high degree. In summary, a catalogue of useful ideas containing details of technical question and general requirements could be obtained. Some of the most interesting statements follow:

Applications

- information and booking for actual events in the city (exhibitions, sport, cinema, conferences, etc.)
- restaurant guide with detailed information concerning the menu and quality assessments
- service for results of sport events
- social and political information about the city/the country
- virtual city tours, information about sights
- regional and global news service
- entertainment for waiting times, Internet games
- city map and city information with guide to locations of cultural interest, administration, service
- information about working hours of offices
- substitution of ways to the office (for registration of cars, new passports, etc.)
- flat and property service
- job and qualification announcements

Locations

- villages and small towns for elderly and less mobiles persons
- central places and transportation hubs for business and tourism
- inner city and important/big buildings (post and other offices, shopping malls...)
- medical centers, parking areas, truck stops, schools and universities

General Requirements

- interaction results should not be obvious to any but current user
- safety from vandalism
- anonymity, only PIN accepted
- guarantee of data security
- ease of use for physically challenged people
- service free of charge in the tourist information centers
- special services (fax, etc.) at normal charges
- clearly structured with a good overview in each application domain
- simple and intuitive handling

2.4 Summary

In the study, a representative variety of potential users took part, thus the results are valuable. The judgments of the application confirmed the SMARTKOM Demo scenarios. The most important locations are tourist and transportation centers. The different input and output modalities were accepted; only the mimic recognition was refused. User verification by PIN and fingerprint was more accepted than by signature or face recognition. The intelligence of the system was expected to be very high in general. The requirements concerning discretion, anonymity and data security were emphasized with a high degree of importance. The criterion "ease of use for physically challenged people" played an important role. The high number of logical correlations proves the representative nature of the study. They give indications for producers of future devices on which applications are compatible with each other and can be integrated into one product. From the point of view of user groups there were significant differences between female and male users. Elderly people showed less interest in SMARTKOM itself and its technological possibilities. The "insiders" of the SMARTKOM scenario showed a more optimistic behavior compared to the "general user." With the verbal notices a valuable collection of ideas concerning application domains and technological improvements is available.

3 Functionality

The SMARTKOM-Public demonstrator provides a wide range of applications considering the results of the user study. The focus is on communication and information services. Since it was not possible to implement all user-suggested services, some exemplary applications were selected:

- phone call as the most common and best-known means of communication, here supported by an electronic address book
- fax and e-mail services to realize the transmission of written documents
- a cinema information and booking system combined with a city map
- multimodal biometric authentification to access personal data

Fig. 1. The former German federal president using SMARTKOM-public e-mail

Altogether, SMARTKOM-Public offers more than 20 functionalities that can be combined in many ways.

These functionalities are accessed through an easy to use multimodal user interface that combines speech, gesture and mimic input with symmetric output. The optical output is projected on an interaction surface, where it can be referenced by natural pointing gestures. This part of the system is based on the Siemens Virtual Touchscreen (SiViT) system (Maggioni and Röttger, 1999). A second dedicated camera scans documents placed on the interaction area for further processing. As demanded by potential users, all applications can be accessed with only a few clearance steps. Although in the user study potential users expressed objections to mimic input, it was integrated into the demonstrator for scientific reasons and to evaluate possible applications.

Another compromise was made on the physical design of the communication booth. On one hand, potential users stressed their need for discretion and anonymity, demanding a traditional closed telephone booth design. On the other hand, SMARTKOM-Public is a showcase to demonstrate new technologies to the interested public, which requires an open design. We decided to use an open information terminal design for the second reason. For the same reason we used hands-free communication instead of a traditional phone handset.

The next sections describe the basic SMARTKOM-Public applications in detail. For more information on the gesture and mimic input, see Racky et al. (2006) and Frank et al. (2006).

Fig. 2. Phone application in SMARTKOM-Public

3.1 Phone and Address book

Just like a traditional public telephone booth SMARTKOM-Public allows users to make standard telephone calls. The life-like character "Smartakus," visible on the projected output, serves as an interface agent to meet users need. Thus SMARTKOM-Public's interaction style breaks radically with the traditional desktop metaphor. It is based on the situated delegation-oriented dialogue paradigm (SDDP (Wahlster et al., 2001). The user simply tells Smartakus to establish a phone connection. This can be done in two different ways:

- dialing manually on a projected phone keyboard
- dialing by a phone book reference

In contrast to other scientific demonstrators, SMARTKOM-Public establishes real phone connections based on the European ISDN standard and thus has to handle signalization events correctly. It also uses hands-free communication based on the SMARTKOM audio modules and provides an address book based on the standard application module (see Sect. 4 for technical details).

Let us close the phone section with a look at two sample phone dialogues:

Example 1: Dialing manually on a projected phone keyboard

USER:	I'd like to make a call.
SMARTAKUS:	(presents phone keys)
	Please dial a number.
USR:	(dials number)
SMARTAKUS:	(establishes connection)

Example 2: Dialing by a phone book reference

USER:	I'd like to call Mr. Blocher.
SMARTKOM	(presents entry for Mr. Blocher)
	I've found an entry for Mr. Blocher.
	Is this correct?
USR:	Okay.
SMARTAKUS:	(establishes connection)

3.2 Facsimiles and E-Mail

Users can send facsimiles (fax) and e-mails with SMARTKOM-Public as well. Smartakus receives the users' requests and guides them through the necessary steps.

Transmission of pictorial information is mainly used for documents related to business or original artifacts that cannot be described in words. Therefore SMARTKOM-Public provides the possibility to capture prepared information with a high-resolution document camera and to send it as a picture to the receiver. For example, it is possible to send an image of your just-purchased new tie — do not try this with a standard fax machine! If the user decides to send an e-mail, the original is captured as a true-color image and sent to the selected e-mail address as an e-mail attachment. For a fax the color image is converted into a black and white version and sent as a real fax to the selected ISDN phone number.

The image transmission functionality is a complex task combining several modules into one service (see Sect. 4 for technical details).

Here is a typical fax dialogue to illustrate the interaction steps:

USER:	I'd like to send a fax to Mr. Wahlster.
SMARTAKUS:	(presents scan area)
	Please place your document here.
USER:	(places the document)
SMARTKOM:	(captures the document)
	Please remove the document.
USER:	(removes document)
SMARTKOM:	The document was captured.
	I've found an entry for Mr. Wahlster.
	Is this correct?

Fig. 3. Fax application in SMARTKOM-Public

USER: Yes.
SMARTAKUS: (sends fax to Mr. Wahlster)

3.3 Cinema

The cinema application was chosen to demonstrate the advantages of the SDDP in a complex information service environment. It was also favored in the user study. The cinema application provides a wide range of choices:

- the current cinema program
- the movie schedule in specific theaters
- information on specific movies
- information on movie actors
- locations of the theaters and directions to those

As in all SMARTKOM-Public applications, the presented information is real and always up to date. A module for external information services scans the Internet for current data. The vocabulary for the speech recognition is updated dynamically as well. Thus it is even possible to ask questions on completely unknown actors, for

Fig. 4. Cinema application in SMARTKOM-Public

example, "Is there a movie with Leonardo DiCaprio?" Here is more serious sample dialogue:

USER:	I'd like to go to the movies tonight.
SMARTAKUS:	(presents a list of movies and a map)
	Here is a selection of tonight's movies.
	I've marked the cinemas on the map.
USER:	(points to the cinema "Kamera" on the map)
	What's on here?
SMARTAKUS:	(presents the program of the cinema "Kamera")
	These are tonight's movies at the "Kamera".
USER:	(points to *Matrix: Reloaded*)
	Tell me more about this movie.
SMARTAKUS:	(presents additional information)
	Here is more information on *Matrix: Reloaded*.
USER:	I'd like two tickets for this film at 8 pm.
SMARTAKUS:	(presents seat plan of the "Kamera")
	Where do you want to sit?
USER:	(encircles specific seats)
	Here.

Fig. 5. Multimodal biometrics in SMARTKOM-Public

SMARTAKUS:	(highlights the selected seats on the plan) Is this correct?
USER:	Exactly.
SMARTAKUS:	I reserved the tickets. Your reservation code is 42.

3.4 Biometrical Authentification

As mentioned in the Introduction, SMARTKOM-Public provides personalized services. Therefore it is essential to determine the identity of the user. This is done by multiple biometrics (see Grashey and Schuster (2006) in this volume). To guarantee comfort and safety, three different biometrics have been integrated: speaker verification, signature verification and hand contour verification. Authentification can be actively initiated by the user with the phrase "My name is" or can be demanded as the situation calls for:

SMARTAKUS:	This action requires an authentification. What's your name?

USER:	My name is "Herzog".
SMARTAKUS:	I've found an entry for Gerd Herzog. Hand contour is the default biometry. Please place your hand with spread fingers on the marked position.
USER:	(puts his hand on the interaction area)
SMARTAKUS:	Authentification accomplished.

4 Technical Realization

As is the case for all SMARTKOM demonstrators, SMARTKOM-Public is based on the common SMARTKOM system architecture handling the dialogue with the user. The SMARTKOM system architecture consists of several independent modules connected via communication pools. Most of these modules have been described in the previous chapters. The following sections describe SMARTKOM-Public–specific modules:

- the communication module supported by the audio modules, realizing phone calls and data transfers on standard ISDN lines
- the module for standard applications, realizing address book, e-mail and calendar functionalities
- the virtual module for sending documents, as a combination of document camera and communication module.

4.1 Communication (TK-Module)

The TK-Module in SMARTKOM realizes the telephone and fax services on a Microsoft Windows PC. It works on the hardware base "Fritz!Card PCI" manufactured by AVM, which provides an ISDN S0 connection with two channels.

The software of the TK-Module is realized in a layer structure. The hardware driver is controlled by CAPI2.0. This is enclosed in a library that provides the functionalities typical for telecommunication. The most important features are the support of n-channel telephone calls and fax, the support of ISDN functions and the conversion between the audio formats PCM (in SMARTKOM) and A Law (in ISDN). The highest layer consists of a console application. It controls the internal workflow and allows the parameter input.

In the "telephone" operation mode the TK-Module works together with the modules AudioIn and AudioOut, which are controlled by the "function modeling." During a telephone call AudioOut is occupied exclusively and thus it cannot submit other system outputs.

Each task for the TK-Module contains the telephone number to be dialed and a name of the connection. This name is also used to identify all the data streams of the connection.

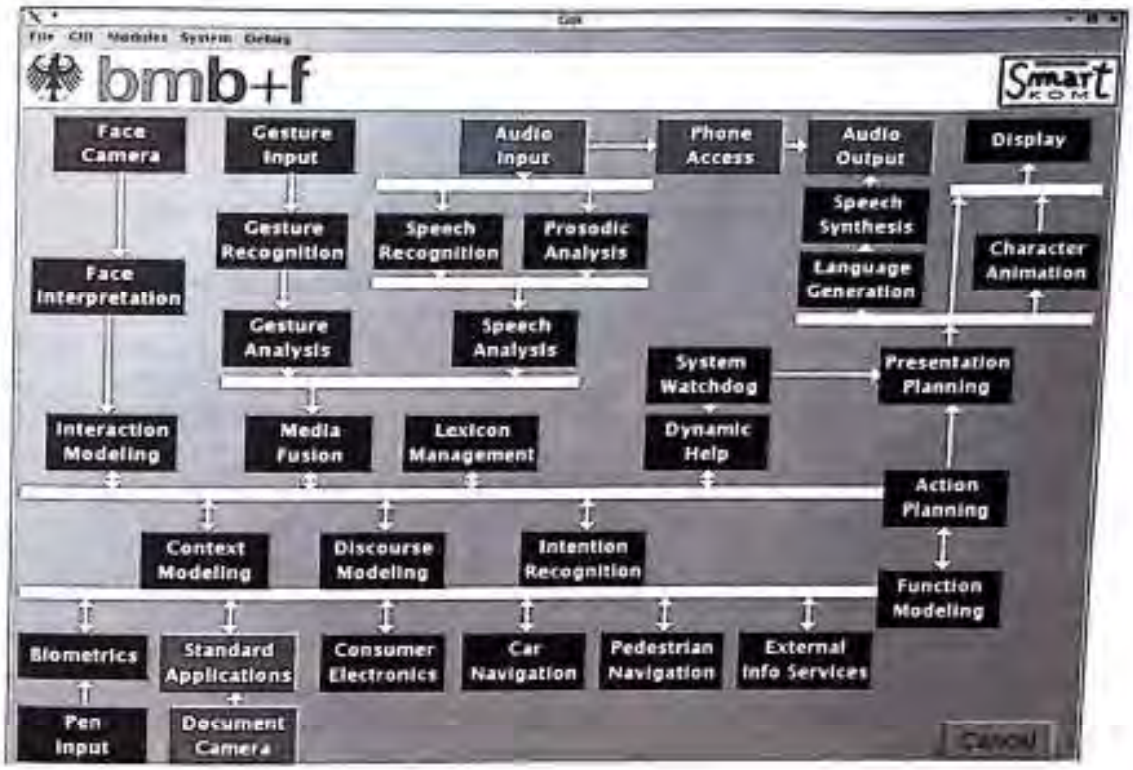


Fig. 6. SMARTKOM-Public module overview

In the "fax delivery" operation mode the TK-Module works together with the "realDocument" module controlled by the "function modeling." The TK-Module receives the documents to be sent in the structured fax file (SFF) format together with the number to be dialed and the name of the connection.

4.2 Audio Modules AudioIn and AudioOut

The AudioIn and AudioOut modules are responsible for the input and output of audio data into and out of SMARTKOM, not only to realize phone calls but also for speech input and output. The modules are realized in different objects:

WIN-32 Console Application: This application realizes the communication of the audio modules with the SMARTKOM system using the SMARTKOM pool architecture supervised by the module manager. It controls the internal workflow and allows the parameter input. It is also responsible for the time synchronization of the audio data.

DLL Object: This multithread capable dynamic link library contains all functions for the communication of SMARTKOM with different audio sources and audio consumers. The most important features are the support of n channels, the mixer functionality and the handling of different PCM audio formats (sample rate, etc.). Audio data can be submitted and delivered local area networks (LAN) and also

by wireless LAN. The connection to the local audio devices is realized by the Multimedia-Microsoft-Win32-API. The remote connection is based on the RTP protocol. The devices for audio input and output can be switched in runtime by parameters.

The audio modules are controlled by the "function modeling" module. The tasks from "function modeling" contain the actual parameters, the purpose and a unique identification of the data stream. Both modules support the synchronous mechanism of SMARTKOM. The actions controlled by the modules can be activated by external events, and the modules themselves can send events related to certain system states.

AudioIn Module: The AudioIn module receives the audio data from the input devices and puts it into chunks of defined formats and sizes. The format definitions are made by the function modeling module. So, for instance, in the operation modes "telephone" and "speaker verification" the data chunks contain 2-KB audio samples with 8-kHz sample rate — thus speech data of 256 ms. In the operation mode "normal" the chunks contain typically 8-KB data sampled with 16 kHz.

The audio data are sent in regular intervals in XML documents. The documents contain time stamps with the absolute time of the first sample of each data chunk. Thus the information of different modalities can be synchronized within the whole system.

In addition, the AudioIn module provides the functionalities "silence detection" and "echo compensation". In the case of silence, the transmission of audio data is going on but with empty data chunks. The echo compensation provided by partner DaimlerChrysler removes the SMARTKOM system messages via loudspeakers from the microphone signals.

AudioOut Module: The AudioOut module provides the output samples for the output devices at the given sample rates. In order to be able to synchronize the audio stream with other output streams like video data, the AudioOut module can start a certain audio output at an absolute time given in the audio document.

4.3 Standard Applications

4.3.1 Introduction

The interface for users of SMARTKOM is intuitive and offers many possibilities, as described in the previous sections. Every interaction with the system can lead to numerous complex activities involving devices and services especially designed and programmed for SMARTKOM. The ability to use standard IT applications within SMARTKOM is, however, also essential.

Groupware applications to manage e-mail, address books or calendars are typical examples of such services. Standard software offering these features has been integrated into SMARTKOM instead of duplicating their functionality within the system. Currently, the supported program suites are Lotus Domino Server and Lotus Notes, which include an application programming interface (API) capable of controlling all necessary groupware components. Other common software products can, however, also take their place.

4.3.2 Lotus Domino Server and Lotus Notes — A Short Overview

Lotus stores all data in objects which can be accessed using an API that exists for different programming languages. These databases, also known as *Notes Storage Facility* or *NSF files*, contain views and/or documents, which are the equivalent of, for example, an address book, an e-mail folder or a calendar. From a technical point of view the Java API that was used to integrate Lotus offers two possible ways of reading or manipulating Lotus data:

- The *local access pattern*, which allows the programmer to open NSF files without the server running in the background. The disadvantage is that the code has to run on the machine on which Lotus is installed.
- The *remote access pattern*, which hides a lot of tedious implementation details. Its drawback is that all methods are invoked through CORBA's *Internet Inter-ORB Protocol (IIOP)* interface, which means that processing is done remotely. Even if no traffic between different computers on the network is created, the bits have to travel through the network adapter of the local machine.

With performance being one of the main issues during the development of SMARTKOM, the local access pattern represented the natural choice for the implementation.

Apart from the possibility of retrieving data from a database or modifying it through its programming interface, Lotus offers standard e-mail protocols such as POP3, IMAP and SMTP, enabling users of the SMARTKOM system to communicate with the outside world.

4.3.3 Lotus Applications Modeled in SmartKom

To satisfy personal communication needs, the following Lotus applications were made available to SMARTKOM users:

- searching the public address book
- sending e-mails
- using the personal calendar

During the course of the multimodal dialogue with SMARTKOM, the system internally generates a plan to meet the demands of the user. The standard applications are employed to perform parts of the overall task through the abstract layer, which models the functionality of the different devices and services. For example, if the user wishes to send e-mail, the retrieval of the address as well as the task of sending the e-mail itself are building blocks of the entire process.

4.3.4 Storing SmartKom User Profiles

All the applications mentioned above make it necessary to log on to the Lotus Domino Server with a user name and a password. Because it is not acceptable to request this information after the user has been authenticated, it has to be stored

somewhere within the SMARTKOM system. Since all users who want to use the standard applications have to be registered on the Lotus Domino Server anyway, it makes sense to store the information relating to the person in the public address book. This also makes it possible to store other pieces of information about users, for example, their preferred method of authentication. Within SMARTKOM these parts of a user profile are only available to the administrator of the Lotus Domino Server through a metasearch in the public address book. When SMARTKOM is initialized, the login information of this metauser is read from the global configuration file. Returning to the example of an e-mail being sent by a SMARTKOM user, it now becomes clear that the standard applications are involved when restoring the user profile, authenticating the person and fulfilling the request of sending the e-mail.

4.3.5 The Import Tool for User Profiles and Address Book Entries

Once Lotus Domino Server and Lotus Notes have been installed, user profiles and contacts which should be accessible to SMARTKOM have to be created on the server. This task is handled by an import tool with a graphical user interface, which lets the SMARTKOM administrator add and edit the relevant data. The offline tool, which supports comma-separated value (CSV) files as input, connects to the server and makes all the desired entries in the address book.

4.3.6 The Interface to Standard Applications

Since all communication within SMARTKOM is carried out using the Multimodal Markup Language (M3L), which is based on XML, it was necessary to design an M3L interface and implement a middleware to pass on the functionality of the groupware applications to the SMARTKOM system. This interface accepts messages with the following content as input:

- an identifier for the data stream
- login information of the user
- the data necessary to complete the request

The data for the request are organized so they can be easily mapped to the data structures of the groupware product.

The output for all requests consists in the simplest case of the following control information:

- the identifier of the data stream that was processed
- the request status

Normally the request status is *finished* or *fail*. In the case that the interaction exceeds a certain time limit, it is set to *pending*. A further description of the output generated requires a consideration of two different kinds of request. First, there are messages that do not expect an answer from the server, apart from status information. Creating, changing or deleting calendar entries are requests of that type, as is

the sending of an e-mail. Under special circumstances it may also be necessary to return unique identification numbers for the data items processed on the server. These numbers represent groupware documents and are referred to as *note ID* or *universal ID* in Lotus (N.B. this is not the same as the identifier of the data stream). An example for the use of such an ID is a user utterance such as "please move the appointment to 7 pm," which leads to the modification of an entry in the calendar that was created earlier during a SMARTKOM session but is still held in the system. The second kind of request is characterized by the fact that it results in the generation of data output on top of the metainformation. The retrieval of contacts from the address book or entries from a personal calendar are requests of that type. These searches are modeled much like an SQL query: restrictions (the WHERE-clause) such as the last name of a person, and the desired answer format (the SELECT-clause) for the resulting data have to be specified. A typical request for example would be "I need John Doe's telephone number" (for the M3L code of this request, see the next section). For the calendar it is also possible to ask for appointments that take place during a certain period of time. This supports the processing of speech input such as "show me all my appointments between 4 and 6 pm".

4.3.7 Address Book Search: An Example

The following example shows a valid M3L representation of a request to the standard applications module for the telephone number of *John Doe* by a user that is registered on the Lotus Domino Server with the name *TestUser*. The format for the result of the query in the address book contains the telephone number and the first and last name of the persons found. Please note that for better legibility the header information has been deleted from all example messages in this section.

```
<externalApplicationRequest> \ a request to
<dataStream> \ standard applications
<name> \
abook_1\
</name> \
<purpose> \
queryAddressBook\
</purpose> \
</dataStream> \
<smartkomUserID> \ user information
TestUser\
</smartkomUserID> \
<action> \
<addressBookSearch> \ the type of the request
<personAnswerFormat> \ the desired answer format
<name> \ (SELECT part of query)
<firstName> \
</firstName> \
<lastName> \
```

```
</lastName> \
</name> \
<contact> \
<telephoneNumber> \
</telephoneNumber> \
</contact> \
</personAnswerFormat> \
<searchPerson> \ the restrictions
<name> \ (WHERE part of query)
<firstName> \
John\ look for 'John Doe'
</firstName> \
<lastName> \
Doe\
</lastName> \
</name> \
</searchPerson> \
</addressBookSearch> \
</action> \
</externalApplicationRequest> \
```

The answer containing the data matching the request is the following (the request status information, which is also produced as output, has been omitted for the sake of clarity):

```
<externalApplication> \ data from standard applications
<dataStream> \
<name> \
abook_1\
</name> \
<purpose> \
queryAddressBook\
</purpose> \
</dataStream> \
<addressBook> \ the type of the data returned
<smartkomUserID> \ user information
TestUser\
</smartkomUserID> \
<person> \ all address book entries
<name> \ matching the restrictions
<firstName> \
John\
</firstName> \
<lastName> \
Doe\
</lastName> \
</name> \
```

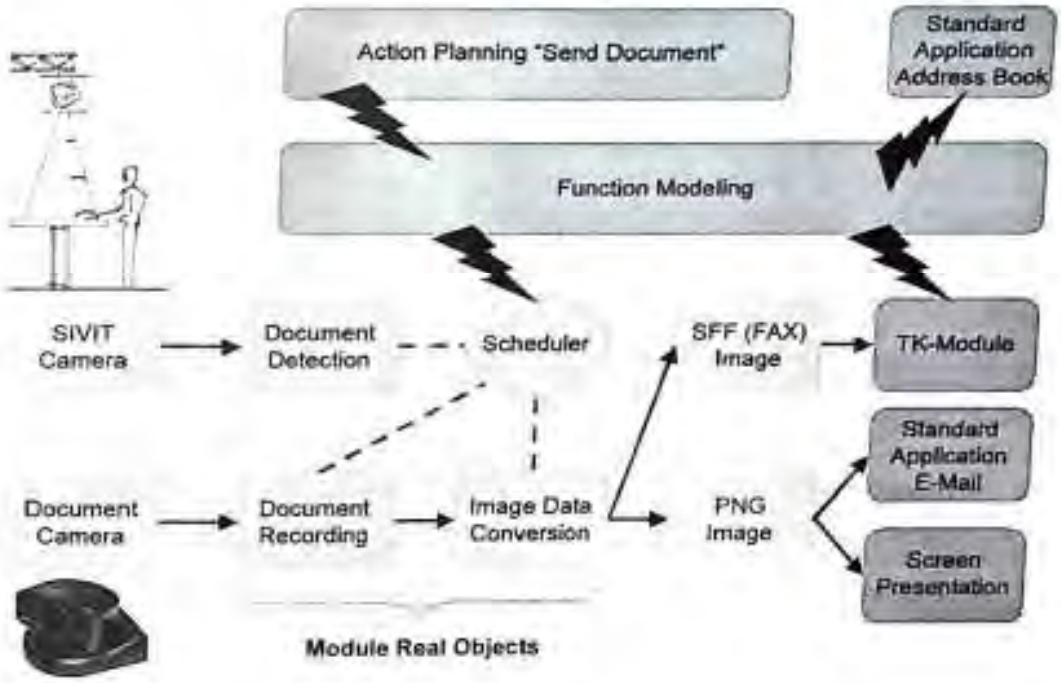


Fig. 7. SMARTKOM-Public fax and e-mail functionalities

```
<contact> \
<telephoneNumber> \
16508451000\
</telephoneNumber> \
</contact> \
</person> \
</addressBook> \
</externalApplication> \
```

4.3.8 Summary: Standard Applications in SmartKom

All standard applications that are available to SMARTKOM users are provided by a middleware that offers an M3L interface and models certain functionalities of a groupware product such as the Lotus Domino suite. Single tasks, like the retrieval of an e-mail address, are small building blocks of plans that are generated based on a multimodal dialogue between the SMARTKOM user and the system.

The standard applications are used to store pieces of information about SMARTKOM users, for example, their preferred method of authentication, whether voice, signature or hand. The different tasks generate a flow of information that is separated into a data and a control flow conforming to the architecture of SMARTKOM. Unique IDs enable back-referencing to data objects already held in the system.

4.4 Composed Applications: Fax and E-Mail

Among the most complex applications in the SMARTKOM system are the fax and e-mail functionalities. A lot of SMARTKOM modules are involved to produce the desired results. Although the internal operating sequence is complex, the user gets the impression of an easy to use functionality. But now we go into the details. Let us start with the previous Fax example user utterance "I'd like to send a fax to Mr. Wahlster." Let us also assume that the action planning module of SMARTKOM already understood the sense of the user's utterance and sent the order "send document to Wahlster fax" to the function modeling module, which coordinates the next steps. What is happening internally?

1. The ISDN fax number of Mr. Wahlster has to be determined.
   (Standard Application Address Book)
2. The Screen presentation "Document scanning" is shown.
   (Module Display)
3. SMARTKOM waits for a document on the interaction area.
   (Module Real Objects analyzes SiViT gesture camera input)
4. The high-resolution image is captured by the document camera.
   (Module Real Objects performs Panoramic Imaging)
5. The image is converted into a display format and rendered on the screen.
   (Module Real Objects and Module Display)
6. The image is converted into a SFF black and white Fax image.
   (Module Real Objects)
7. Image generated in step 7 is sent to the Fax number determined in step 1.
   (TK-Module).

If the user wants to send an e-mail the steps are slightly different: instead of the ISDN fax number the e-mail address is determined, and instead of the black and white image a PNG color image is generated and sent as an attachment to the selected e-mail address.

References

C. Frank, J. Adelhardt, A. Batliner, E. Nöth, R.P. Shi, V. Zeißler, and H. Niemann. The Facial Expression Module, 2006. In this volume.

S. Grashey and M. Schuster. Multiple Biometrics, 2006. In this volume.

C. Maggioni and H. Röttger. Virtual Touchscreen – A Novel User Interface Made of Light — Principles, Metaphors and Experiences. In: *Proc. 8th Intl. Conf. on Human-Computer Interaction*, pp. 301–305, Munich, Germany, 1999.

J. Racky, M. Lützeler, and H. Röttger. The Sense of Vision: Gestures and Real Objects, 2006. In this volume.

W. Wahlster, N. Reithinger, and A. Blocher. SmartKom: Multimodal Communication with a Life-like Character. In: *Proc. EUROSPEECH-01*, vol. 3, pp. 1547–1550, Aalborg, Denmark, September 2001.

SmartKom-Home: The Interface to Home Entertainment

Thomas Portele[1], Silke Goronzy[2], Martin Emele[2], Andreas Kellner[1], Sunna Torge[2], and Jürgen te Vrugt[1]

[1] Philips Research Laboratories GmbH, Aachen, Germany
{Thomas.Portele,Andreas.Kellner,Juergen.te.Vrugt}@philips.com

[2] Sony International (Europe) GmbH
Sony Corporate Laboratories Europe, Advanced Software Laboratory, Stuttgart, Germany
{goronzy,emele,torge}@sony.de

Summary. SMARTKOM-Home demonstrates the use and benefit of an intelligent multimodal interface when controlling entertainment devices like a TV, a recorder, and a jukebox, and when accessing entertainment services like an electronic program guide combining speech and a handheld display with touch input. One important point is emphasizing the functional aspect, i.e., the user's needs, conveyed to the system in a natural way by speech and gesture, are satisfied. The user does not need to know device-specific features or service idiosyncrasies. The function modeling component in SMARTKOM-Home has the necessary knowledge to transform the abstract user request into device commands and service queries.

1 Introduction

The central goal of the SMARTKOM-Home scenario is to support intuitive access to home entertainment functionality using the SMARTKOM paradigms. The SMARTKOM-Home system was jointly defined and monitored by Philips Research in Aachen and Sony International (Europe) GmbH in Stuttgart.

We start with a description of the features of the SMARTKOM-Home system in Sect. 2. The hardware used is described in more detail in Sect. 2.1; the applications are discussed in Sect. 2.2. Hardware and applications pose some particular constraints on the dialogue system. On one hand, devices the user usually uses at home such as TV, video cassette recorder (VCR), MP3 jukebox, etc., need to be controlled by SMARTKOM. As a consequence, the many functionalities of these devices and services need to be modeled within the system to be able to control them and offer their full functionality to the user. In our system this is handled by the function modeling component, which is described in Sect. 2.3. Furthermore, services such as an electronic program guide (EPG) are characterized by their highly dynamic content, i.e., the daily changing TV program and information for current shows. In order to handle dynamic content appropriately, various modules in the system need

User	Hello!
System	Welcome to SMARTKOM. My name is Smartakus. How may I help you?
User	What's on TV this noon?
System	This is the TV program at noon.

Fig. 1. Typical interaction with SMARTKOM-Home (translated from German), part 1. After initial greeting, the results of the electronic program guide are shown

to be aware of it. Therefore, a dynamic lexicon, described in Sect. 2.4 serves as a central knowledge source. Also, the interaction modes are different from the other scenarios. In the home scenario we explicitly distinguish between lean-forward and lean-backward modes (Sect. 2.5). Feedback strategies as an important part of the interaction are also discussed. In Section 3 we mention the evaluations that were conducted to evaluate the SMARTKOM-Home demonstrator.

2 Features

The environment of a SMARTKOM-Home system is, not surprisingly, the living room of the user. The global application is accessing entertainment functionality. Key features are multimodal interaction using speech and a tablet PC, switching between a multimodal lean-forward and a speech-only lean-back mode, and an animated inter-

User What else is on this channel (points to a program from provider ARD)?
System This is the TV program on ARD in the afternoon (shows only program items from ARD)
User Which genres do you know?
System These are the genres you can use.

Fig. 2. Typical interaction with SMARTKOM-Home (translated from German), part 2. Dynamic help delivering information about the genres

action agent called "Smartakus". One important point is emphasizing the functional aspect. The user does not need to know device-specific features or service idiosyncrasies. Figures 1–3 show a typical interaction (translated from German). After an initial greeting, a TV show is selected using dynamic help. The video recorder is programmed to record that show, and the TV is switched on to a currently running program. The system is set to lean-back mode while the user watches TV.

2.1 Hardware

The application hardware is a high-end digital Sony VCR DHR-1000VC and a standard Philips TV set. This represents a sample set of devices. However, the layout of the modules that control these devices is designed to support other devices as well, such as hi-fi components, etc. In choosing the interaction device we had to consider that it should reflect the necessities of the home environment. A platform that is

User	Is there sports on any channel?
System	These are the sports programs this afternoon (shows all sports program items)
User	Record that (points to a tennis show)
System	The show "US Open Semifinals" today at eight p.m. on RTL will be recorded
User	Switch the TV to ARD
System	(switches the TV on and selects channel ARD). The TV is now showing the station ARD.
User	Turn off the display
System	(turns display off) The display is now deactivated.

Fig. 3. Typical interaction with SMARTKOM-Home (translated from German), part 3. A TV show is selected. The video recorder is programmed to record that show, and the TV is switched on to a currently running program. The system is set to lean-back mode while the user watches TV.

mounted at a fixed place in the room will not be accepted by people used to remote controls, but the constraints of a mobile scenario on device size do not apply either. A portable easy-to-use device with a sufficiently large interaction display to allow showing several information items (EPG info and device status) and the SMARTKOM persona Smartakus simultaneously was judged to be the optimal interaction device. At the time of the definition project phase (early 2000), the Fujitsu Stylistic 3500 tablet PC (a full-fledged PC with a touch screen, see Fig. 4) was the best choice

— nowadays several similar Web pads are available. The device is sufficiently powerful to run the SMARTKOM modules for audio I/O and display generation locally.

2.2 Applications

The main applications are the control of consumer electronics (CE) devices and access to an EPG. Furthermore, the personal calendar application is used to store and retrieve shows to be recorded. For each application a description for the SMARTKOM action planning module was created. The action planning module is responsible for coordinating/triggering the different modules when necessary. Links between these descriptions allow switching from one application description (also termed *action plan*) to another, e.g., if the user indicates in the EPG application that she wants to record a certain program with the VCR. This process is transparent for the user, but facilitates development and integration of new applications. An important difference between device control and EPG is the way of interaction. EPG is performed in a *browsing mode*, where information is always displayed, and missing information is substituted by meaningful default values (e.g., "today, now" for time, "all" for channels and genres). These values are changed and refined during the browsing process (Kellner and Portele, 2002). On the other hand, device control needs to have all values determined before any action can be carried out. This is in line with traditional mixed-initiative slot-filling dialogues (Aust et al., 1995). The SMARTKOM backbone supports both styles of interaction. Each application has a visual representation (see screenshots in Figs. 1–3). The EPG application usually takes the full screen (Figs. 1 and 3), while TV and VCR information are displayed in the bottom (see Fig. 3). When delivering user assistance in a metadialogue, the screen changes again (Fig. 2). The EPG application supports preferences for genres and channels. These preferences, however, are not used as a filter to eliminate hypothetically unwanted information, but help to resort the results — preferred items are displayed (or spoken in lean-back mode) first. The preference values are extracted by the dynamic help component (Streit, 2003) according to emotive user reactions. For instance, if information about a thriller is displayed and the user likes it (she says something like "fine" or smiles), the preference value for the genre "thriller" is increased.

2.3 The Function Model

Since we consider SMARTKOM-Home as a uniform interface to various services and devices, all of these are modeled by the function model. This was motivated by the fact that users will have complex wishes involving several devices and/or services. However, the user should not have to care about which device needs to do what at a certain point in time. Therefore the function model formally describes the functionalities of all services and devices in the system. The action planner triggers the requests to the function model. A planning component within the model generates a plan that includes all devices/services needed to fulfill the user wish. Furthermore, this plan includes the necessary steps to be taken for each of the applications. A user wish could, e.g., be "Record the thriller with Julia Roberts tonight". This involves

querying the EPG services for the information needed (in this case channel and starting time) to then program the VCR accordingly. The function model is described in more detail in Torge et al. (2002a,b).

2.4 The Dynamic Lexicon

Most of the applications in the home scenario are characterized by their highly dynamic content. This holds in particular for the EPG, since the TV program changes daily or even more frequently. Many modules are affected by this dynamic content. The dynamic lexicon handles this dynamic content and serves as a knowledge source for those modules that need lexical information. Concretely, these are the speech recognizer, the prosody recognizer, the speech analysis, and the speech synthesis for obvious reasons: Of course the user should be able to speak any name of a show, actor etc., genre, time channel and so on when browsing the EPG. These new words must be recognizable and interpretable, and also the speech synthesis on the output side uses the new words when presenting the EPG search results. It is the task of the lexicon to keep track of words that need to be added/removed to/from the lexicon when the user request requires this. Words can also be removed from the lexicon when not needed any longer. This is important for keeping the speech recognizer's vocabulary as small as possible to maintain high speech recognition rates.

Also, the pronunciations for new words are necessary for the modules named above. These are automatically generated by a decision tree–based grapheme-to-phoneme conversion. Pronunciations are mainly generated for German. However, some lexical entries, such as movie titles or actor names, are often not German but rather English. As a consequence, English pronunciations are generated simultaneously for those entries.

2.5 Interaction

The interaction with SMARTKOM-Home is governed by the principle that the user should have control over the system. The system is designed more like a butler than like an autonomous agent. The user must be able to understand *what* the system is doing, and *why* it is doing it, and it is one important task of the system to support this.

The home scenario with the tablet PC demands some specific interaction features. Most important, users should be able to switch between two modes:

- The *lean-forward* mode supports using the display for touch input and visual output. This is the "normal" mode for focused interaction with the system.
- The *lean-back* mode uses only the acoustic channel (i.e., speech) for input and output. This mode is assumed to be used when the user is either not willing to leave the sofa and reach for the device, or when only short commands like channel switching are necessary. An example corresponding to the second system turn of Fig. 1 would be read as: *"Some of the broadcasted programs are the news show "Tagesschau" on ARD at 12:30, the sitcom "Alles unter einem Dach" at*

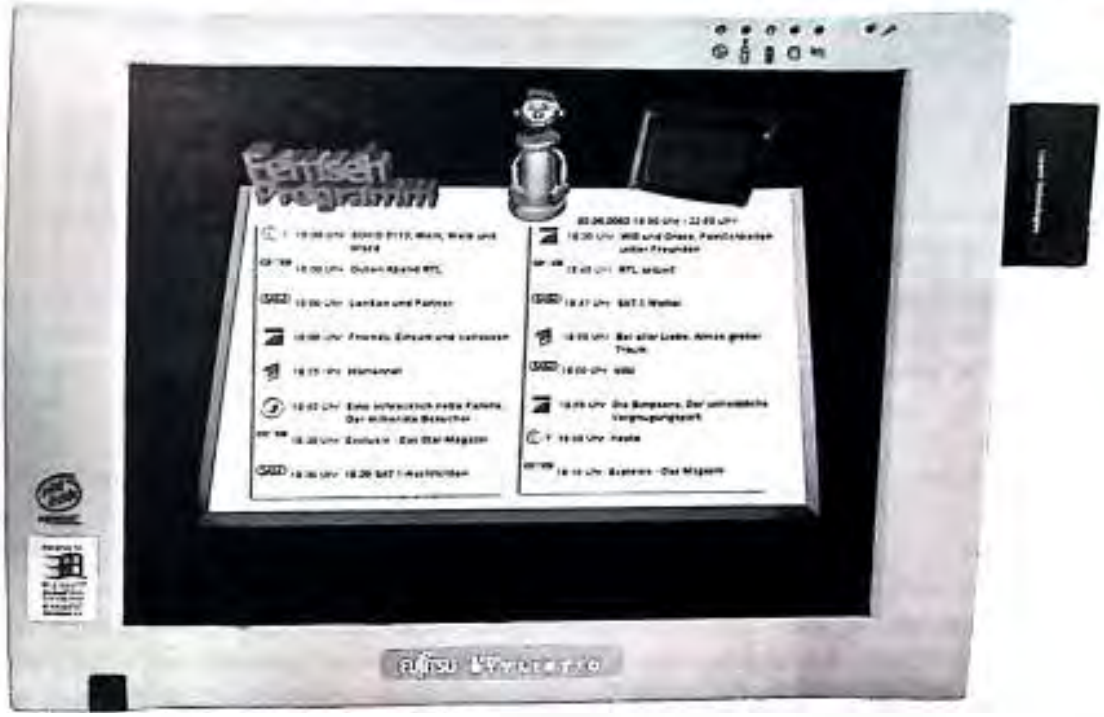


Fig. 4. Tablet PC as interaction device of SMARTKOM-Home

12:30, and the news show "Tagesschau" on ZDF, also at 12:30. Only a limited number of item (currently three) is read.

Switching between the modes must be initiated by the user. The switch to lean-back mode is given via one of several speech commands (e.g., *"Go to sleep!"*). The lean-forward mode is turned on either by a speech command (e.g., *"Wake up!"*) or by touching the display. The system can suggest a mode switch to lean-forward mode if a list with more than three items has to be conveyed to the user.

Feedback is an important issue in a complex dialogue system (Brennan and Hulteen, 1995). Three levels of feedback are distinguished in SMARTKOM-Home, and each level has its own modality in the lean-forward mode:

- **System State:** The state of the system is conveyed by the Smartakus persona. When a user command can be accepted, it makes a listening gesture. Processing stages (recognition, analysis, processing, presentation) are symbolized by different persona gestures. Thus, the user knows when the system is working (and even how long it may take), and when it is ready to accept commands.
- **System Belief:** The results of the analysis are presented by speech, e.g., EPG constraints ("Here you see this evening's sports shows."), or device states ("The TV is now switched to ARD."). Thus, the user can detect misinterpretations. The visual presentation also contains information about the system belief, e.g., the current constraints in the EPG application.

- **System Answer:** The results of the user query (EPG information, device states) are displayed on the screen.

In the lean-back mode, only spoken feedback is possible. The system state is not conveyed to the user, but the other two items are (at least partially for result lists) spoken.

Another issue for complex systems is user assistance. Three cases are considered in SMARTKOM-Home:

- **Help:** The user can ask for help. Supported questions contain formulations like "What can I say?" for general help, "Which channels/genres/... do you know?" for the possible EPG constraints, and "Which commands do you know?" for the device control. The possible alternatives are then displayed (see central screenshot in Fig. 1).
- **Problem detection:** The system can detect problems during the interaction. These problems can be caused by failure to recognize or interpret user input. The system then gives assistance by describing current input possibilities (similar to the response to "What can I say?"), or by asking for missing information.
- **Constraining and Relaxing:** The EPG database access can lead to no or too many results. The system then displays the available information and gives hints to constrain or relax the query — it will not do that automatically, because the user should remain in control. The same holds for failures during VCR operation (e.g., missing recording media).

Modality-switching, feedback, and user assistance are designed according to the butler metaphor: the user has control, the system gives assistance.

3 Integration and Evaluation

The complete system contains 28 modules communicating via a multiblackboard architecture (Klüter et al., 2000). The interface format is an XML dialect which allows using online validation and several XML tools. The modules were developed by ten partners and span the whole range from device handlers for audio and display, recognizers for gestures and speech, analyzers, dialogue, function, lexicon and presentation managers, service and hardware interface modules, to generation and synthesis modules. The complexity of the system is therefore substantial, and intensive testing and checking against the specifications was performed by the integration group at the DFKI and the scenario managers (Philips and Sony). To avoid a mixture of incompatible module versions, fixed integration dates were defined by the system group, and several versions of SMARTKOM-Home existed until the final version was rolled out.

3.1 Informal Evaluation of Prerelease Systems

In the initial phase of the project, the SMARTKOM-Home scenario was specified by the scenario managers. Among other things, the scenario specification included:

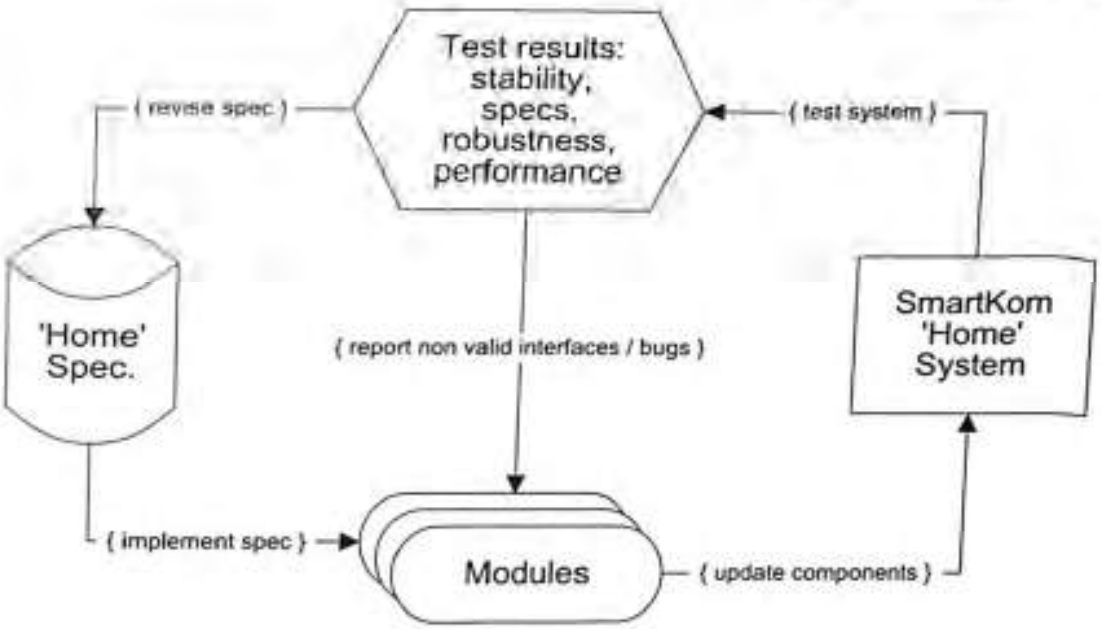


Fig. 5. Workflow of informal evaluations of prerelease SMARTKOM-Home systems

- **Interaction patterns:** These patterns define the modalities that can be used to interact with the system in various situations. Basic patterns that describe the typical task-specific dialogue behavior have been formulated. For example, for the user input speech and pointing gestures are used to interact with the system in the lean-forward mode. In the lean-back mode, only speech is enabled as input modality. In this mode, audio is the only modality used for system output, while in lean-forward mode the screen of the portable touch-screen device is the main output device.
- **Applications:** A number of applications to be controlled by the SMARTKOM-Home system have been identified. The unified control of different applications by SMARTKOM supports accessing functionality enabled by the interplay of two or more applications in an easy and intuitive way.

Besides the abstract specification, examples of the interactions between user and system have been provided. These contained formulations for input and output and sketches for the screen output. These examples have been used by the module designers to create first versions of their modules and to generate the required knowledge sources (e.g., the grammar templates).

To build a system for end-user tests, the workflow depicted in Fig. 5 was applied by the sites responsible for the SMARTKOM-Home scenario.

Starting with an initial SMARTKOM-Home system, this system was tested by developers according to the scenario specifications regarding possible interaction patterns and applications. These interactions were recorded, and the results were evaluated against the scenario specification. The robustness of the interaction was also

tested: The interaction examples provided by the scenario managers and enriched by the system group defined a minimal baseline the system must handle. To obtain a robust system, alternative approaches to solve a task were also investigated. Other issues during the test were the system stability and performance in terms of processing speed.

After a test phase, the specification of the scenario was revised. The updated scenario description was delivered to the project partners to adapt the modules. Interface format problems and obvious bugs were directly reported to the responsible partners.

The system group integrated components that had been updated (i.e., modules or knowledge sources) due to the system tests or due to feature updates into a new prerelease system. Based on this new system, the tests started again.

The main difficulties in this workflow arose from the complexity and the distribution of the development over several sites. The complexity stems from the large number of modules that had to work together and the large set of functionalities specified for the system. The distributed development lead to a significant delay in reactions on problem reports and communication overhead.

The inherently dynamic nature of a dialogue system made it inevitable to perform online tests. Subsets of the complete systems have been tested in offline tests to ensure stability of the modules. Continuous testing and checking of new versions against a given specification proved to be one of the key factors for the success of the project.

3.2 Evaluation Using the PROMISE Framework

A formal evaluation of an earlier version (SMARTKOM-Home 2.1) had been carried out by project partners from the university of Munich using their PROMISE framework (Beringer et al., 2002). The evaluation established several problematic aspects such as unacceptable system response times that could be resolved by module optimization and new hardware. Furthermore, the evaluation resulted in an improved feedback mechanism and optimized module behavior. The current prototype has also been subject to a similar evaluation, and an improved performance has been observed.

4 Conclusion

SMARTKOM-Home is a multimodal assistant for home environments. SMARTKOM-Home serves as a uniform interface to all service and devices by combining applications for EPG and device control in a transparent way. The interaction style adheres to the principle: the user has control, the system gives assistance. We described particular requirements for this scenario and how the involved modules take this into account. This included a description of the hardware, the function model that jointly controls all services and devices, and the dynamic lexicon, which serves as central knowledge source for all modules that need to deal with dynamic content such as

the entries in the EPG data. Also the interaction modes — lean-forward and lean-backward mode — that are specific for the home scenario were discussed together with strategies for feedback and user assistance. The system was iteratively improved by conducting user evaluations, which confirmed the importance of feedback and user assistance.

References

H. Aust, M. Oerder, F. Seide, and V. Steinbiss. The Philips Automatic Train Timetable Information System. *Speech Communication*, 17:249–262, 1995.

N. Beringer, U. Kartal, K. Louka, F. Schiel, and U. Türk. PROMISE: A Procedure for Multimodal Interactive System Evaluation. In: *Proc. Workshop "Multimodal Resources and Multimodal Systems Evaluation"*, pp. 77–80, Las Palmas, Spain, 2002.

E. Brennan and E.A. Hulteen. Interaction and Feedback in a Spoken Language System: A Theoretical Framework. *Knowledge-Based Systems*, 8:143–151, 1995.

A. Kellner and T. Portele. Spice - A Aultimodal Conversational User Interface to an Electronic Program Guide. In: *Proc. ISCA Tutorial and Research Workshop on Multi-Modal Dialogue in Mobile Environments*, Kolster Irsee, Germany, 2002.

A. Klüter, A. Ndiaye, and H. Kirchmann. Verbmobil From a Software Engineering Point of View: System Design and Software Integration. In: W. Wahlster (ed.), *Verbmobil: Foundations of Speech-to-Speech Translation*, pp. 635–658. Berlin Heidelberg New York, 2000. Springer.

M. Streit. Dynamic Help in the SmartKom System. In: *Proc. ISCA Workshop on Error Handling in Spoken Dialogue Systems 2003*, Chateau-d'Oex, Switzerland, 2003.

S. Torge, S. Rapp, and R. Kompe. The Planning Component of an Intelligent Human Machine Interface in Changing Environments. In: *Proc. Workshop on Multi-Modal Dialogue in Mobile Environments*, Bad Irsee, Germany, 2002a.

S. Torge, S. Rapp, and R. Kompe. Serving Complex User Wishes With an Enhanced Spoken Dialogue System. In: *Proc. Int. Conf. on Speech and Language Processing*, Denver, Colorado, 2002b.

SmartKom-Mobile: Intelligent Interaction with a Mobile System

Rainer Malaka, Jochen Häußler, Hidir Aras, Matthias Merdes, Dennis Pfisterer, Matthias Jöst, and Robert Porzel

European Media Laboratory GmbH, Heidelberg, Germany
{rainer.malaka, jochen.haeussler, hidir.aras, matthias.merdes, dennis.pfisterer, matthias.joest, robert.porzel}@eml.villa-bosch.de

Summary. This paper presents SMARTKOM-Mobile, the mobile version of the SMARTKOM system. SMARTKOM-Mobile brings together highly advanced user interaction and mobile computing in a novel way and allows for ubiquitous access to multidomain information. SMARTKOM-Mobile is device-independent and realizes multimodal interaction in cars and on mobile devices such as PDAs. With its siblings, SMARTKOM-Home and SMARTKOM-Public, it provides intelligent user interfaces for an extremely broad range of scenarios and environments.

1 Introduction

Novel ensembles of services and interaction paradigms have become available for stationary systems and applications. As a result of higher bandwidths for mobile networks, increasing computing power in mobile devices, and improvements in display and I/O technologies, it is now possible to make these services and interaction paradigms available on mobile systems. The need to adapt to tiny displays or anachronistic protocols like WAP is rapidly diminishing (Rapp et al., 2000), thereby causing a transition from thin clients toward thick(er) clients. Interaction via speech input can be performed with high-quality audio streams instead of GSM-based ones (Haiber et al., 2000). To some extent acoustic as well as graphical rendering is possible on the mobile device itself or can be executed on a server, and the ensuing results can be streamed from the server to the client. Some of the most advanced techniques of human–computer interaction can now be ported to the mobile world, including natural language interaction, multimodal interfaces, embodied conversational agents, semantic Web integration, and more (Console et al., 2002; Huang et al., 2000; Malaka, 2000; Malaka and Zipf, 2000).

Many other research systems fail in one of the aspects outlined above. They are either too narrowly limited to some "toy world" applications (Boualem et al., 2002), or they have rather limited user interface capabilities (Gurevych et al., 2003a). SMARTKOM, however, features a complex system with free speech, multiple domains, symmetric multimodality, and a broad range of realistic tasks.

In general, mobile systems and their prototypes can be classified in terms of their realization of one or more of the aforementioned levels of complexity. For example, we find individual prototypes and systems that allow for very sophisticated multimodal input, such as the MATCH system (Johnston et al., 2002) or the QuickSet system (Oviatt, 1996), which, however, are either limited to a single end-device or a single application domain.

Others such as described by Krüger et al. (2004) enable seamless device changes and connectivity but, in turn, are lacking interaction and service flexibility. In this paper, we describe how—for the first time—a system was developed in which all levels of complexity are fused into an operational prototype.

We will show how such a system was adapted and customized from the core SMARTKOM system for mobile use, as sketched in Fig. 1. This task is by no means trivial, as we want to keep the full flexibility along all the aforementioned levels of complexity and also support mobility-specific aspects in a scalable and intuitive way. In particular, we show how the processing of geographical information and mobile situation-dependent interaction was integrated in realizing the SMARTKOM-Mobile system.

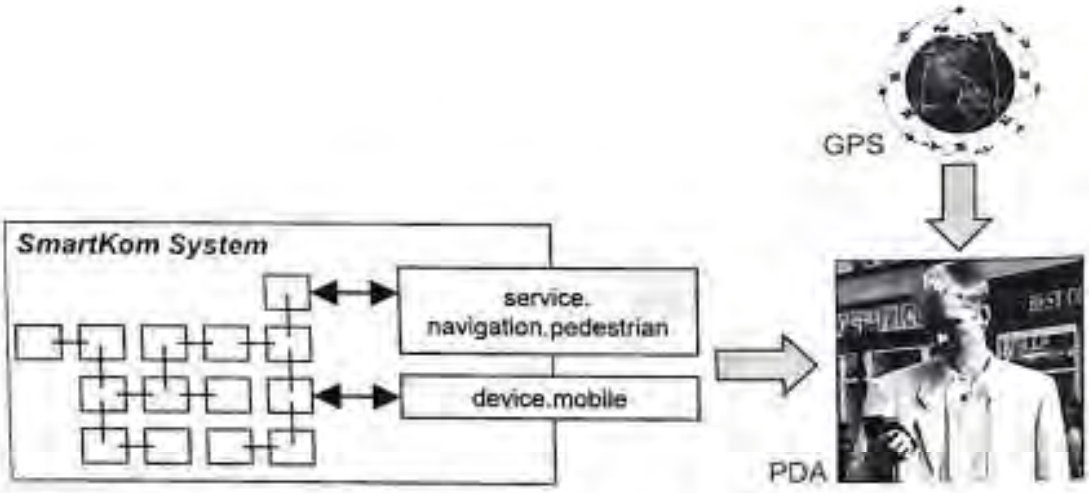


Fig. 1. General architecture layout of SMARTKOM-Mobile. The SMARTKOM system is connected through two modules to the mobile infrastructure: "service.navigation.pedestrian" and "device.mobile". Localization information comes from a GPS module

However, in this fusion of technologies a multitude of mobility-specific aspects remain to be solved. In particular mobile applications, such as route planning or spatial directions, are inherently different from desktop applications. Mobile architectures must be suited for the various user interface components involved and must support multiple end-devices. For mobile systems context changes more dynamically and dramatically than for desktop ones and thus plays a key role. An intelligent mobile interface must therefore take the user's context into account. This context can vary from driving or walking to performing several tasks concurrently.

This paper introduces SMARTKOM-Mobile. It is the mobile version of the SMARTKOM system (Wahlster et al., 2001), a novel multimodal and multidomain dialogue system. SMARTKOM-Mobile enables the user to interact with the system while driving a car as well as while walking around, the pedestrian case, using different mobile devices, interfaces, and modalities. In one case, the built-in screen of the navigation system and the audio system of the car serve as SMARTKOM end-devices, while in the pedestrian case the user cooperates with SMARTKOM through a handheld computer. We show how such a complex interactive system can be adapted to create a *full-fledged* mobile system. For comparing mobile systems, we propose at least three distinct dimensions of complexity:

1. single system and device *versus* heterogeneous and seamless system/device integration
2. single static application *versus* multiple internal and external service integration
3. single-mode, reactive, and controlled interaction *versus* intuitive, proactive, and conversational multimodal interaction.

A system that does not fall short along any of these dimensions of complexity can, in our opinion, be considered a full-fledged mobile system.

In this paper we will introduce the ontology-driven interactions with and within SMARTKOM-Mobile, the mobile services, and the mobile system architecture. Thereafter, we will describe how the system is implemented and present first evaluation results.

2 Mobile Semantic Integration

One of the key factors in realizing a robust multidomain system is the use of an extensible ontology that consists of a domain-independent top-level and process ontology that is combined with multiple specialized domain ontologies. These domain ontologies capture the domain-specific knowledge necessary for representing the domains covered by the individual SMARTKOM scenarios. This ontology is used in two ways:

1. for inference tasks involved in understanding, analyzing, and responding to user requests (Gurevych et al., 2003c)
2. for generating interface specifications for all the SMARTKOM core components (Gurevych et al., 2003b).

This approach allows a consistent extensible and unified knowledge repository. In contrast to many other systems, it comes with an advanced interface technology together with a broad range of application tasks.

For the integration of multidomain services and the mobile interaction described above into a multimodal dialogue system, it is necessary to represent all information and tasks in an abstract and amodal way. For this, a system-wide ontology is used to represent information on all respective domains (Gurevych et al., 2003c). The ontology is organized in a top-level ontology that follows the modeling principles

outlined in Rößler et al. (2001), a domain-independent process model that follows the frame semantic analyses described in Baker et al. (1998), and a number of subontologies for the domain- and task-specific elements. The resulting joint ontology uses domain- and task-independent concepts for general top-level concepts such as *types* or *roles* and processes such as *MotionDirectedTransliterated* for representing both the user's wish to be guided to a specific place as well as for the system's incremental responses. The subontologies refine top-level concepts down to route segments, spatial relations, electronic devices, or the Heidelberg Castle.

The ontology is used for a number of tasks within the system, e.g.:

1. Scoring coherence of speech recognition hypothesis (Engel, 2002): The ontology allows for the identification of the most coherent speech recognition hypothesis, thus increasing recognition reliability even under adverse acoustic conditions.
2. Multimodal fusion (Wahlster, 2002): Ontological knowledge is the basis for multimodal disambiguation, e.g., in the case of pointing gestures on a displayed map, the ontology-driven representation of the accompanying speech input helps to identify the intent of the pen input.
3. Basis for interface specification (Herzog et al., 2003): The interface specification for all multimodal processing modules is formulated on top of the amodal representation of relevant objects, processes, and categories.
4. Speech interpretation (Engel, 2002): For interpreting and intentional analysis of utterances the ontology provides the necessary knowledge base for making inferences and allows for underspecification.
5. Dialogue and discourse management (Pfleger et al., 2002): Complex tasks are also modeled as processes within the ontology. They are then, in turn, used to manage dialogue context.

For modeling the additional domains necessary for the mobile scenario, we added concepts for space, tourism, and navigation to the ontology. This domain-specific ontological modeling was based on the existing ontology of Deep Map and ontologies that have already been proposed for geographical objects (Malaka and Zipf, 2000; Zipf and Krüger, 2001). For the SMARTKOM-specific mobile interactions, no additional processes had to be modeled since user interactions such as navigation requests, tour planning, sight information, or map interaction could already be captured by the domain-independent process model of the SMARTKOM core system.

The semantic bridge from the SMARTKOM world to external agent ontologies in Deep Map or that of Web services supplying topical information is realized in the SMARTKOM module "navigation. pedestrian," which translates between internal and external worlds. The abstract modeling of the mobile SMARTKOM domain allows for a seamless integration of a variety of internal and external services into the SMARTKOM-Mobile system.

As with the other domains Public and Home, language interpretation, multimodal disambiguation, action planning, multimodal presentations, and dialogue management are based on a homogeneous ontological model. Since the ontology is used systemwide, it also allows for mobility-related requests in the other scenarios, i.e., users at home can also plan a route to a city, request information on sights, or interact

with a map. Additionally, the ontology, which is formulated in DAML/OIL, can be used as the basis for message contents in SMARTKOM, and, in fact, the SMARTKOM tool OIL2XSD translates concepts from the ontology into M3L schemata (Gurevych et al., 2003b). This allows for a consistent way of defining message formats that can then be further interpreted and analyzed with the ontology.

3 Mobile Services

In order to build a flexible, user-, and context-adaptive mobile system, one has to take into account a model of the user, the environment, and the interaction between them on the basis of a common ontology. The SMARTKOM-Mobile system realizes such a system by using an agent-based approach that implements intelligent agents that are capable of dealing with positioning, spatial cognition, and contextual information. The agent system incorporates these aspects by using an overall ontology that encodes the environment model as well as the interaction with the user. This way the system tries to react to user requests in an autonomous, independent, and context-aware way (Porzel and Gurevych, 2002), while taking into account the user preferences and certain conditions of a given situation or location.

In SMARTKOM-Mobile, we integrated such a multiagent system for intelligent location-based services for the tourism domain. The Deep Map multiagent system provides a number of location-based services for the city of Heidelberg (Malaka and Zipf, 2000) realized as intelligent agents. Deep Map was developed using a FIPA-based agent platform, Resource-Aware Java Agents (RAJA), that is built on top of FIPA-OS (Ding et al., 2001).

Figure 2 gives an overview of the integration of Deep Map into the SMARTKOM-Mobile system via the navigation.pedestrian module. It also illustrates the interaction between both systems and the involved agents. Both systems have their own respective communication language. In SMARTKOM all content in intercomponent communication is encoded as M3L messages. In Deep Map, messages are encoded in an agent communication language (ACL), where content objects are represented as Deep Map Objects (DMO). Both M3L and DMO are defined by their respective ontologies. The SMARTKOM Deep Map bridge translates between both systems, while the "navigation.pedestrian" control agent (NCA) manages the individual pedestrian navigation sessions. The other agents within the Deep Map system are:

- SpaCE is an agent for a number of spatial cognitive tasks such as interactive knowledge acquisition, localization, route management, and more (Kray and Porzel, 2000). The agent is able to divide a route into segments that are to be verbalized in navigation instructions. This segmentation has to be computed according to features of the route such as landmarks, and turning points, and not just distance (Johnston et al., 2002). SpaCE makes use of a set of low-level services provided by the other agents.
- The GPS agent, for example, is needed by SpaCE to get the current user position or to be informed when the user enters or leaves a certain region or location (region or time triggering).

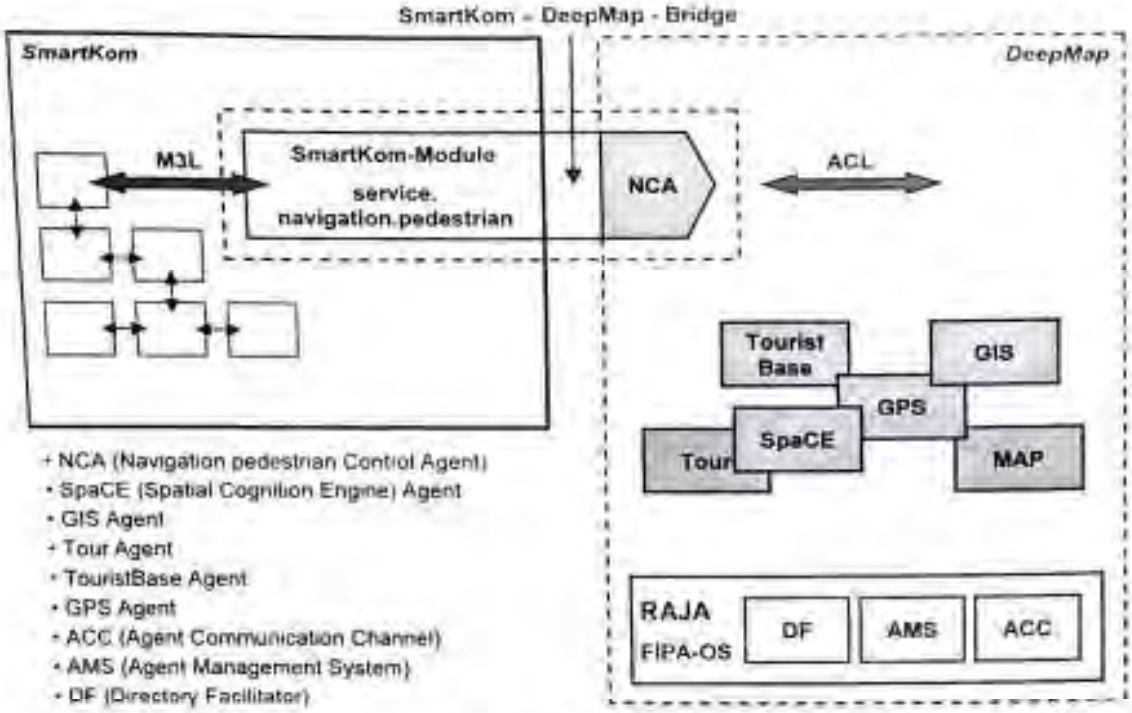


Fig. 2. SMARTKOM Deep Map multiagent system integration. The multiagent system RAJA/FIAP-OS consists of the core components DF, AMS and ACC, which are responsible for agent communication, management, and service discovery, respectively

- The tour agent is a flexible tour planner that optimizes a tour for a given number of sights, time, and other parameters, while calculating user-adapted tours (Joest and Stille, 2002).
- The GIS agent allows access to geospatial information and allows for geospatial querying, distance measurement, and visibility analysis for real-world objects.
- The map agent generates maps for the pedestrian environment and thus allows free map interaction.

Together, these agents constitute a multilayer GIS that contains the topology of the street network, the geometry of buildings, and other spatial objects as well as geo-referenced objects *cum* touristic information that can be subject to user interactions.

As described above, Deep Map provides SMARTKOM with the essential knowledge and reasoning capabilities for realizing mobile and location-based services, which are fused with the SMARTKOM system and services. In the following we will show what kinds of services are realized and how the involved ontologies support the creation of flexible and intelligent mobile user interfaces.

The SMARTKOM-Mobile system differs from the other two scenarios through its support of mobile context-dependent domains and applications. The two main applications in SMARTKOM-Mobile are route planning and spatial navigation. These applications are specific for a mobile environment and are very suitable for developing and testing multimodal interaction.

Since both applications have a spatial context, interactive maps are offered to the user not only to give her an orientation but also as a means to talk about and refer to spatial objects. Therefore a central base application for 'map interaction' was realized (Häußler and Zipf, 2003). The interactions with maps can be divided into:

1. Informative map interactions: The map can be used to identify or select objects and to request information about them. The map itself is not manipulated.
2. Manipulative map interactions: Map interactions with the intention to manipulate the map itself. It can be distinguished between map navigation (pan and zoom), where only the extent of the visible area is changed, and interactions where the content of the map is changed, e.g., by displaying additional objects or hiding others.

Fig. 3. The use of dynamic maps in the application "route planning" *(from left to right)*: Avatar "Smartakus" presents a street map of Heidelberg. After the user has selected the start and destination points for her sightseeing tour, the computed path is presented on the map. For a better orientation, the user may interact with the map again by giving commands like "zoom out"

To make the described map interactions possible, the SMARTKOM system has to know about the content of the currently displayed map. Thus the "no presentation without representation" paradigm (Wahlster et al., 1991) needs to be fully realized for this application. An image as provided by usual map services is insufficient — a semantic description of every generated map is required. We have modeled the description of a map in M3L. The map agent answers every map request from the SMARTKOM system with an image plus semantic description of this map. This description contains general information like map size and geocoordinates of the map's bounding box as well as information about the displayed objects like buildings, etc. Therefore other SMARTKOM modules get all the information that is necessary to

make these objects themselves part of the further dialogue: names, identifiers and types of the objects, their position, and the display style of an object (e.g., green route, blue route). An example for the use of dynamic maps is shown in Fig. 3.

The next application for the SMARTKOM-Mobile system is the pedestrian navigation system with incremental tour guidance support. For route planning in the car, a commercial system was integrated. The pedestrian system differs in many ways. For actual navigational instructions, the route is segmented into "cognitively"-reasonable chunks (Johnston et al., 2002). The GPS can now track the user, and, depending on the relative location of the user to the actual route segment, the system will proactively start interacting with the user. It should be noted that a route instruction is a very unique user interaction within this system. Most tasks in SMARTKOM can be accomplished in only a few dialogue turns and take only a few minutes. A whole navigation or walk through a city can take up to several hours. Moreover, system initiative can be triggered by external/context changes.

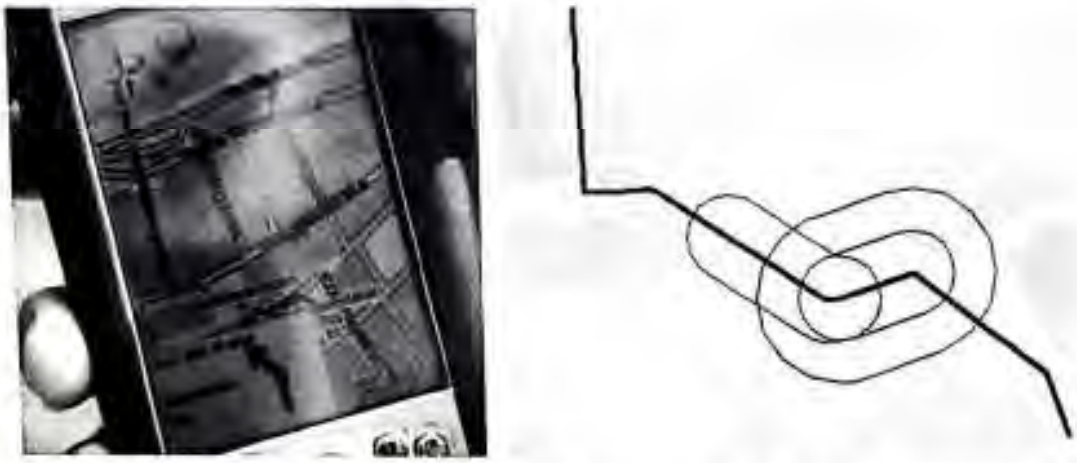


Fig. 4. Route segmentation and instructions. *Left:* SMARTKOM display with dynamically generated map, interface agent, and arrows indicating route direction. *Right:* Segmented route and trigger regions. Two regions around the active route indicate the valid and the egression region. The next route segment's valid region is also shown.

The basic interaction of the navigation system, however, is always related to route segments. The system assumes that the user wants to follow the actual route segment and computes a corridor around this segment which is the "valid region" for the user's location (Fig. 4). As long as the GPS indicates that the user stays within this region, no action is initiated. When the user leaves the valid region of a segment and enters the next valid region of the next segment, the system will request a navigational instruction from Deep Map. The SpaCE agent generates a preverbal message that includes the route segment, landmarks, and necessary user actions such as to turn right or left, for example. While SpaCE controls the spatial context of the user and detects changes concerning the environment, the NCA controls the whole pedestrian navigation session and provides SMARTKOM-Mobile with the needed M3L re-

sponse messages, i.e., maps, preverbal navigation instructions, or textual descriptions of objects, about which the user is informed. The NCA translates between the agent system's ACL representation and SMARTKOM's multimodal markup language.

In the described example, the preverbal message is sent to SMARTKOM and translated into a natural language generation request that, in turn, results in an utterance of the embodied character representing SMARTKOM (Müller et al., 2002). While the SMARTKOM system allows for English and German as interaction languages, the preverbal messages have also been employed for the generation of Japanese instructions in the Deep Map framework (Kray and Porzel, 2000). With this instruction, an update of the map with an arrow indicating the direction of the navigational instruction — relative to the direction from which the user is coming — is generated. Together with the user's position and sights in the vicinity, this is sent to the presentation management components of SMARTKOM.

For each route segment an "egression region" is calculated that surrounds the valid region (Fig. 4). If the user leaves the valid region and enters an egression region, Deep Map proactively notifies SMARTKOM and suggests either a redirection or a recalculation of the tour. If a simple redirection can lead the pedestrian easily back to the route, SMARTKOM will generate an instruction to go back to the precalculated tour. Again, this is done on the basis of preverbal messages from the navigation.pedestrian subsystem. If the user is already too far off the route, a dynamic recalculation of the tour is done and new route instructions lead the user to her goal.

A third proactive behavior occurs when a user enters the vicinity of an interesting sight. SMARTKOM then generates a message that there is a sight and offers to present more information. The user may also decide to request information on objects directly by asking SMARTKOM. This can be done by language requests only, e.g., "Give me more information on St. Peter's Church," or by multimodal interaction using pen input on the map with a pointing gesture and an utterance like, "I want more information on this."

Since all map information is semantically available (and not just rendered as pixel images), it can be used in SMARTKOM and deictic gestures on the display can be resolved (Gurevych et al., 2003c; Häußler and Zipf, 2003). The names of locations labeled on a map are extracted from a generic description of the map by the action planner (dialogue manager) and used for updating the dynamic lexicon (Rapp et al., 2000). This enables the system to recognize and resolve user input with spoken references to locations and to synthesize speech output, including these novel location names. Thus, the mobile SMARTKOM system can make full use of the multimodal processing capabilities of SMARTKOM.

4 Mobile Infrastructure

The mobile SMARTKOM system has to face a number of challenges. They can be classified into those related to resources, those related to services, and those related to interaction. The resource challenges arise as a result of SMARTKOM's complexity.

As described above, more than 40 software components are combined in a multi-blackboard architecture. The prototype system needs at least two high-end PCs for acceptable performance (Herzog et al., 2003; Wahlster, 2003). However, this does not constitute a major restriction for future application of this technology, considering the rapid developments regarding the computing power in mobile devices. Additionally, new wireless broadband networks such as third-generation cellular connections (e.g., UMTS) or wireless local area networks (WLAN) make it possible to distribute software components that are based on such a multiblackboard architecture onto multiple machines and to connect the end-devices through a wireless network.

The goal of the SMARTKOM-Mobile research framework, however, goes beyond porting the system to just one mobile setup and using a mobile terminal instead of a PC screen. The mobile SMARTKOM system should instead implement ubiquitous information access to the SMARTKOM services through any device. The user, in turn, should be able to decide which device she wants to interact with. In our experiments and implementations, we prototypically focused on the information technologies (IT) infrastructure typically found in modern cars — such as the display of a car navigation system, microphone array for free speech input, and audio system — as one setup, and a handheld computer such as a personal digital assistant (PDA) as a second setup. The management of these multiple devices is implemented in a SMARTKOM module called "device.mobile," sketched in Fig. 5 and described below.

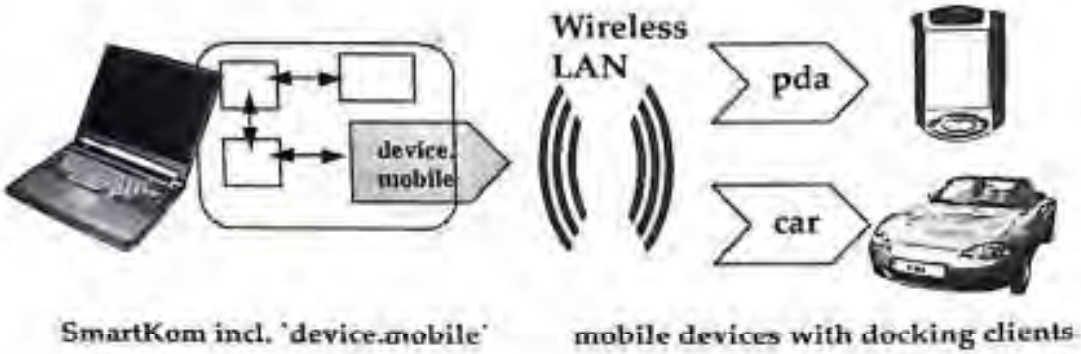


Fig. 5. Setup of the "device.mobile" docking mechanism for SMARTKOM-Mobile

For the integration of end-devices, it is not sufficient to think of one particular device but rather of a personal IT infrastructure owned by the user that can be used in multiple combinations. In order to realize such ubiquitous computing in SMARTKOM, we want to support the user with her own devices. In particular, in situations where multiple devices are available, the user has the final decision regarding which device to use. For instance, in the car a user can either use the built-in screen for visual system output and the microphones for speech input, or she may decide to use her PDA. This could be useful, for instance, because of more convenient peninput with the PDA when the car is parked. It should, however, be noted that it would in principle be possible to serve multiple devices concurrently.

In order to allow maximal flexibility, we realized the device.mobile module as a device broker (Fig. 5). It has two components. One is the SMARTKOM module that resides on the SMARTKOM system. On each potential client, a docking client component establishes the connection to the SMARTKOM system via a WLAN. In principle, it can support any number and type of mobile device. Within the mobile scenario support for four different client devices has been developed, namely PDA (Compaq IPAQ), test and simulation clients, and a client for the onboard computer of a test car. With these inherently different client types it is neccessary to abstract from the specific hardware by a client-specific connection software.

The device.mobile module is designed to support the user in her mobile activities with as little disruption as possible. This includes the seamless transition from one mobile device to another. For example, the user might travel in a car using the built-in display, speakers, and microphone to communicate with the SMARTKOM system in order to have a trip planned. When the user arrives at her destination she can pick up her PDA, leave the car, and the SMARTKOM display will "follow" her to the new mobile device. In order for this to be possible, the physical connections between the SMARTKOM system and the end user device have to be managed and the specific mobile device's capabilities must be communicated to the SMARTKOM system.

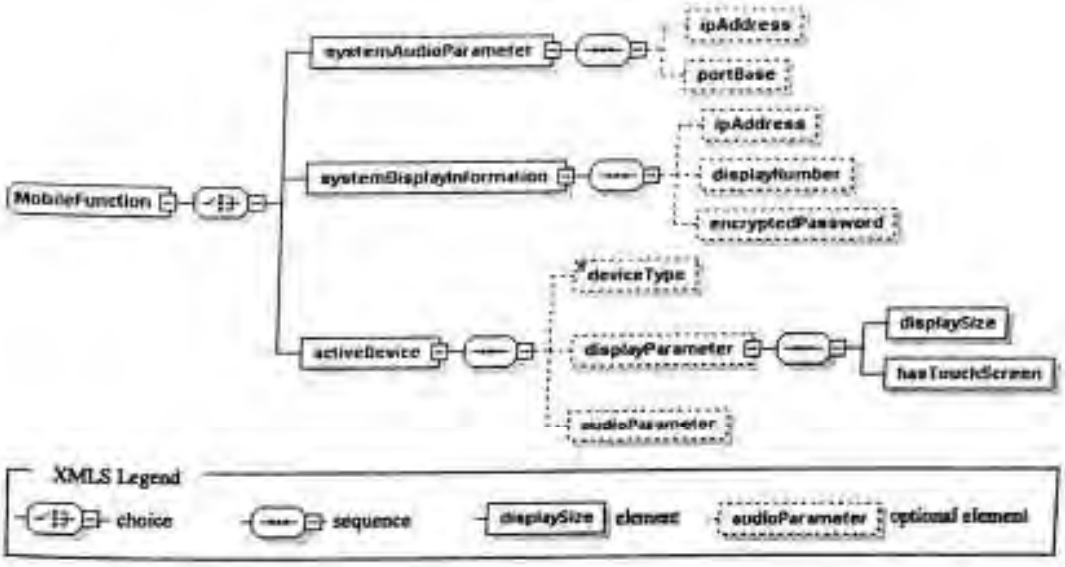


Fig. 6. Model of mobile devices with their specific capabilities expressed in M3L schema

It is therefore necessary to model the display and audio input and output capabilities and the parameters for establishing and maintaining the display and audio connections. As with other domains within the SMARTKOM system, this model is expressed as part of M3L. The relevant portion of the schema is displayed in Fig. 6.

4.1 Modeling Multimodality for Mobile Devices

Apart from the management of physical connections for display and audio data, it is necessary to model the input and output capabilities of the devices on a more general and abstract level. For a multimodal dialogue system such as SMARTKOM, the explicit modeling of supported modalities and their combinations is very important. When the system is aware of the modalities both for input and output of the current active device it can adjust the presentation and its communicative goals to the existing infrastructure. This is especially important since mobile devices can change frequently, requiring constant adaptation of the dialogue to the new situation. Through this mechanism of device abstraction, the system can adapt the output and rendering as well as input and dialogue behavior. This constitutes a necessary step in realizing a framework for a flexible multimodal human–machine interaction in mobile environments (Bühler et al., 2002). Device- and content-dependent interfaces can then be build upon such a setup (Elting et al., 2002).

Whenever the mobile device of the SMARTKOM system changes, the device management device.mobile module checks the multimodal capabilities of the new device and communicates these to the suitable SMARTKOM communication pool in order to enable other modules such as the presentation planning or display modules to use this information. The supported modalities are speech, gesture, mimic, audio, graphics, and persona. Of course, not all modalities are physically identical for input and output devices. For example, the "persona" which represents the life-like character Smartakus is only a virtual entity serving as output modality as compared to the physical entity of the user on the input side. Characteristics of special mobile devices as well as typical usage situations may impose further restrictions on modalities. An example is the situation in a moving car where the user should not be distracted too much and also should not user her hands for pen input.

4.2 User-Driven and System-Driven Activation of Mobile Devices

In order for a multimodal mobile system to be truly flexible it must be able to cope with a number of situations that arise from the special properties of mobile devices and the interaction with these devices. We discovered the following requirements for the handling of mobile devices within the SMARTKOM system:

1. Mobile devices must be discovered automatically.
2. The system must keep track of all mobile devices.
3. If a mobile device gets lost the system must be aware of this.
4. The user should be able to explicitly switch from the currently active device to one of the waiting devices (user-driven activation).
5. If the currently active device becomes unavailable, e.g., due to connection problems, one of the waiting devices should become the new active device (system-driven activation).

The device.mobile module was developed to meet these requirements. The correct implementation of this behavior could be successfully demonstrated. This is espe-

1	Start of SmartKom system	
2	*Registration* of device PDA	
3	Turning on of hardware CAR	
4	*Registration* of device CAR	
5	User-initiated *Activation* of CAR	
6	Loss of active device (connection problem)	
7	System-driven *Activation* of waiting device	

Fig. 7. Typical mobile device scenario with user- and system-driven device change

cially important for supporting the seamless transition between pedestrian and automotive use of the SMARTKOM system. A typical scenario including user- and system-driven change of device is depicted in Fig. 7.

This general setup of the SMARTKOM controls also enables future additions for using multiple devices at the same time and to mix input and output channels, e.g., using audio input from the free speech microphone array in the car while using thr display of a handheld PDA.

5 Implementation and Evaluation

The SMARTKOM-Mobile system has been implemented and tested in a prototypical setting. We realized several variants of the system. For testing with mobile pedestrians in a city, SMARTKOM runs on two laptops. As a handheld device, we used a Compaq iPAQ H3660. The GPS is a Garmin eTrex system. We connected the PDA using WLAN (ad hoc mode) to one laptop running LINUX with two network adapters, one for connecting the Windows laptop and one for the WLAN connection. For the setup in the car, the PCs are located in the trunk. The pedestrian can either take the laptops in a backpack or another person carries them. An alternative is setting up enough WLAN access points in the test area.

Using this configuration, we tested the systems in a number of trials through Heidelberg. In the current configuration, performance is still one of the biggest problems. However, this can be remedied using more powerful high-end computers. Due

to the fact that interaction turns in the pedestrian scenarios are rather infrequent, the performance was generally perceived as sufficient.

The first trials with the system also showed that in noisy environments, the PDA speakers are only usable to a limited extent because the speaker volume is insufficient. However, this problem was remedied by using earphones. In many narrow alleys in the old town of Heidelberg, the positioning using GPS is problematic because of missing contact to a sufficient number of satellites. However, such problems can be fixed using more advanced dead-reckoning algorithms.

In general, the integration phase of SMARTKOM-Mobile showed that it is a big challenge to integrate such a system, and often the AI-loaded components are not the bottleneck but rather technical components such as GPS or WLAN that are not available in all situations. However, we think it is worthwhile to continue the effort of bringing the system into the streets because realistic evaluation environments are needed for further research and development.

As the SMARTKOM system combines the intuitive interaction paradigm of multimodal user input and output with aspects of mobile information systems, it is crucial to evaluate under which user contexts such multimodal interaction is more suitable than existing alternatives. Recently we conducted an evaluation of the SMARTKOM-Mobile system under laboratory conditions in order to address this question (among others).

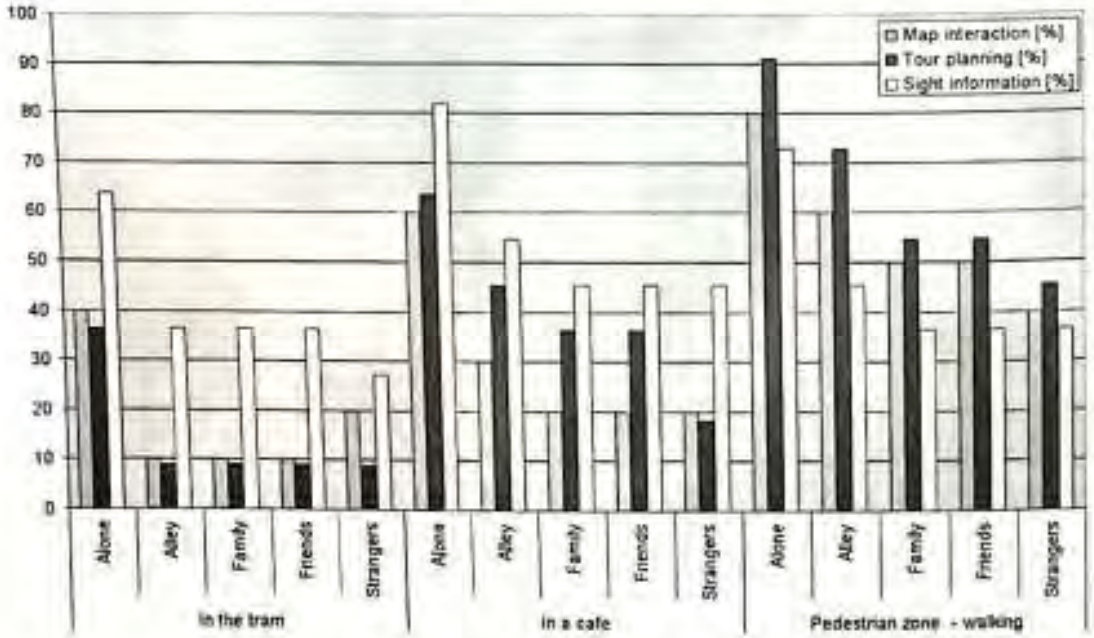


Fig. 8. Rating of multimodal interaction in different tasks. This interaction was compared to the best method a user knows to perform the specific task, e.g., printed map or tour guide

During the study the participants had to perform three different tasks. At first they had to interact with a map of the city of Heidelberg, including panning and

zooming. Afterwards they had to request information about famous sights, and last they had to request a tour from a start location to an end location. These entire tasks had to be performed via multimodal interaction including speech input and pointing gestures. After each task the participants had to name the best method they know to perform the specific task, for example, using a printed map or a Web-mapping solution. Furthermore, they had to compare the performed method with the method they mentioned in different situations, e.g., alone or with friends, walking, or sitting in a cafe.

The results (see Fig. 8) showed that multimodal interaction is rated better than any named method if the user is alone (except for the situation when the user is using public transportation). Also, multimodal interaction is generally more accepted for more complex tasks like tour planning than for easier ones such as map interaction or sight information.

6 Conclusion

In this paper, we presented the mobile SMARTKOM system. SMARTKOM-Mobile represents a new generation of systems for mobile multimodal user interaction, combining three so far disjoint levels of complexity. It realizes these new levels of complexity by integrating many tasks from multiple domains into a single dialogue system and allowing users to interact in a variety of scenarios using heterogeneous devices. Our mobile architecture fuses mobile services and devices together with novel interaction paradigms that enable a user to conduct an uninterrupted multimodal dialogue even while changing devices, e.g., from cars to PDAs.

SMARTKOM-Mobile can be used with different display, audio, and deictic devices. For this, the content is adapted to each situation and device configuration. Therefore, the system needs to adapt to the dynamic mobile situation of the user, e.g., position, topic, activity. Consequently, the integration of a multiagent system was developed in order to deal with location and context-awareness. This system provides the user with flexible location-based services. Through the semantic modeling of geographical content, it is possible that users can also communicate about their spatial environment with the system.

The next steps will be dedicated to user evaluations. The system enables us to evaluate all tasks either in the old town of Heidelberg with mobile users or (feasible for specific questions) in laboratory conditions. In contrast to many other systems, we are now able to evaluate a complex system with real-world data.

Besides those questions relating to human factors of the system, a number of technological questions remain. For instance, network access is still a big bottleneck in public spaces. On the one hand, commercial networks such as UMTS are still not available; on the other hand, private networks like WLAN have become so popular that we encountered "frequency pollution" in some spots.

With the fusion of the described systems, services, and semantics we set the basis for ubiquitous interaction with a personal IT infrastructure. The system allows interactions with potentially arbitrary devices in the user's environment. If we extend this

vision to "ambient intelligence" and multiuser scenarios, the system will need a kind of presentation planning wherein SMARTKOM cooperates with the users through the environment.

The focus of this paper was to present the architecture and concepts of the SMARTKOM-Mobile system. Many aspects could, therefore, not be discussed in detail. One of the biggest advances of SMARTKOM-Mobile is not a single new technology or algorithm but the fusion of a multitude of intelligent user interface methods, services, and devices into a single mobile system. In order to do so, many research questions had to be solved, but also a great deal of engineering had to be performed. The result is a highly complex research prototype that allows for user studies which so far have only been possible in Wizard-of-Oz studies.

References

C.F. Baker, C.J. Fillmore, and J.B. Lowe. The Berkeley FrameNet Project. In: *Proc. COLING-ACL'98*, pp. 86–90, Montreal, Canada, 1998.

M. Boualem, L. Almeida, I. Amdal, N. Beires, L. Boves, E. den Os, P. Filoche, R. Gomes, J. Eikeset Knudsen, K. Kvale, J. Rugelbak, C. Tallec, and N. Warakagoda. Eurescom MUST Project: Multimodal, Multi-lingual Information Services for Small Mobile Terminals. In: *Proc. 9th Conf. of Traitement Automatique des Langues Naturel*, Nancy, France, 2002.

D. Bühler, J. Häußler, S. Krüger, and W. Minker. Flexible Multimodal Human-Machine Interaction in Mobile Environments. In: *ECAI Workshop Notes of AIMS 2002*, Lyon, France, 2002.

L. Console, S. Gioria, I. Lombardi, V. Surano, and I. Torre. Adaptation and Personalization on Board Cars: A Framework and its Application to Tourist Services. In: *Adaptive Hypermedia and Adaptive Web-Based Systems*, LNCS, pp. 112–121, Berlin Heidelberg New York, 2002. Springer.

Y. Ding, C. Kray, R. Malaka, and M. Schillo. RAJA — A Resource-Adaptive Java Agent Infrastructure. In: *Proc. 5th Int. Conf. on Autonomous Agents*, pp. 332–339, Montreal, Canada, 2001.

C. Elting, J. Zwickel, and R. Malaka. Device-Dependant Modality Selection for User-Interfaces — An Empirical Study. In: *Proc. Int. Conf. on Intelligent User Interfaces (IUI) 2003*, Miami, FL, 2002.

R. Engel. SPIN: Language Understanding for Spoken Dialogue Systems Using a Production System Approach. In: *Proc. ICSLP-2002*, pp. 2717–2720, Denver, CO, 2002.

I. Gurevych, R. Malaka, R. Porzel, and H.P. Zorn. Semantic Coherence Scoring Using an Ontology. In: *Proc. Human Language Technology Conference / North American Chapter of the Association for Computational Linguistics Annual Meeting 2003*, pp. 88–95, Edmonton, Canada, 2003a.

I. Gurevych, S. Merten, and R. Porzel. Automatic Creation of Interface Specifications from Ontologies. In: H. Cunningham and J. Patrick (eds.), *Proc. HLT-NAACL*

2003 Workshop on Software Engineering and Architecture of Language Technology Systems (SEALTS), pp. 59–66, Edmonton, Canada, 2003b. Association for Computational Linguistics.

I. Gurevych, R. Porzel, E. Slinko, N. Pfleger, J. Alexandersson, and S. Merten. Less Is More: Using a Single Knowledge Representation in Dialogue Systems. In: *Proc. HLT-NAACL'03 Workshop on Text Meaning*, pp. 14–21, Edmonton, Canada, May 2003c.

U. Haiber, H. Mangold, T. Pfau, P. Regel-Brietzmann, G. Ruske, and V. Schleß. Robust Recognition of Spontaneous Speech. In: W. Wahlster (ed.), *Verbmobil: Foundations of Speech-to-Speech Translation*, pp. 46–62, Berlin Heidelberg New York, 2000. Springer.

J. Häußler and A. Zipf. Multimodale Karteninteraktion zur Navigationsunterstützung für Fußgänger und Autofahrer. In: *Proc. Symposium für Angewandte Geographische Informationstechnologie — AGIT 2003*, Salzburg, Austria, 2003.

G. Herzog, H. Kirchmann, S. Merten, A. Ndiaye, and P. Poller. MULTIPLATFORM Testbed: An Integration Platform for Multimodal Dialog Systems. In: *Proc. HLT-NAACL 2003 Workshop on Software Engineering and Architecture of Language Technology Systems (SEALTS)*, pp. 76–83, Edmonton, Canada, May 2003.

X. Huang, A. Acero, C. Chelba, L. Deng, D. Duchene, J. Goodman, D. Hon, H. Jacoby, L. Jiang, R. Loynd, M. Mahajan, P. Mau, S. Meredith, S. Mughal, S. Neto, M. Plumpe, K. Wang, and Y. Wang. MIPAD: A Next Generation PDA Prototype. In: *Proc. ICSLP-2000*, Beijing, China, 2000.

M. Joest and W. Stille. A User-Aware Tour Proposal Framework Using a Hybr Optimization Approach. In: *Proc. 10th ACM Int. Symposium on Advances in G ographic Information Systems*, pp. 81–88, Washington, DC, 2002.

M. Johnston, S. Bangalore, G. Vasireddy, A. Stent, P. Ehlen, M. Walker, S. Whittaker, and P. Maloor. MATCH: An Architecture for Multimodal Dialogue Systems. In: *Proc. 10th ACM Int. Symposium on Advances in Geographic Information Systems*, pp. 376–383, Washington, DC, 2002.

C. Kray and R. Porzel. Spatial Cognition and Natural Language Interfaces in Mobile Personal Assistants. In: *Proc. Workshop Artificial Intelligence in Mobile Systems at the 14th ECAI (AIMS 2000)*, Berlin, Germany, 2000.

A. Krüger, A. Butz, C. Müller, C. Stahl, R. Wasinger, K. Steinberg, and A. Dirschl. The Connected User Interface: Realizing a Personal Situated Navigation Service. In: *Proc. Int. Conf. on Intelligent User Interfaces (IUI) 2004*, Funchal, Madeira, Portugal, 2004.

R. Malaka. Artificial Intelligence Goes Mobile. In: *Proc. Workshop Artificial Intelligence in Mobile Systems at the 14th ECAI (AIMS 2000)*, pp. 5–6, Berlin, Germany, 2000.

R. Malaka and A. Zipf. Deep Map: Challenging IT Research in the Framework of a Tourist Information System. In: D.R. Fesenmaier, S. Klein, and D. Buhalis (eds.), *Proc. 7th. Int. Conf. on Information and Communication Technologies in Tourism (ENTER 2000)*, pp. 15–27, Berlin Heidelberg New York, 26–28 April 2000. Springer.

J. Müller, P. Poller, and V. Tschernomas. Situated Delegation-Oriented Multimodal Presentation. In: *Proc. AAAI-Workshop on Intelligent Situation-Aware Media and Presentations (ISAMP), AAAI-2002*, Edmonton, Canada, 2002.

S.L. Oviatt. Multimodal Interfaces for Dynamic Interactive Maps. In: ACM (ed.), *Proc. CHI-96*, pp. 95–102, New York, 1996.

N. Pfleger, J. Alexandersson, and T. Becker. Scoring Functions for Overlay and Their Application in Discourse Processing. In: *Proc. KONVENS 2002*, pp. 139–146, Saarbruecken, Germany, September–October 2002.

R. Porzel and I. Gurevych. Towards Context-Adaptive Utterance Interpretation. In: *Proc. the 3rd SIGdial Workshop on Discourse and Dialogue*, pp. 90–95, Philadelphia, PA, 2002.

S. Rapp, S. Torge, S. Goronzy, and R. Kompe. Dynamic Speech Interfaces. In: *Proc. Workshop Artificial Intelligence in Mobile Systems, ECAI 2000*, Berlin, Germany, 2000.

H. Rößler, J. Sienel, W. Wajda, J. Hoffmann, and M. Kostrzewa. Multimodal Interaction for Mobile Environments. In: *Proc. Int. Workshop on Information Presentation and Natural Multimodal Dialogue*, Verona, Italy, 2001.

W. Wahlster. SmartKom: Fusion and Fission of Speech, Gestures, and Facial Expressions. In: *Proc. 1st Int. Workshop on Man-Machine Symbiotic Systems*, pp. 213–225, Kyoto, Japan, 2002.

W. Wahlster. SmartKom: Symmetric Multimodality in an Adaptive and Reusable Dialogue Shell. In: R. Krahl and D. Günther (eds.), *Proc. Human Computer Interaction Status Conference 2003*, pp. 47–62, Berlin, Germany, June 2003. DLR.

W. Wahlster, E. André, B. Bandyopadhyay, W. Graf, and T. Rist. WIP: The Coordinated Generation of Multimodal Presentations From a Common Representation. In: O. Stock, J. Slack, and A. Ortony (eds.), *Computational Theories of Communication and Their Applications*, Berlin Heidelberg New York, 1991. Springer. Also available as Technical Report DFKI-RR-91-08, DFKI, Saarbruecken, Germany.

W. Wahlster, N. Reithinger, and A. Blocher. SmartKom: Multimodal Communication with a Life-like Character. In: *Proc. EUROSPEECH-01*, vol. 3, pp. 1547–1550, Aalborg, Denmark, September 2001.

A. Zipf and S. Krüger. TGML — Extending GML Through Temporal Constructs — A Proposal for a Spatiotemporal Framework in XML. In: *Proc. 9th ACM Int. Symposium on Advances in Geographic Information Systems*, Atlanta, GA, 2001.

SmartKom-Mobile Car: User Interaction with Mobile Services in a Car Environment

André Berton, Dirk Bühler, and Wolfgang Minker

DaimlerChrysler AG, Research and Technology, Ulm, Germany
{andre.berton,Dirk.Buehler,wolfgang.minker}@daimlerchrysler.com

Summary. People tend to spend an increasing amount of time in their cars and therefore desire high comfort, safety, and efficiency in that environment. A large variety of electronic devices has been made available to meet these requirements in the vehicle. These electronic devices should allow for speech interaction in order to minimize driver distraction and to maximize driver comfort.

This contribution studies the user requirements for potential assistant functionalities operated by speech in the car. The architecture is of the dialogue system is defined based on the user requirements study. Speech dialogues were designed according to state-of-the-art principles of human machine interaction for the functionalities desired by the users. Results and ideas for future work conclude this contribution.

1 Introduction

People spend an increasing amount of time in mobile environments, notably in their cars, and want to make that time as enjoyable and productive as possible. Therefore, a large variety of electronic devices has been made available in the vehicle for comfort, ease of driving, entertainment, and communications. All these systems require a rather complex human–machine interaction.

Since driver distraction can become a significant problem, highly efficient human–machine interfaces are required. In order to meet both comfort and safety requirements, new technologies need to be introduced into the car, enabling drivers to interact with onboard systems and services in an easy, risk-free way. Undeniably, spoken language dialogue systems considerably improve safety and user-friendliness of human–machine interfaces (Gärtner et al., 2001; Green, 2000; Minker et al., 2003).

More recently, multimodal recognition and knowledge processing techniques have rapidly grown, enabling their use in mobile environments. A user requirement study has revealed that certain driving assistant functionalities, such as navigation and information services, are highly appreciated by potential mobile users. Multimodal operations may not only be performed in the car environment, but also by pedestrians. A theoretical discussion will analyze the interaction modalities (speech,

gesture, mimics), which are suited for both environments. After a presentation of the conceptual framework, we will discuss the technical realization of the mobile services and propose a well-adapted system architecture. In order to adapt the interaction to human habits and to render it as natural and efficient as possible, we will analyze the most important principles and guidelines in dialogue design. All these definitions, including the different mobile environments, the interaction modalities, the system architecture, and the dialogue design principles, will form the basis for specifying and implementing the services required by the potential users of these multimodal interfaces. The conclusion and outlook section of this chapter summarizes the results and proposes some ideas for future work in this area.

2 User Requirement Studies

A user study on mobile speech-operated services was performed at DaimlerChrysler Research and Technology. It was aimed at defining applications (services) for the SMARTKOM-Mobile environments, which may be of interest for potential users and should be realizable with state-of-the-art technology. Further important issues have been discussed:

1. Which prospective human–machine interaction modalities will enable the use of new approaches to access and process information?
2. Which applications are expected to emerge first on the market?
3. Which new possibilities to exchange and process information may be exploited in the different applications and environments?
4. Which information and interaction modality will be preferred by the potential user?
5. How may the different SMARTKOM applications be combined?
6. Which databases and background intelligence processing are required for a particular application?
7. How may background knowledge be exploited to design interaction modalities that are as natural as possible?
8. How may communication and information processing be efficiently combined?
9. Which modular concepts need to be considered in order to realize the different applications in one generic SMARTKOM device?

The user requirement studies should allow us to combine the effective pursuit of technical concepts with an ingenious strategy of their introduction in the market. Therefore the study focused on three main points: technological trends, business models, and applications. In order to make reliable predictions about emerging technological trends and user behavior (the years 2005–2010 were considered), it is necessary to define target groups. Such groups maybe classified by age/life phase, income, and education. The following groups were identified as potential users for the SMARTKOM services:

1. Early adaptors and fun seekers: young (16-24 years), enthusiastic about technologies

2. High flyers: focused on career and conscious about communication styles
3. Family managers: modern middle class, practically-oriented consumers
4. Conservative, highly educated elderly people

The volume market (late followers), which will, among others, also be composed of these four groups, will develop with a certain delay. Using a user acceptance questionnaire we found that among a large number of services, the following were the most desired ones: mobile phone, emergency calls, dynamic route planning, parking place reservation, and health check.

The most important predictions of this study will be grouped in four main areas: future interaction modalities, financial aspects, data security, and personalization. The findings for each of these areas are summarized in the following.

2.1 Emerging Interaction Modalities (Speech, Gesture, Mimics)

1. Preserving privacy will play an important role in the use of mobile devices. The operation of the devices by speech in public places may be affected by this issue. Mobile phone conversations in the subway are, for instance, outlawed in Japan. Speech input with close microphone seems to be less suited for public use.
2. Cars will be important carriers for speech technology due to safety aspects in operating complex services while driving.
3. The input modalities gesture and mimics are considered science fiction, since a camera observing the driver might lead to acceptance problems.
4. The tolerance of errors will be pretty low in all groups. While input errors using the keyboard are usually considered as personal mistakes, the system will be blamed for input errors using speech operation in most cases. Younger user groups will be more tolerant than older ones.

2.2 Financial Aspects

1. A strong downward pressure of prices is expected in telematics services.
2. Financing models:
   a) Subscription fees: like I-Mode in Japan, about 3 euro per month.
   b) OnStar model by General Motors (telematics services): monthly charges of US$ 15, 1 million customers.
   c) The type of the mobile device plays a significant role.

2.3 Data Security:

1. High user requirements (comparable with elk test for cars).
2. Mistrust toward stored information will be increased if the user requirements are ignored.
3. A process of getting used to operating the services is expected.
4. Trusted suppliers will be certified.
5. Permanent biometric access control will oblige the security needs.

2.4 Personalization

1. A critical attitude of elderly users is expected.
2. Users will require some influence on the level of personalization.
3. Personalization may help to overcome weaknesses of the system.
4. Users might not be willing to explicitly indicate their preferences.
5. Even slightly negative experiences with the system will create a negative user attitude against the device and the technology.

3 Concept of Functionality for Mobile Use

Today's users of mobile devices expect to have access to data and services in the car. The large differences in development and life cycles of cars and communication devices have so far prevented the establishment of a standard for the majority of mobile devices. State-of-the-art solutions either only serve some customer groups (like adaptors for mobile phones) or they do not offer complete functionality when used in the car.

The car manufacturers develop proprietary infotainment systems with high effort. The components of this type of system are designed and developed particularly for in-car use. These integrated solutions have so far overcome the external solutions due to a more comfortable operation. But they are restricted to a very small number of services, like navigation, and operation of the audio system and the telephone. Meanwhile, all car manufacturers offer solutions based on a central display and several operation elements (buttons). In-car telephony has pushed the integration of particular audio hardware, like microphones, amplifiers, and noise cancellation techniques.

The future scenario represented in SMARTKOM-Mobile aims at combining the advantages of a hardware particularly adapted to car environments and the flexibility and large variety of services available for mobile devices. The communication system logically and physically runs on the mobile device, while the car provides a particular hardware for input and output operations. The driver benefits from the larger and better situated in-car display and the improved audio input and output functionalities provided by the adapted microphones and the sound system in the car. The GPS data received from the car may potentially be used in combination with car data like velocity and steering angle.

In the user requirement studies we found that the different SMARTKOM services need to be combined in many situations in order to solve a task in a scenario (like calendar and route planning). A higher-level knowledge source is required to combine the services and to coordinate the processing order. Repeating input prompts not only irritate the user but also leads to more complicated interaction models even for simple tasks. For example, it is not necessary to ask a user who has already reserved several rooms in the same hotel for his preferences in subsequent hotel reservation dialogues. Repeating prompts result in a lower user acceptance and hence should be avoided if possible. This information may be achieved by storing the user profile

and using it in the intelligent background processing. Such a procedure allows for efficient human–computer interaction and an improved user-friendliness.

The goal of the development in the mobile scenario is a mobile communication assistant. The general framework in such a mobile environment significantly differs from the home and public environments. Furthermore, the mobile environment in the car and for a pedestrian/traveler yield different characteristics. The user is in a particular situation while driving:

1. The main task of the driver is to concentrate on the traffic and to drive in an adapted manner. The SMARTKOM services need to be realized in such a way that the driver may operate them while driving using speech, but incidentally. The SMARTKOM services aim at assisting the driver without causing safety risks through driver distraction. The interaction should incorporate only little graphical output presented in a simple manner. Input by means of gesture or pen will not be considered, due to the their higher level of driver distraction. Therefore, speech plays a very important role as input and output modality in car applications.
2. The services should only interact with the driver in situations without safety risks. The dialogue needs to be aware of the situation if possible. Possibilities to interrupt and to continue the dialogue should exist.
3. The environment of the driver is characterized by noises of the car, the street, and the weather conditions, which interfere with speech input and output.
4. The user will operate the service in the car. The functionalities of the system will be offered whilst driving and when parking. When the car stops, additional modalities, like pen and gesture input may be considered..

3.1 Transition Between Mobile Devices

The conceptual framework of the mobile functionality considers a scenario in which the user may switch between the two mobile environments "car" and "pedestrian". The user enters the car with an active mobile device (PDA) and fits the device in the designated cradle. The mobile device maybe operated by pen if the car is not moving. For moving cars, however, only the car-specific input and output modalities should be considered. The user pulls the push-to-activate (PTA) lever at the steering wheel, which causes the transition from the mobile device to the car-specific hardware, namely the central car display and the audio input and output devices. Now the driver may operate the mobile services using speech dialogues by pulling the PTA lever. If the user arrives at destination he may deactivate the car-specific hardware by pressing the PTA lever in the car. This deactivation transfers the activity to the previously active in-car device-specific hardware (the mobile device in our case). Alternatively to transfer the activation to the mobile device would be to press the PTA button at the mobile device.

4 System Architecture

The two mobile environments, car and pedestrian, use their particular communication devices, e.g., the car-specific hardware and the PDA. Since it is impossible to run the entire SMARTKOM system on state-of-the-art PDAs and car hardware due to limitations in memory and processing power, both systems have to communicate with the SMARTKOM main processing system, which runs on a computer with high processing power.

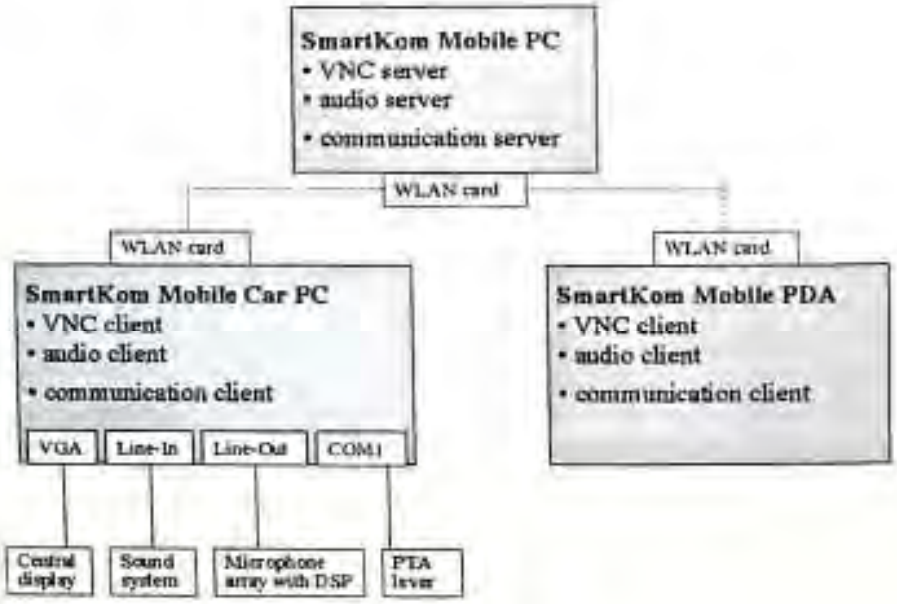


Fig. 1. Architecture of the SMARTKOM-Mobile system

The consortium agreed on the usage of an iPAQ H3600 Series PDA by Compaq as mobile device and a standard laptop as the in-car device. Both devices will run PocketLinux and communicate using wireless LAN[1] (WLAN) and the standard protocol TCP/IP. The PDA will only be used as input/output device by incorporating its display, the pen, the internal microphone, the speaker, and the PTA lever. The in-car PC is connected to the car- specific hardware: the central display, a microphone array, the sound system, and the PTA lever at the steering wheel. Both mobile devices only serve for transmitting the audio and graphic data and as input/output devices. The processing of the data in a dialogue system is handled on the SMARTKOM PC. The concept and implementation for the audio server and clients were realized at MediaInterface GmbH, Dresden. The audio clients can be adapted to the environmental

[1] The limited range of WLAN was accepted in the consortium, since communication technology is not the main focus of the project and UMTS technology will enable sufficient bandwidth and high availability within the next few years.

noises, which greatly differ between the two mobile scenarios. Such an adaptation leads to a more precise voice activity detection in the speech recognition front end. The communication server and client were developed at the European Media Lab, Heidelberg.

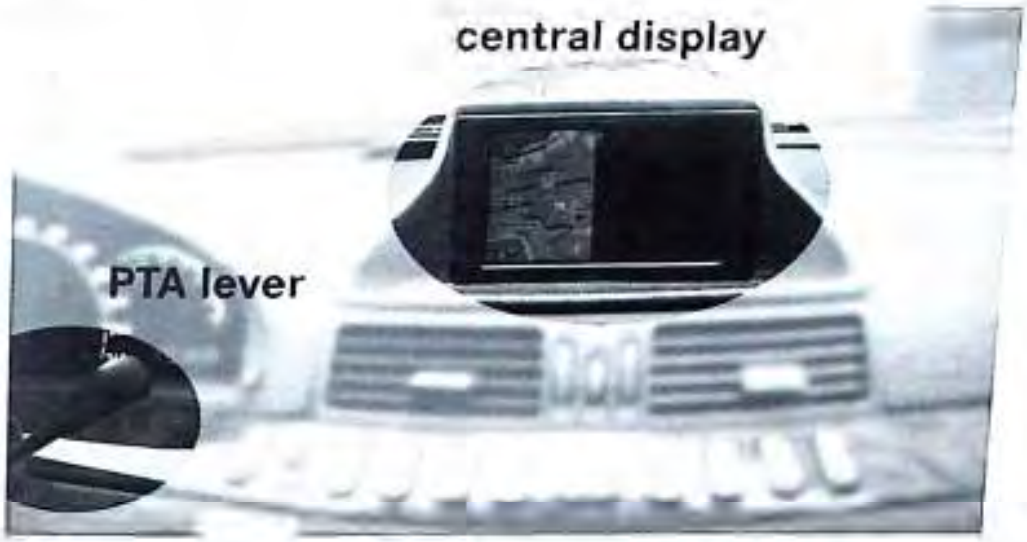


Fig. 2. Integration of the display and the push-to-activate (*PTA*) lever in the car

The SMARTKOM-Mobile environment in the car consists of the speech input and output and the graphical output modalities. A PTA lever is incorporated to initialize the speech input. The display will not have a touch screen functionality since grease residues on touch-screen displays would not appear valuable. The following components have been installed in the car and connected to the SMARTKOM system using the mobile car PC.

4.1 Central Display for Visual Output

A 7-inch coated backlight-amplified central TFT display from Toshiba with a resolution of 800 × 480 pixel has been integrated in the car. Since the resolution of the display differs significantly from the resolution of the iPAQ (240 × 320 pixel), a different presentation mode was developed in cooperation with the DFKI. The car display is segmented in two presentation zones. On the left, the iPAQ display is reproduced. On the right, detailed information about sights or parking places may be displayed. If the user does not ask for detailed information, the right display area contains a large representation of the virtual animated agent Smartakus like in the SMARTKOM-Public scenario. The cockpit also contains buttons to switch the display on and off and to adjust its brightness.

4.2 Sound System for Audio Output

The audio output was realized by streaming audio data via the mobile car PC to the stereo speakers in the front doors. An amplifier improves sound quality.

4.3 Microphone Array for Speech Input

An array of four microphones has been installed in the car for speech input. It has been calibrated toward the head of the driver, so that simultaneous speech input from the codriver can be compensated to a large extent. A particular signal processor (DSP) allows for reducing noise and combining the four microphone signals to a monosignal, passed to the speech recognition component.

4.4 PTA Lever for Input Initialization

The PTA lever on the right side of the steering wheel is connected to the mobile car PC via a microcontroller and a serial connection. The lever is pulled either to activate the car environment and its mobile device or to start speech input. Pushing the lever deactivates the mobile device in the car and switches the activation to the mobile device that was active before the one in the car. Most users decide, however, to activate the desired device instead of deactivating the undesired one, since all available devices are registered at the SMARTKOM system.

5 Dialogue Design

When designing multimodal dialogues for mobile environments the situation of the user needs to be considered, since some modalities maybe reasonable and desired, while others may distract the user, violate privacy, or disturb other people. The transition between different modalities needs to be modeled accordingly, i.e., the pedestrian user is able to use a combination of speech and pen input on the PDA, while the car driver has to rely on speech input only.

Dialogues have to be designed with respect to the users, technical constraints, budget, and time. The user requirement studies allowed us to specify the desired functionalities and modalities in detail. Wizard-of-Oz experiments needed to be performed, due to the high complexity of the system. These experiments, carried out at Ludwig-Maximilians-University, Munich in cooperation with the partners who designed the mobile scenario, i.e., European Media Lab and DaimlerChrysler, influenced the design process strongly. The dialogues were realized as naturally as possible using mixed-initiative, natural speech, barge-in, and brief prompting. With one aim of the project being to give the user a high level of control of the interaction, suitable repair mechanisms for potential errors had to ensure that the dialogue strategy appears clear and consistent to the user. Another consideration was to keep the internal organization of dialogue structures as simple as possible by breaking complex dialogues into smaller, simpler modules, if possible.

SMA: [Map of Germany]
USR: Ich will nach Heidelberg fahren.
SMA: [Map of Germany] Wo wollen Sie starten?
USR: Hier.
SMA: [Map of Heidelberg] Wo in Heidelberg wollen Sie hinfahren?
USR: <Points to Peterskirche> Da möchte ich mehr drüber wissen.
SMA: [Info presentation] Hier sind Informationen zur Peterskirche
USR: Ich will zur Peterskirche.
SMA: [Map of parking places] Wo wollen Sie parken?

USR: Gib mir Informationen zum Parkhaus Kisselgasse.
SMA: [Info presentation] Hier sehen Sie Informationen zum Parkhaus Kisselgasse.
USR: Da will ich parken.
SMA: [Map with auto route] Hier sehen Sie die Route von Ulm nach Heidelberg. Ich kann Ihnen auch den Fußweg zur Peterskirche zeigen.
USR: Ja, bitte.
SMA: [Map with pedestrian route] Hier sehen Sie den Weg vom Parkhaus Kisselgasse zur Peterskirche.

Fig. 3. A sample route planning dialogue using different modalities

6 Mobile Services

The consortium agreed to focus on services that are highly desired according to the user requirement studies and that are not already available in state-of-the-art cars and mobile devices. Therefore it was decided to implement route planning and parking place reservation as car-specific services and an integrated route planning, which is a mixture of car and pedestrian navigation service using a device transition from car to PDA. Standard speech-operated functionalities like air conditioning and audio operation were not considered since they are already available in state-of-the-art cars. The module that implements the mobile services designed for the car was dubbed *service.navigation.car*.

This module contains the following three core functionalities: route planning, mapping, and parking place reservation. The route planning consists of calculating a route between geographical locations according to the type of the route (shortest or quickest). The mapping retrieves a map image for a geographical region and allows for panning and zooming the map image. The parking place reservation consists of a database of parking places and garages. Each database entry is annotated with its geographic location to be shown in the map. The entry also contains detailed information about opening hours, the number of available parking spaces, and the parking fee. The functionality also allows for the calculation of the parking places that are closest to the current position of the car.

The simple route planning service for cars incorporates car route planning and mapping functionalities, while the complex integrated route planning service also incorporates the car route search to the destination city, the pedestrian path to the destination within that city, and the search for a parking place close to the desired destination. The car route is active until the parking place has been reached, when activity is switched to the PDA and pedestrian route guidance is activated. The user

may also ask for detailed information about parking places and places of interest in the destination city; the latter are available from the module for pedestrian navigation. A complete integrated route planning dialogue is shown in Fig. 3.

The M3L protocol employed is defined in the *geographical.xsd* XML schema, which guarantees compatibility with the *service.navigation.pedestrian* module for the pedestrian navigation. The protocol is based on request–response exchanges that are initiated by the function modeling module.

The map request function returns a high-quality bitmap image of the desired geographical region. The effective extent of the geographical region may be specified exactly or, alternatively, be inferred from the geographical extent of the vendor objects to be shown in the map. The vendor objects include calculated routes and points of interest (city locations).

The module *service.navigation.car* can be configured to work with different map data sets. For the SMARTKOM project two data sets have been devised: a full map of Germany, and a smaller map of southern Germany. The reason for using different map data sets relies in the extensive amount of disk space (about 500 megabytes for the full map) that is needed for the data.

By default, streets and cities are shown in the map image. When using the full map data, a geographical relief is also shown as a background image. The binary bitmap data is encoded as a JPEG file. The compression level is configurable and allows us to significantly reduce the amount of transmitted data.

Technically, the first version of the module *service.navigation.car* was implemented as a Windows Java application that interacted with the Navigon 2000 product via an ActiveX interface. This interface, however, proved to be rather cumbersome to use, and so for the SMARTKOM 2.0 milestone an individual solution with direct access to the underlying libraries was implemented in C++. This also made it possible to provide the complete module and data sets within the SMARTKOM distribution.

The existing functionality is sufficient for the current scope of the car application. However, some limitations seem to be apparent:

1. The service does not return high-level information about text labels. This information could be important for interpreting user gestures, which tend to focus on the text label rather than on the city center, which is also represented in the map. Also, for the presentation of the map (e.g., the placement of Smartakus), it would be advantageous to highlight potentially important text in the maps.
2. It is unclear if the font sizes used (that seem to be pixel-based) are adequate for different resolutions (in dots per inch). A bitmap approach would be rather inelegant for presentation, since it prevents us from incremental presentations of routes and other objects.

7 Outlook: Toward Flexible Modality Control

We argue that a truly multimodal interaction paradigm must not place restrictions on the user or on the environment as to which modalities to use. Instead, a multimodal

dialogue system should be flexible enough to be used even as a monomodal system, if it is required. The situations in which restricted multimodal interaction is useful are manifold: Unrestricted multimodal output may lead to privacy problems. Requiring multimodal input, such as pointing gestures, from the user who may be occupied with something other than interacting with the system may also lead to inconvenience, or, much worse, to security problems, most apparently in the SMARTKOM-Mobile scenario.

The technical realization of a flexible modality control will be an enormously complex task. However, we are convinced that SMARTKOM and M3L would provide a suitable groundwork for doing so.

In this section we concentrate on the SMARTKOM-Mobile scenario and try to identify its particular requirements for modality control from an application-oriented perspective.

Based on experience in developing and testing speech-based user interfaces in mobile environments [1] we have identified five major combinations of modalities, *interaction modes*, that seem characteristic to the SMARTKOM-Mobile scenario: (see Fig. 4):

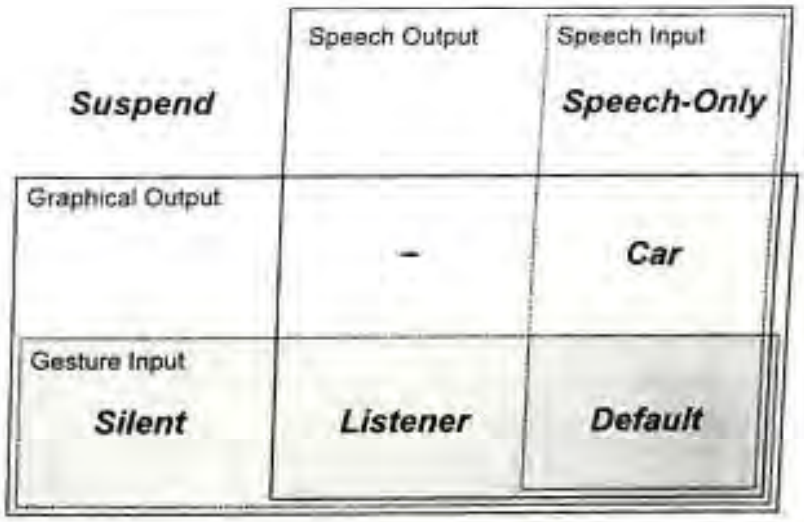


Fig. 4. Interaction modes as combinations of communication modalities in SMARTKOM-Mobile. Transitions are described in the text

1. **Default:** Mainly used in the pedestrian environment and when privacy or disturbing others is not an issue. All modalities should be enabled in this mode. Stemming from safety considerations (discussed below), this mode is available in the driver environment only when the car is not moving.
2. **Listener:** In this mode the user accepts spoken system output but he does not use speech as an input modality (i.e., the user is the listener here). Spoken output is useful for quickly providing concise summaries or extended background

information to the user whose focus of attention is placed on tasks other than interacting with the system. The user may employ earphones in order to protect his privacy. The listener mode may also prove useful for avoiding confusion of the system by off-talk. In this mode, the system does not try to excite spoken input from the user. It should therefore generate prompts like "*Show me ...*" rather than "*Tell me ...*".

3. **Silent:** This mode is useful when spoken language human–machine dialogue is problematic, for instance, in certain public places or in public transportation. The interface should be similar to traditional graphical user interfaces.
4. **Car:** This mode is a restricted version of the default mode for use mainly in the driver environment. Speech is the dominant communication modality, while graphical displays are used only for presenting additional (nonvital) information, such as maps for presenting a route while driving or the locations of parking options. Consequently, the display does not need to be touch-sensitive.
5. **Speech-Only:** This mode is mainly used in the driver environment notably when the car is moving. Any additional safety risk (such as driver distraction from the driving task by gesture input or graphical output) must be avoided. System interaction is restricted to the audio channel, i.e., the use of a speech-only dialogue is mandatory.

In the driver environment an emergency requires the dialogue with the system to be temporarily suspended. This *Suspend* situation (Fig. 4) occurs when the driving situation becomes dangerous and the entire user attention is required. Although the system suspends any interactivity, it should be prepared to resume the dialogue, especially after very short interruptions. Temporarily suspending interaction may also be useful in the pedestrian environment when the user is engaged in task others than communicating with the system, such as crossing a street or talking to another person.

Having identified these five interaction modes, we need to determine under which circumstances intermodal transitions are possible. We distinguish between *user-* and *system-driven* transitions. A user-driven transition is the result of an explicit command to use or not to use a specific modality. In turn, system-driven transitions are triggered by the system without any user interaction.

The user should be able to switch between modalities anytime, although some modalities may temporarily be disabled for safety reasons. A user-driven modality transition may be performed by the following actions:

1. The user may suspend/resume the operation with the system. In the pedestrian environment this operation could be performed by activating a button on the portable device. In the driver environment this could be realized by a push-to-activate button or lever.
2. In the pedestrian environment the user may switch off speech input and output using a button on the portable device.
3. In the pedestrian environment, the user may toggle display operation. A request for turning on the display could be recognized by the touch-sensitive screen of the portable device.

4. Reopening the speech channel by uttering spoken commands re-enables spoken language dialogue, e.g., when the system interaction is suspended or when graphical output is exclusively used.
5. Requesting graphical output by spoken commands (like "Show me the map") enables graphical output, unless this modality is disabled for safety reasons (see speech-only mode).

In addition to the user, the system may initiate a modality transition in one of the following ways (Table 1):

Table 1. Possible modality transitions in SMARTKOM-Mobile

Signal	System reaction
User-driven transitions	
Push-to-activate	Suspend/resume operation
Command/button	Turn on/off speech
Command/button	Turn on/off graphics
System-driven transitions	
Install device in the car	Turn off gesture input
Take device out of the car	Switch to default mode
Start car	Turn off graphical displays
Stop car	Turn on graphical displays
Detect emergency	Suspend operation
Misrecognition/noise	Turn off speech input

1. As outlined above, specialized software running on the car PC may detect an emergency in the driver environment. To this end, the software has access to the necessary pieces of information about the state of the car (e.g., driving speed, state of brakes, etc.) through the CAN bus.
2. Analogously, starting the car is detected by the car PC and is interpreted as a transition into the speech-only mode. Likewise, stopping the car leads to the default mode.
3. In the pedestrian environment, i.e., when graphical output is available, repeatedly failing to understand spoken input from the user should lead the system to infer that the current situation is not suitable for speech recognition (e.g., though strong background noise). The system therefore switches to the listener mode in order to not rely exclusively on speech input.

8 Conclusion

A large variety of electronic devices serve for comfort, ease of driving, entertainment, and better communication in modern cars. These devices as well as future services require a rather complex human–machine interaction. Our work focused on

new interaction modalities, which may allow for fewer driver distraction and higher comfort for operating the devices and future mobile services.

Studying user requirements for future mobile services we found that the most important issue will be which interaction modalities (speech, gesture, mimics) may be applied for which services, how to finance the services, how to guarantee data security and preserve privacy, and how to personalize the services. The services most desired by the subjects, who were asked to fill in the questionaire of our study, were the mobile phone, emergency calls, dynamic route planning, parking place reservation, and health check.

In a theoretical discussion we analyzed the interaction modalities (speech, gesture, mimics) that are suited for operating the services. This discussion led to the conceptual framework for the mobile services. We argued that in the future scenario represented by SMARTKOM-Mobile, the advantages of hardware particularly adapted to car environments and the flexibility and large variety of services available for mobile devices should be combined in such way that the communication system logically and physically runs on the mobile device, while the car provides a particular hardware for input and output operations. The technical realization of the mobile services carefully integrated the hardware available in future cars into future mobile services. Such hardware may be a microphone array, a sound system, and a large display.

In order to adapt the interaction to human habits and to render it as natural and efficient as possible, we considered the most important principles and guidelines in dialogue design, and applied them to specify the new services, their interaction modalities, the transition between modalities, and the transition between different mobile devices.

The consortium agreed to focus on services that are highly desired according to the user requirement studies and that are not already available in state-of-the-art cars and mobile devices. Therefore the following mobile services were specified and implemented in SMARTKOM: route planning and parking place reservation as car-specific services and an integrated route planning, which is a mixture of car and pedestrian navigation service using a device transition from car to PDA.

In an outlook, we have presented a framework for a flexible modality control oriented at the needs of the SMARTKOM-Mobile application scenario. The main goals are to illustrate the necessity of being able to handle modality transitions, if an efficient and unobtrusive multimodal interaction is to be achieved in different usage situations. An implementation will, of course, face interesting challenges. Even if the events signaling a modality transition can be detected effectively and a modality-independent output mechanism is in place, the question of how to achieve a consistent interaction experience will arise. Nonetheless, a modality control framework is clearly necessary, not only in the face of new interaction devices with different capabilities, but also for the sake of a personalization of the interaction (considering, for instance, speech- or sight-impaired users).

References

U. Gärtner, W. König, and T. Wittig. Evaluation of Manual vs. Speech Input When Using a Driver Information System in Real Traffic. In: *Int. Driving Symposium on Human Factors in Driver Assessment, Training and Vehicle Design*, Aspen, CO, 2001.

P. Green. Crashes Induced by Driver Information Systems and What Can Be Done to Reduce Them. In: *Proc. Convergence 2000*, Warrendale, PA, 2000. Society of Automotive Engineers.

W. Minker, U. Haiber, P. Heisterkamp, and S. Scheible. Intelligent Dialog Overcomes Speech Technology Limitations: The SENECa Example. In: *Proc. Int. Conf. on Intelligent User Interfaces (IUI) 2003*, Miami, FL, 2003.

Part VI

Data Collection and Evaluation

Wizard-of-Oz Recordings

Florian Schiel and Ulli Türk

Bavarian Archive for Speech Signals
c/o Institut für Phonetik und Sprachliche Kommunikation
Ludwig-Maximilians-Universität Münchenn, Germany
{schiel,tuerk}@bas.uni-muenchen.de

Summary. This chapter gives a concise overview of the empirical Wizard-of-Oz recordings done within the SMARTKOM project. We define the abstract specifications of the intended simulated communicative situations, describe the necessary technical setup (including numerous useful practical hints), and finally outline the specifications of the resulting multimodal corpus, which may be obtained from the Bavarian Archive for Speech Signals (BAS).

1 Introduction

The SMARTKOM data collection was aimed to provide empirical data about human behavior in human–machine communication (HMC) for several purposes: to train statistical models for pattern recognition as well as user modeling, to gain insight on how humans may react or interact in HMC situations and to elicit situations where the human displays a certain degree of an emotional state. [1] To achieve these goals a so-called Wizard-of-Oz recording paradigm (WOZ) was used throughout the investigations. That is, users were not aware that they were interacting not with a machine but rather a simulated machine.

The successful simulation of a complex system with several input and output modalities such as SMARTKOM is a very complex task that requires a large range of expertise ranging from purely technical to psycholocigal knowledge. However, in this contribution we will concentrate on two main issues:

- How to define a task for a human user that enables her to interact with a simulated system without any explicit restrictions. On the one hand, we would like to give the user as much freedom as possible to encourage her to show new ways to interact multimodally with a system like SMARTKOM. On the other hand, we have to observe the implicit restrictions of a limited technical system. A WOZ simulation would be rather implausible if the system is able to process any kind of human multimodal input data.

[1] Please note that the description found here is rather abbreviated due to space limitations. More detailed descriptions on the topics outlined here are found in the references given at the end of this chapter.

- How to implement a technical recording setup that not only allows the successful simulation of the intended system but simultaneously allows the recording of all relevant communication channels between human and machine. The simulation must run smoothly, without any artifacts that might tip the user of the simulation and allow the controlling wizards to react appropriately within the assumed restrictions of the simulated system. The data must not only be gathered in real time but must also be synchronized across several data types and sampling frames.

Many other issues such as the design of a graphical interface that does not exist yet, the design of task flow plans for the simulation, the evaluation of the simulation, the training of the controlling agents (the so-called wizards) behind the system, the annotation, labeling and categorization of the recorded data, and others cannot be addressed here. Some of them are discussed with more detail in the literature given at the end of this contribution.

This contribution may serve two purposes for the interested reader: First, he might get some important background knowledge about the SMARTKOM Multimodal Corpus that may be obtained at the Bavarian Archive for Speech Signals (BAS, http://www.bas.uni-muenchen.de/Bas). Second, he might learn about the specific problems of such a large-scale WOZ recording and get some hints and tricks on how to circumvent known problem areas if he intents to do his own investigations.

The next section deals with is the first problem addressed above, likewise section 3 gives a concise description about the technical setup that was used in the SMARTKOM WOZ recordings. Section 4 summarizes the main features of the resulting multimodal corpus.

2 Recording Plan

2.1 Overview and Definitions

The SMARTKOM WOZ recordings consist of a large number (to satisfy the statistical requirements) of recording sessions with a single user interacting unsupervised with the system. The user gets a short instruction and a task to solve with the help of SMARTKOM. She is convinced that the system is real in the sense that it is an automatic technical system. In reality she is interacting with two so-called wizards steering the technical "surface" of the system.

The *recording plan* is the script of the experiment. It contains all information regarding the experiment, ranging from the instruction to the user to the rules of behavior for the wizards, and gives an overview on how all these single components of the experiment shall be combined. Of course, the recording plan implicitly defines the expected data that will be recorded during the real experiment. However, there is always room for surprises, and there should be.

The fully detailed recording plan is documented in Steininger and Kartal (2000) and implicitly also in the corpus documentation and corpus metadata. Here we will restrict the presentation to the basic design principles and performance rules. [2]

2.2 Experimental Design

2.2.1 Typical Session

A typical SK recording session[3] consists of the following phases:

- Questionnaire for user
- Instructions for user
- Session 1 (4.5 min)
- Second instruction
- Session 2 (4.5 min)
- Interview or questionnaire
- Gathering of user metadata
- Signing of nondisclosure agreement and payment of incentive

The user fills out the first part of the user questionnaire. After that the supervisor leads the user to the recording room, instructs the user briefly on how to use the system, "starts" the system and leaves the room.

The first session begins; it lasts 4.5 min. The supervisor is in radio contact with the two hidden wizards and may decide to interrupt the session if something seriously goes wrong.

The settings for each of the two sessions are defined according to an overall session plan, which defines in which technical scenario (Public, Home or Mobile) the task takes place, which primary and secondary task is to be solved and some other conditions as listed in Sect. 2.3. Sessions 1 and 2 usually have different definitions. In most cases the tasks are different; in some cases they might be based on each other (that is, a task from session 1 is continued in session 2).

After the second session, the second part of the questionnaire is filled out by the user. Then the user metadata are registered in the database (later exported into the "spr" file of the corpus), and the session definitions are stored into the database as well (later exported into the "rpr" file of the corpus).

Finally, the user signs an agreement that the recorded data may be used for the intended scientific investigations and receives her incentive. The user is strongly urged not to reveal the nature of the recordings to others because this might lead to severe recruiting problems (only "naive" users can be used for WOZ recordings).

[2] The recording plan as presented here is based on a concept that was proposed in Dec. 1999 to the SK consortium, augmented by contributions of the partners in Jan. 2000, at the second main SK workshop and again after the MoKo workshop of 2001.

[3] In the future simply referred to as "session."

2.2.2 Description of the Scenarios

There are three basic technical scenarios in SMARTKOM. We will give a very brief description of the setups as they occur to the user. Further technical details are found in Sect. 3.

Public and Mobile

The user is standing in front of a table (Fig. 3). On the table a graphic tableau (Wacom A2) is mounted. The graphic tableau is not tilted as usual but is flat and covered with a purely white foil so that it is not recognizable as a graphic tableau. In some sessions the user has a pen available, in some not. Approx. 1.5 m above the table a cube about 40-cm long is attached to a double column behind the table. From there the graphical output is displayed onto the white surface of the table. Between the columns in front of the user a camera is aimed at the general direction of the head; a second camera is situated about 2-m away to the left-hand side. The sound output of the system appears to come from under or behind the table; other sounds like artificial background noise may be played in on a speaker set behind the user. In some sessions the user is required to wear a headset, in some a collar microphone, in some no microphone at all. The environment is neutral office equipment with some sound damping elements on ceiling, floor and walls. The visual background behind on the right-hand side of the speaker is grey–white; the background behind the speaker is colored in different abstract patterns and can be changed from session to session.

In the Mobile scenario the size of the projected output is reduced to indicate the small display in a car or on a PDA. The user wears a headset, because SMARTKOM-Mobile will not have any far-field microphones.

Home

The user is sitting on one of two low, comfortable living room chairs. One of them faces a low table of the size and height of a coffee table on which the display is projected; the other chair faces, in the opposite direction, a TV set and a VCR. On top of the VCR there is a video tape ready for use. The user wears a collar microphone that is connected to the ceiling so that the user can move freely in the room. Behind the coffee table is a large bookshelf along the wall; other living room accessories are scattered across the room.

2.2.3 Dealing with the User

The recruitment, briefing, questioning and instruction of the user is the task of the experimental supervisor. The supervisor has to take care that:

- The user takes the WOZ simulation for real.
- The user is not reluctant to deal with the machine.
- The user is relaxed.
- The user acts naturally and does not comment all her actions.

- The user uses multimodal input channels and does not concentrate on one channel only.
- The user might also show something about her user state.

During the recruitment the supervisor creates the illusion that the user will be involved in a market study. This diverts the attention of the user from the technical details to the "usability" of the system and thus improves the plausibility of the simulation. Between the sessions the supervisor might ask the user if she used gestures or not and encourages her to do so in the next session etc.

As for the instruction itself, it turned out that written instructions are not very effecive (especially with elderly users). Verbal instructions by the supervisor according to an (internal) manual and a short handout with very short keywords that the user may reread just before the session starts were better. Of course, both the manual and the handout have to be adapted to the selected session definition. It seems to be important that the core instruction (the "task") can be formulated in one sentence only. Otherwise the user starts with repetitions of parts of the instruction to remind herself of the different points during the recording, which is clearly not what is wanted.

The supervisor must verify that the user is ready for the session and fulfills all the points listed above. If not, he may add some more coaching before he decides to start the session; in extreme cases he might reject the user altogether.

Fig. 1. The simulated GUI in scenario Public and domain Tourism before the dialogue starts as seen by the controlling wizard (navigation bar is visible to the *right*)

2.2.4 Graphical User Interface (GUI)

The graphical user interface (GUI) consists of the SMARTKOM logo, a help field, the persona or agent, and the main display area where the information is presented. It turned out that users almost never pointed to the help area. In most cases help was ordered verbally. For the simulation the GUI is implemented as simply HTML pages enriched with animated flash sequences. To avoid association with the standard PC GUI all familiar elements like mouse pointer, slider and menus were removed. The controlling wizard sees the same display as the user but has a visible mouse pointer as well as a navigation bar to the right where he can control the changes of pages or other elements of the GUI. The navigation bar is, of course, not visible to the user. The information content may be short texts, keywords for the next command, pictures/graphics or maps.

The system status is displayed to the user via different appearances of the persona. The persona in our simulation does not really interact with the user; that is, we do not have an animated speaking face. The persona Smartakus — which was the name the users were asked to use to address the system — showed only four different system states:

- inactive ("sleeping")
- active and listening
- active and processing
- system error

Fig. 2. The WOZ version of the persona "Smartakus" in the state "processing"

In the Public scenario the persona starts in the "inactive" state and goes to "active and listening" when pointed to or when addressed with the name Smartakus. In the Home scenario the persona goes into "active and listening" state when the portable display is switched on. In the Mobile scenario no persona was shown on the display because of the reduced size of the display; the system is always in the "active and listening" state here.

The simulated system will only react to input in the state "active and listening"; all inputs at other times — even partly — are ignored. A more detailed description to the simulated GUI can be found in Beringer et al. (2000).

2.2.5 Speech Output

The voice output is spoken by one of the wizards and then electronically distorted to achieve a machinelike voice. We do not use a real synthetic voice because it turned out that typos in the synthesis raise more suspicion than speaking errors. The wizard can open the channel to the recording room by means of a foot pedal. A red warning light in the control room indicates that the channel is open and that no other noise should be produced. Alternatively, he may activate prerecorded standard sentences, but this requires a full set of prerecorded samples per wizard team.[4] The voice output is softly restricted to a list of sentences or sentences types, respectively. The wizard should preferably stick to this list of possible outputs. The lists of preferred output sentences is context-dependent. That is, for each major display the wizard has a special list of sentences available (of course, some situation-dependent sentences are always possible, like error messages or help messages).

2.2.6 Simulated Contents

The simulated system must, of course, provide information to the user to enable her to solve the task successfully. Since all scenarios had to take place in the German city of Heidelberg, most of the contents were taken from there.

The date (if shown on the display) is always that of the day of recording. The TV program is based on the EPG database provided by the partner Philips and updated by this partner on a regular basis. The cinema program of Heidelberg, Heidelberg maps as well as restaurant and tourist information were not changed during the experiments. This might lead to a reduced vocabulary in terms of content words. However, the effort to keep a fully updated simulation all the time for more than three years was simply to high.

2.2.7 Guidelines to the Wizards

Two wizards control the simulation from a hidden room. One is responsible for the voice output; the other for the GUI. The user may interact verbally or via gestures with the system. All inputs are interpreted by a wizard; the user has no real controlling abilities. Both wizards can watch the user's face and the upper body from the left as well as the display area. The speech output wizard needs a considerable amount of training to always speak with a constant rate and a more or less normalized intonation. Also, he has to avoid any other noises such as coughs, laughing, clearing his throat, etc.

General rules for the wizards are:

[4] In prime times we had up to five teams working in shifts.

- Try to simulate the intended SMARTKOM system as well as possible (this includes misinterpretations as well as technical errors). The "intelligence" level of the simulated system is significantly higher than would be expected with today's (2000) state-of-the-art technology.
- Try to achieve a natural and varying dialogue with the user.
- Only clear gestures that are in relation with the actual graphical display are recognized; gestures outside the interaction area (for instance, in front of the body) are ignored.
- The user may use any kind of (correct) sentence or command to formulate a question, granted that the sentence is not to long (more than 25 words) and not to complex (e.g., contains subclauses).
- Even long and complex utterances are accepted by the wizards, if the very first part contains a simply statement or question, e.g., in the spoken input sentence:

  "I'd like to go to the cinema tonight ... can you help me ... at best I'd like to go to a cinema in the center ... oh, yeah, and it should have THX ..."[5]

  the wizards will ignore everything except "go to the cinema tonight".
- If the dialogue stalls (for instance, if the user does not know how to go on), the system will offer simple assistance. For instance, it will indicate how to give input to the system:

  "You may use free speech or indicate things on the screen"

  or which options exist at a certain dialogue stage:

  "If you want, I could make some suggestions."
- If the user consequently uses only one modality for input, the wizards try to steer her into a multimodal interaction.
- Technical instructions, e.g., "Please try to repeat or use other words for your questions" are embedded in the dialogue; there exists no training sequence at the beginning of the dialogue.
- If the dialogue runs too smoothly, the wizards will insert artificial errors into the processing.
- If a user thinks she has already solved the task before the recording time of 4.5 min. has ended or the user "sticks" at a problem and does not go on with her task, the wizards will cause an error to continue the interaction.
- If a user shows signs of anger, the wizards will try to enforce that emotion by using wrong answers persistently, etc.
- If the user tries to leave the possible domains, the wizards will block this by issuing a standard sentence like, "This function is not implemented yet. May I help you with something else?"
- The wizard has to avoid repetitive outputs, because it is almost impossible to reproduce the same sentence with the same rhythm and speaking manner. It is very risky that the user may detect such differences and deduce that this cannot possible be a machine speaking.

[5] All examples in this contribution were adapted to English.

2.3 Session Plan

For each recording there exists a so-called session plan that defines the varying parameters (called "dimensions") of the individual session. The distribution of dimensions across the individual sessions is controlled by the global session plan, which in turn reflects the requirements of the project partners.

The dimensions of each session concern the technical setup as well as the task the user has to solve. There are nine basic dimensions that may vary from session to session:

- Scenario (Public, Home, Mobile): This dimension defines a large number of technical settings and changes to the recording setup. Therefore the change of this dimension is something that can be done only a few times during the recording period. See Sect. 3 for more details about this.
- Primary domain (cinema, tourism, navigation, TV program, VCR, fax, task planner): The primary domain reflects the task explicitly given to the user, e.g., "With the aid of Smartakus, find a good cinema in Heidelberg for tonight." The primary domain also includes a role model for the user and a situation in which the user is at the moment of the recording. Both have to be explained thoroughly to the user before the experiment starts.
  For example, "You are on a business trip in Heidelberg; you had an appointment in the morning, and you are free for the afternoon and evening. You would like to see a movie, possibly together with a friend or colleague." The primary task also includes a defined number of possible complications that may or may not be implemented by the wizards during the recording. This is done to give the wizards some guidelines and to prevent too exotic problems that might come to the minds of the wizards.
- Secondary domain (restaurant guide, telephone, task planner, select music, e-mail, TV program, VCR): The secondary task is not explicitly given to the user, but to the wizards. The wizards will try to steer the dialogue that starts with the primary task to the secondary task. For instance, after selecting a movie and getting reservations in the cinema, the system asks the user if she would like to look for a nice restaurant before or after the movie.
- Politeness: The system addresses the user either with "Du" or with the more polite "Sie".
- Biometric authentification: This means that the user has to authentificate herself before using certain protected functions of the system (for instance, to use the task planner). This dimension was never implemented in the SMARTKOM recordings because the simulation effort was too high.
- Evoked emotions: In some sessions the wizards are instructed to evoke anger in the user.
- Pen: There are three types of sessions:
  - User has no pen; she will always use her hand or finger to gesture.
  - User is told to use only the pen to gesture.
  - User is told that she might use the pen or might use her finger.

- Background acoustic: For each technical scenario one of three different types of corresponding background noise is played back during the recording. The sound level is kept constant throughout the recording. Two conditions have a rather high noise level (S/N ratio approx. 20 dB, but not verified); one condition is of rather a low level such as in a quiet phone booth, a quiet home or a standing vehicle (S/N ratio approx. 35 dB).
- Background visual: The visual background for the lateral camera was constant (either gray for Public and Mobile, or a bookcase for Home). The background of the frontal camera, however, was changed between three different patterns:
  - the "real" background consisting of white acoustical damper elements mounted on the ceiling ("None")
  - colored cubical patterns with sharp edges ("Kubismus")
  - colored flowing patterns without any sharp edges ("Bunt")

The exact distribution of these dimensions across the recordings are described in Steininger and Kartal (2000) or in the metadata of the corpus, respectively (see Sect. 4.4).

3 Technical Setup

3.1 Overview

The SMARTKOM system should be able to recognize natural speech as well as gestures on a flat interaction area. Additionally, facial expressions are analyzed with regard to visible emotions or so-called user states. The output of the system is presented with a GUI, which is projected on the interaction area, and with synthesized speech (see Fig. 3 for a schematic view of the recording setup).

In WOZ experiments subjects are recorded in sessions of 4.5-min length while they are interacting with a simulated version of the system. During these sessions all capture devices of the system (and some more) are used for collecting data:

- Audio is captured using a directional microphone, a microphone array with four channels and (alternating) a headset or a stereo clip microphone.
- Video is recorded by two standard DV cameras (one for the facial expression, one for the side view of the subject) and by an infrared camera that is part of the gesture recognizer SiViT (Siemens).
- The graphical output is recorded in a low-frame-rate video (used only for labeling)
- Gesture coordinates captured by the SiViT system and the graphical tablet.

Figure 4 shows the processing stages during and after the recording together with resulting changes on the data file set. Except for the last two stages, which are covered by one team, each block represents a different working team. In the following we give a detailed description for each stage, focusing especially on the capture of synchronous multimodal data with standard equipment.

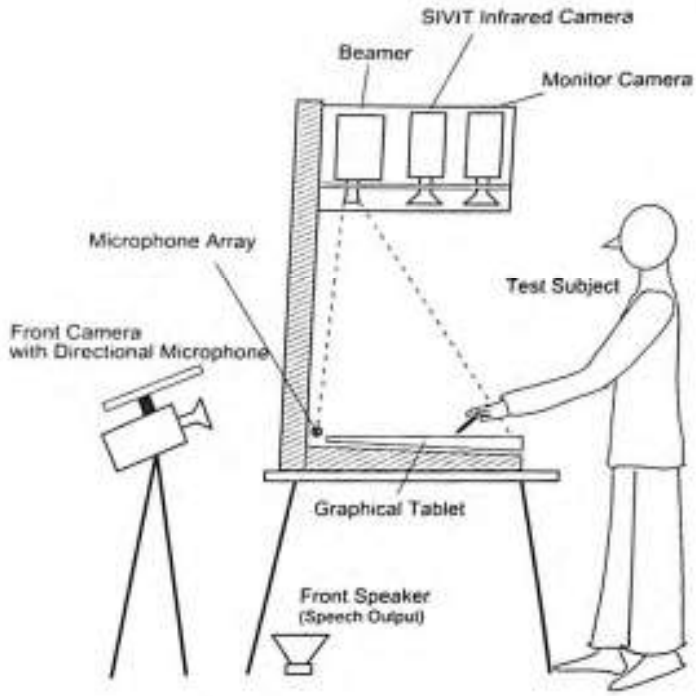


Fig. 3. SMARTKOM setup used for data collection in the Public scenario

3.2 The QuickTime File Format

QuickTime[6] allows the integration of several kinds of media like text, video, audio, images and vector graphics in one multimedia file format. All SmartKom files are compatible with this technology. Together with these data files, we provide a so-called framework QuickTime file in the corpus. The user can play back the recorded session, select the tracks to be presented, choose the style of the presentation and navigate to individual turns in the recording. In addition, we use QuickTime's capabilities to render new data files from the existing set of files (see, e.g., Sects. 3.4.1 or 3.6).

3.3 Data acquisition

During the recording of a WOZ experiment all data are stored on a set of currently five Windows NT workstations. Each computer is dedicated to record a single data stream because continuous capturing of audio and, especially, of video demands a great amount of computing power. The data from two video streams (front view and side view) are recorded separately via a FireWire bus between camera and computer and are stored as Quicktime files encoded in DV. The video signal of the infrared camera is digitized by an external analog–DV converter and then recorded in the same way. Video data encoded with DV offer a high picture quality and show

[6] http://www.apple.com/quicktime/

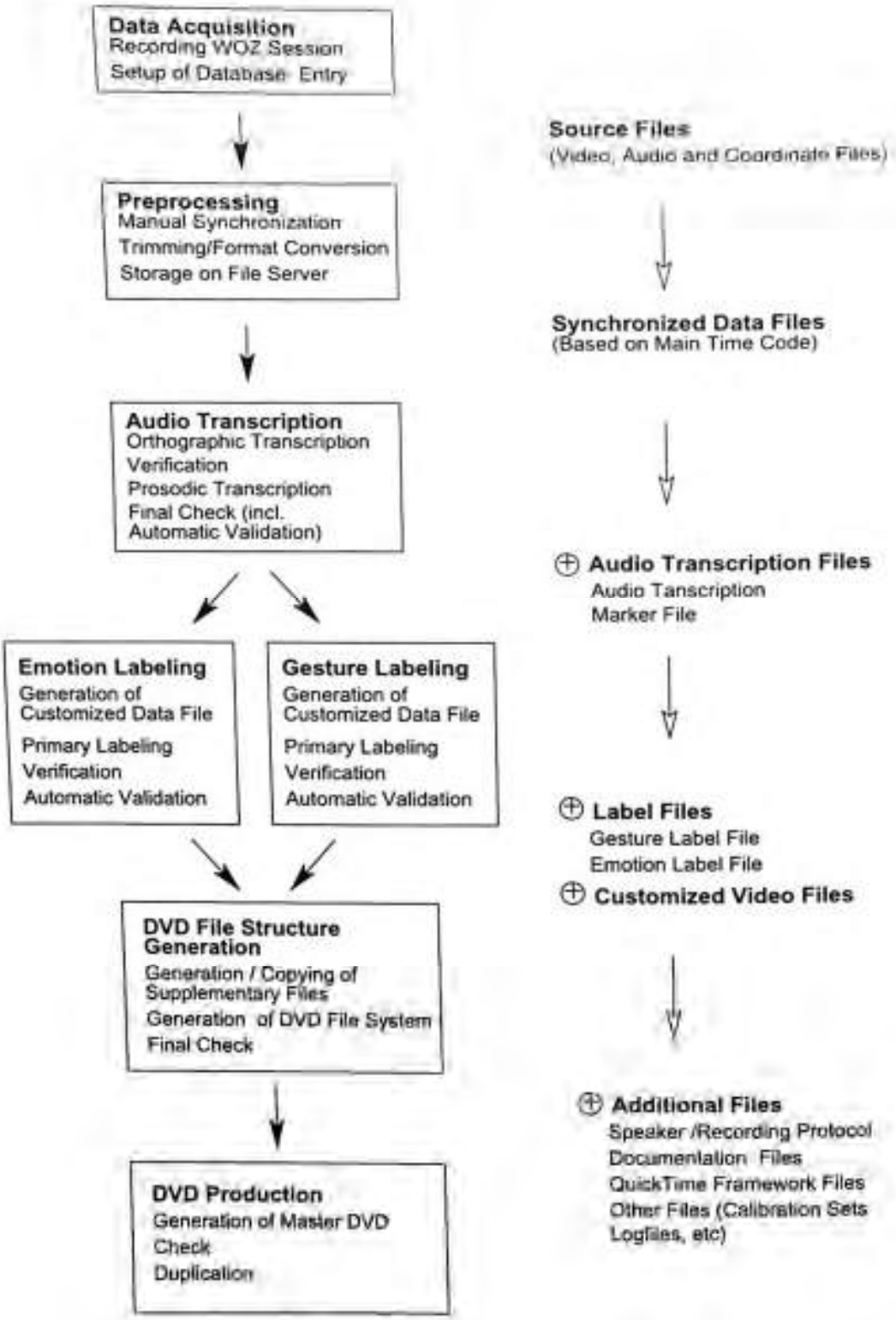


Fig. 4. Processing stages in SMARTKOM data collection (*left*) and corresponding changes in the data file set (*right*)

very few artifacts that could be otherwise problems for image processing algorithms. However, the advantage in quality comes together with large amounts of data (see Sect. 3.9 for details on storage requirements). For capturing the audio data we use a ten-track audio card. The recorded audio files are stored in Windows WAVE format, using a resolution of 16 bit and a sampling frequency of 48 kHz.

Capturing of the graphical system output is done via a screen capturing tool. It allows recording to a video file in AVI format at low frame rates; in our setup capturing at 4 fps is sufficient because the interaction speed is low. The remaining data tracks, the coordinate files of the SiViT gesture recognizer and of the graphical tablet are recorded with specially designed tools. The file format is a text file containing a list of coordinates together with the time stamps. In addition to the actual recording, this stage also comprises the generation of new entry for the session in the database. Information about the speaker as well as his or her behavior during the recording or details about the setup are stored here. In the following stages gradually more and more information is added, e.g., about the current state of the session in the processing pipeline. The database is presented in more detail in Sect. 3.9.

3.4 Preprocessing

This stage comprises all steps that are necessary to deliver a set of synchronized data files to the following processing stages. Because the recording is done on several computers, each running on its own internal clock, it is thus necessary to align all data tracks to a single master clock and to define a common start and end point.

3.4.1 Generation of "visible" coordinate log files

In a first step the coordinate data files must be made accessible for the editor in order to perform the temporal alignment. Our idea was to use the sprite track feature of Quicktime, where small bitmaps (called sprites) can be blended over another video track. The parameter of the sprites like, for example, the position or orientation can be changed dynamically. Figure 5 shows an overlay of three video tracks:

- the infrared video from the SiViT camera, showing a gray-scale picture of subject's gestures
- the video track of the graphical system output
- the visualization of the SiViT coordinates

The coordinate data is displayed by a moving dot with a cometlike tail, which gives not only a graphical representation of the current position but also of the history of coordinates at previous time steps.

3.4.2 Manual synchronizing

The actual synchronization process is done manually on a video editing workstation running Adobe Premiere.[7] We investigated several automatic or semiautomatic approaches using defined synchronizing signals that appear in all captured modalities,

[7] http://www.adobe.com/products/premiere/main.html

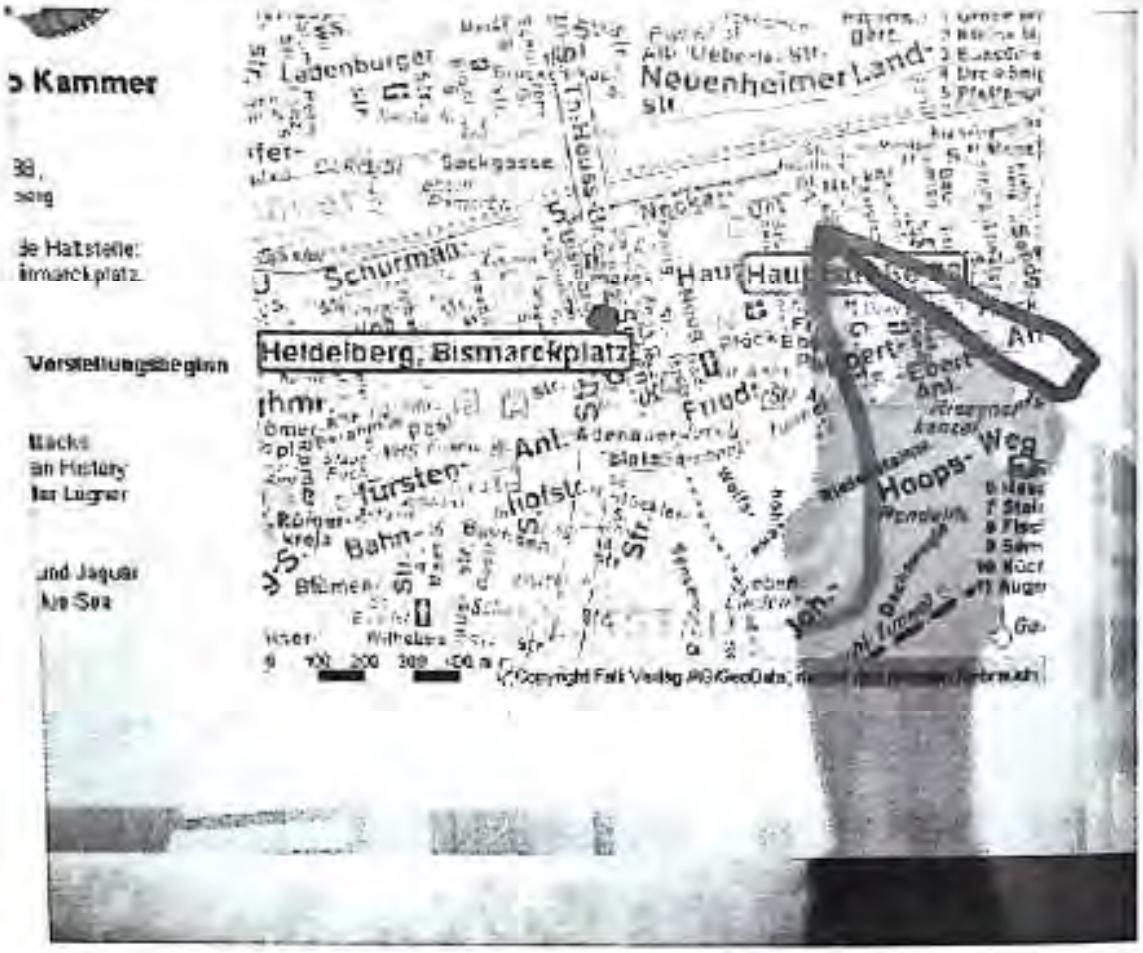


Fig. 5. Coordinate log file of the SiViT gesture recognizer visualized by cometlike data plot

but it turned out that all of these attempts resulted in a considerable number of errors so that a manual inspection of the results was necessary anyway. Therefore we decided to skip these techniques and do the synchronization manually. [8]

The process is done in two steps. First, the audio files, which are recorded separately on the audio capture computer, are aligned to the two video camera tracks. This is done by visually aligning the waveforms of the audio tracks to the waveforms of the audio tracks recorded by the cameras.

Second, the remaining video tracks are aligned by matching characteristic changes in the video tracks, e.g., a change of graphical system output or a stroke of a hand gesture. The circumstance that the test subject starts each session with a finger tap on the Web persona helps with this task. At this point we have aligned all data tracks at the beginning of the recording.

[8] Another advantage of manual synchronization is that a human operator actually looks at and hears the captured data. That way faulty data caused by malfunctioning devices, etc., can be detected early.

In a last step the start and end point of the recording is defined and all tracks are cut to equal length. At the same time, some file format conversions are done, e.g., downsampling the audio data files to 16kHz. The coordinate files are updated: the temporal shift and trimming that were performed on the visual coordinate tracks must also be considered in the raw data files.

3.4.3 Verification of the Synchronization Process

A big challenge in the SMARTKOM data collection was the ability to record different data sources and to combine them to one temporally consistent data entity. Because we had a limited budget, we had to use standard components for the recording studio. While audio hardware with synchronizing features is available, video systems that allow capturing of multiple streams can only be found in the high-end (and also high-price) market. The problem of how the other data sources like the gesture coordinate data in our project can be integrated in such a setup still remains.

Another approach would have been to synchronize the time base of the recording computers via network using, e.g., the network time protocol (NTP).[9] This would have required the design of new capturing tools in order to profit from the distributed master clock. Besides the fact that this would have caused a great delay for the start of the SMARTKOM data collection, this approach failed because of missing APIs for the video hardware.

Our current synchronization process is based on the assumption that the timing deviations of the different capturing processes stay in an acceptable range. We try to achieve this by a setup that is as homogeneous as possible. For example, the three video capture computers are equipped and configured identically.

We performed several test recordings in order to specify the timing deviations in our setup. These test recordings were limited to a length of 9 min as the size of the video files would otherwise hit the 2-GB limit for the file size given by the operating system. All data tracks were recorded and manually aligned as described in the previous section. We found an increasing timing deviation between the video tracks and the remaining tracks, which reached the maximum of one full video frame (40 ms in PAL) after 9 min. If we assume a linearly increasing drift, we end with a timing deviation of 20 ms in a typical SMARTKOM session of 4.5 min. Deviations in this range are below the range human beings normally can perceive. Dialogue systems like SMARTKOM operate with much higher timing windows when combining events from different modalities. We therefore consider the timing deviation as acceptable for this multimodal corpus.

3.4.4 Data Storage on Server

Finally, the aligned data files are stored on a Linux file server. The working teams of the following process stages access these files over the network in order to prevent repeated copying of large files. The actual path to the data is noted in the database sheet.

[9] http://www.eecis.udel.edu/ ntp/

3.5 Audio Transcription

Audio transcription is a time-consuming manual process. It includes orthographic transcription (proper names, hesitations, noise and pronunciation variants are also tagged), a verification step and prosodic transcription (primary and secondary stress, intonational movement at phrase boundaries). A detailed description of the SMART-KOM transcription system can be found in Steininger et al. (2006)

During the orthographic transcription a marker file is also generated that describes the start and end point of each turn of the HMC (segmentation). This timing information may be used as structural element. For example, it is used to display the transcription turnwise in the video stream in a subtitle style. The final transcription file is checked manually and validated automatically against the transcription conventions.

3.6 User State and Gesture Labeling

These work steps require customized video files that are rendered from the source data files on the file server. Figure 6 shows a frame of a video track used for gesture labeling. It contains four different views on the video data in one picture (front camera, side camera, graphical system output, infrared camera merged with the system output) and an audio track. In addition, the audio transcription of the current turn is inserted on the top. The file format is AVI with Cinepak encoding as this the only video format which can be used together with the labeling tool *Interact*.[10] For the emotion labeling a simplified version (only the front camera together with the audio TRL and an audio track) is produced. Both customized video tracks are created by a Java application using the Quicktime API for Java (Maremaa and Stewart, 1999).

The label files generated in these stages are stored in ASCII text files. They are later integrated in the framework Quicktime file and in the BAS Partitur Format files (Schiel et al., 2002) in the same manner as the audio TRL file. For more information on user state and gesture labeling see Steininger et al. (2006)

3.7 Final Data Preparation

In this stage all source files and annotation, resp. label files, of a session are collected, final-checked and prepared for distribution on DVD-R media. A specially designed tool provides help with the required working steps.

Supplementary files for the sessions are generated, for example, a speaker protocol file, which gives information about the characteristics of the subject or a recording protocol file describing parameters of a recording. Final checks of the source files are done. They include semiautomated checks of cross-references in the data files (e.g., same speaker ID used in speaker protocol and recording protocol) as well as manual consistency checks.

[10] http://www.mangold.de/

Fig. 6. Video stream used for gesture labeling

The framework Quicktime file is generated in the next step. It offers the possibility to choose among different views on the data files and to use these specialized views for playback of the recorded session. In the last step the DVD file structure is generated and additional files, for example, technical documents or useful tools for viewing the data, are copied.

3.8 DVD Master Production

Finally, all data are moved to the DVD-R station, an Apple Macintosh running DVD Toast by Adaptec. A master DVD is burned and verified; it then serves as source for duplication. As soon as the DVD is verified successfully and the master data on the server is archived into a *Tivoli* storage system (which is located in a different building to maximize security), all data are removed from the server.[11]

3.9 Data Logistics

The database mentioned earlier is a key element in organizing the recordings and the further processing of the data. Due to the fact that each recording makes up a data

[11] See Sect. 4.2 for details about the distribution of data.

amount of 4 GB (temporarily even up to 6.5 GB) and because up to 60 recordings are processed in parallel, the whole processing must be well planned.

The database reflects the processing scheme as indicated in Fig. 4 and gives up-to-date information about the status of each recording during its processing via a Web interface. Members of the team may check in certain tasks they perform, lock certain files while working on them and report the successful completion of a task. Furthermore, special observations or found bugs and faulty data can be reported here, session and speaker specifics entered and linked to session IDs. Finally, the database can be used as an automatic reporting generator not only for progress statistics but also for XML-rendered metadata, which are later included into the final distribution of the corpus.

In the final stage of SMARTKOM the file server offered approx. 2200 GB disc space and the processing time needed from the recording to burning the DVD was about 6 weeks. Unfortunately, these bottlenecks restrict the number of recorded test subjects to an average of 4 per week.

4 SmartKom Corpus

The final corpus was distributed continuously during the course of the project to the project partners. The Ministry of Education and Sciences (BMBF) as the principal funding source of the SMARTKOM project assigned the task of further maintenance and distribution of the corpus to the BAS. As in the former Verbmobil project, all data produced during the project phase should be freely available to interested scientists and engineers one year after the project internal release date. There are no restrictions with regard to the usage of the corpus except the redistribution to third parties. The corpus will also be available through the European Language Resources Association (ELRA).

The SMARTKOM Corpus consists of four parts:

- BAS SMARTKOM-Public (SK-P)
- BAS SMARTKOM-Home (SK-H)
- BAS SMARTKOM-Mobile (SK-M)
- BAS SMARTKOM Biometry (SK-B)

In the following we will give a short description of the first three parts containing the WOZ recordings and annotations as described above. [12]

4.1 Corpus Overview

The BAS edition of the SMARTKOM multimodal data collection contains the same data as the original project edition newly structured and validated against basic BAS guidelines. The following deviations to the original edition should be noted here:

[12] The SMARTKOM Biometry corpus will not be described here since it consists of an entirely different set of data.

- transliteration, annotations, lexicon and BAS Partitur files added
- additional documentation of formats
- revalidation against BAS standards
- partly extended to replace incomplete sessions

The size of the full corpus is approx. 1.4 TByte. It contains the recorded data of 233 users (92 male, 141 female). The basic structural unit is the recording session; the total number of sessions is 428 (P: 162, M: 139, H: 127). Each recording session contains all recorded modalities together with the annotations and metadata. The size of a single recording should be less than 4.5 GB to allow distribution on standard DVD-R. Sessions are stored on numbered DVDs (since the original corpus was distributed on DVD-R). Therefore each directory named "DVD-##.V" is the root directory to a "DVD" even if more than one DVD is stored on a medium.[13] If the corpus was distributed on a larger medium than DVD-R, there usually exist structures above the DVD directories that summarize contents in a more practical way. Please refer to the README on the top level of the medium for that.
Each session is named by a seven-character string:

```
TNNN_SD  where:   1st char
                  T = type of recording  b : Biometric data
                                         w : Wizard-of-Oz
                                         d : Demo session
                                         p : Test session
                                         e : evaluation
                                             session
```

Wizard-of-Oz is the standard case where the person is not aware that he/she is dealing with a human-controlled system.

A *demo session* is a fake recording to demonstrate the capabilities of the system; the person acts after a script.

A *test session* is a real recording but the person is cooperative and knows about the WOZ situation.

A *evaluation session* is the recording during the real usage of the prototype system either communicating with a naive person or with an expert.

Only the *Wizard-of-Oz* sessions are part of the BAS SMARTKOM edition; the remaining data are kept available at the BAS for interested parties.

```
2nd to 4th char
NNN = session number starting with 001
```

Session numbers start with 001 within each type. The numbering is not continuous; there might be missing numbers due to faulty recordings that were omitted before distribution.

```
5th char
'_'
```

[13] "##" = DVD number; "V" = DVD version number.

```
6th char
S = technical scenario p : Public
                       m : Mobile
                       h : Home

7th char
D = primary task       k : cinema
                       t : touristic
                           planning
                       f : TV guide
                       r : restaurant
                       n : navigation
                       v : VCR programing
                       m : music jukebox
                       a : phone
                       x : fax
```

In addition to the seven-character session name above, there exist the following name extensions to denote more specific parts of the session:

```
8th char : recording
           channel:     a : clip-on microphone, channel 1
                            Sennheiser KA 100 P XLR
                            placement: collar
                        b : clip-on microphone, channel 2
                            Sennheiser KA 100 P XLR
                            placement: collar
                        h : headset microphone
                            Sennheiser NB2
                        1-4 : microphone array 4 channels
                            Sennheiser KA 100 SP
                            placement: upper end of
                            interaction area
                        d : directional microphone
                            Sennheiser ME 66
                            placement: on face camera
                            distance to mouth: 60 cm
                        w : system output
                        p : playback backround noise front
                        q : playback background noise back
                        t : tableau coordinates
                        s : SiViT coordinates
                        i : infrared video of
                            interaction area
                        m : front capture camera (face)
```

```
l : left lateral capture
o : system display capture
```

Please refer to Türk (2000) for a detailed description of the technical setup.

```
9th char : '_'

10th - 12th char : turn numbering, starting with '001'

13th char : '_'

14th - 16th char : speaker ID
```

Extensions denoting turns or speaker IDs are only used to denote events in a turn of a session, e.g., within a labeling scheme. The SMARTKOM system always has the "speaker ID": "SMA".

The following extensions are used throughout the corpus:

```
.qt   QuickTime file
.trl  transliteration
.spr  speaker protocol file
.rpr  recording session protocol
.par  BAS Partitur Format file
.avi  video file AVI
.mov  video file DV
.ges  gestic labeling file
.ush  user state labeling file ('holistic')
.usm  user state labeling file ('mimic')
.trp  user state labeling file ('prosody')
```

For example, a session recording "w003_pk" consists of a set of files like what follows (when all channels are present):

Type	Name
QuickTime frame *	w003_pk.qt
Front capture *	w003_pkm.mov
Lateral capture *	w003_pkl.mov
Infrared video	w003_pki.mov
System display output	w003_pko.avi
Audio clip 1 #	w003_pka.wav
Audio clip 2 #	w003_pkb.wav
Audio headset #	w003_pkh.wav
Audio Directional #	w003_pkd.wav
Audio array 1	w003_pk1.wav
Audio array 2	w003_pk2.wav
Audio array 3	w003_pk3.wav
Audio array 4	w003_pk4.wav

Audio system output *	w003_pkw.wav
Audio background front	w003_pkp.wav
Audio background back	w003_pkq.wav
Tableau coordinates	w003_pkt.qt
SiViT coordinates	w003_pks.qt
Recording session protocol *	w003_pk.rpr
Speaker protocol file *	AAB.spr
Transliteration *	w003_pk.trl
Bas Partitur Files (BPF) *	w003_pk.par
User State Holistic $	w003_pk.ush
User State Mimic $	w003_pk.usm
User State Prosody $	w003_pk.trp
Gesture Labeling *	w003_pk.ges
Turn Segmentation *	w003_pk.mar

The obligatory files are marked with a "*". At least one of the audio channels a, b, h or d (marked with a "#") must be recorded. If the front video recording is present, the annotation files ush, usm and trp (marked with "$") must be present. If one of these files is missing, this is regarded as an error and listed in the 'Known Errors' section of the documentation.

4.2 Distribution

Because of the large size of the corpus (at least large at the time of production), the distribution of the whole corpus or even of the individual subcorpora Public, Home and Mobile cannot be done on regular DVD-5 media. Nevertheless, the BAS offers the distribution of selected sessions or of special extracts of modalities on DVD-5.

A full corpus will be distributed on IDE hard disc drives of 120 or 160 GB size each. The file system is FAT 32 to allow for easy mounting on all PC platforms. The total SMARTKOM corpus requires 11–12 hard disc drives. The data integrity of these drives is guaranteed for approx. 5 years. The end user should therefore copy the received data as soon as possible on a file server (with a backup facility) or — as an inexpensive alternative — insert the original drives into a server system.[14] The user may also copy parts of the corpus to DVD-5 at his/her facilities; to simplify this we structured all BAS SMARTKOM corpora into chunks smaller than 4.5 GB.

All symbolic information to the corpus, that is, all metadata, documentation and annotations as described below, are available on the BAS FTP server for free. There you will always find the latest version of these data. Since it is very likely that during the usage of the corpus some errors occur, the BAS strongly encourages the users of the SMARTKOM corpus to report these errors back to the BAS. Bug-fix messages will then be automatically sent to all known users, and the fixed data can be downloaded from the BAS servers.

[14] The data integrity lasts much longer if using the drives in a life server system rather than just storing the drives in an archive.

4.3 Signal Formats

Most of the file formats used in this corpus are widely used standards. The top QuickTime (QT) file has the extension ".qt" and summarizes all recorded data into one synchronized data stream. Individual streams are stored in standard format files (e.g., *.wav, *.mov, *.avi), which can be also accessed independently from the top QT file. The current QT player allows only one video track to be activated. You may switch from different views during playback. The same is true for the nine audio tracks. As an additional feature, the QT file contains specially compiled versions of the transliteration to display the spoken text in a subtitle frame. A chapter track allows users to jump to selected turns of the recorded dialogue. Each distributed medium of the corpus contains the publicly available QT library and player for Mac OS and MS platforms.

4.4 Metadata

For each recording session and each SMARTKOM user there exist metadata description files.

4.4.1 Recording Protocols

The recording protocols list all relevant parameters to a recording session and are stored in the file "TNNN_SD.rpr", where "TNNN_SD" denotes the session ID. The recording protocol is stored in XML and is mostly self-explanatory (English tag set). A DTD for the format can be found in the online documentation.

For example (excerpt only):

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE RPR SYSTEM "rpr.dtd">
<RPR>

  <HEAD>
    <version number="2" subnumber="0" />
  </HEAD>

  <DVD no="163" />

  <SPEAKER id="ADB" />
  <SESSION-PARAMETERS>
    <session_id value="w160_mt" />
    <recorded_domains>
      <domain_planned use-case="Touristik" />
      <domain_recorded use-case="Touristik" />
    </recorded_domains>
    <atmosphere place="mobile" number="1" volume="soft" />
    <background pattern="Bunt1" />
    <pen mode="pen" />
```

```
<emotions evoked="no" />
<content_variation version="1" />
<recording_date year="2001" month="09" day="23" />
<recording_location value="LMU" />
<recording_setup wizard="wizard3" />
<experimenter name="Silke Steininger" />
<wizard_speech_output name="Ulrich Reubold" distortion="soft
 distortion" />
<wizard_navigation name="Ulrich Reubold" />
<session_sequence_no position="2" />
</SESSION-PARAMETERS>

<DATA-TRACKS>
  <VIDEO>
    <Data fieldname="(m)_mimic" device="Panasonic NV DX 100 EG"
     present="yes" />
```

4.4.2 Speaker Profile

Speaker profiles are stored in the file "AAA.spr", where "AAA" denotes the three-character speaker ID. The speaker ID to a given session can be either obtained from the corresponding recording protocol, from the standardized turn marker or from the header of the corresponding BAS Partitur Format file. The speaker profile is stored in XML and is mostly self-explanatory. A DTD for the format can be found in the online documentation.

For example:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!DOCTYPE SPR SYSTEM "spr.dtd">
<SPR>

  <HEAD>
    <version number="2" subnumber="0" />
  </HEAD>

  <SPEAKER>
    <Personal_Data speaker-id="AAJ" sex="F"
     date_of_birth="13.07.1940"
     height="164  cm" weight="60  kg" handed="right" />

    <School degree_state="BY (Bayern)"
     degree="Volksschule/Hauptschule"
     profession="Rentnerin/Rechtsanwaltsgehilfin" />

    <Languages mothertongue="DEU" mothertongue_mother="DEU"
      mothertongue_father="DEU" dialect="I3 (S\"udbairisch)"
      bilingual="no">
```

```
      <foreign_languages>
        <language value="ITA" />
      </foreign_languages>
      <bilingual_languages>
        <language value="none" />
      </bilingual_languages>
    </Languages>

    <Culture german_nationality="yes"
     cultural_environment="DEU" />

    <Experience speech_singing_training="no"
      computer_experience="yes"
      speech_dialogue_experience="no" />

  </SPEAKER>

  <RECORDING-SPECIFIC>
     <glasses exists="yes" />
     <smoker exists="yes" />
     <beard exists="no" />
     <piercing exists="no" />
     <jewels exists="yes" />
  </RECORDING-SPECIFIC>

  <COMMENTS>
VPN war zweimal da! Zweite Session: w335_mt und w336_mt
  </COMMENTS>

</SPR>
```

4.5 Annotation Formats

The SMARTKOM corpus contains several annotation layers: transliteration, 2D gesture, user state holistic, user state face and prosodic segmentation. Please refer to the contribution by Steininger et al. (2006) for a more detailed discussion of the problem on how to annotate multimodal data. In the following we will merely give a short formal description of the formats used for the different annotation types.

4.5.1 Transliteration

Input: audio channel

This file type contains the orthographic transliteration of spontaneous spoken language. That includes, for instance, information like interruptions within phrases, revision and repetition of utterances, reduction or hesitation. There are symbols for the following categories:

1. lexical units
2. syntactical–semantical formation
3. nonverbal articulatorical utterances
4. noise
5. pauses
6. acoustical interaction (speech overlap)
7. comments
8. special comments
9. prosodies

The audible events are recorded orthographically only. There is no phonological transliteration, no phonetical transcription and no temporal correlation with the signal.[15]

The format is based on the well-known Verbmobil II transliteration format (TR2) extended by markers for prosodic event, for barge-in, for off-talk as well as some methodological adaptations. Therefore this type of transliteration is denoted by "TRS" (transliteration SMARTKOM) in the BAS Partitur Format to distinguish it from other formats.

A detailed description of the conventions can be found at our website.[16] The transliteration is stored in a file with extension *.trl with one turn per line.

4.5.2 2D Gestures

Input: audio channel, facial video, lateral video, system output overlapped with infrared video of interaction area

This file type contains information on gestures that occurred in the session. All gestures that occur within the borders of the SiViT camera field are labeled. Additionally, emotional gestures that occur elsewhere are labeled. The following label information is assigned to each found gesture:

- begin of the gesture (onset) in samples
- duration of the gesture in samples
- category of the gesture
- label of the gesture
- hand and finger used
- reference word
- reference zone on display
- reference object
- begin of the stroke in samples
- duration of the stroke in samples
- optional: comments about the gesture

The data fields are tab-separated and stored in a BPF tier file with extension *.ges.

[15] However, a temporal correlation can be obtained by using the segmental information in the MAR files (see below) or by using the information in the TRN tier, respectively.

[16] http://www.phonetik.uni-muenchen.de/Forschung/SmartKom/Konengl/engltrans

4.5.3 Turn Segmentation

Input: audio channel

Since all recording contain the data of a complete dialogue of approx. 4.5-min length, it is useful to segment this data stream into shorter units called turns. The turn is defined in the transliteration conventions to SMARTKOM (see our website [17]). This information is stored in so-called "marker files" (*.mar).

Marker files contain the information when a turn of the dialogue starts and ends in the signal files as used in the transliteration. Each line contains three entries separated by white spaces:

```
Start of turn          End of turn          Turn marker
```

Start and end are given in samples (16 kHz) from the beginning of the file (0). The turn marker reads as follows: XXX_A_B, where "XXX" is the speaker ID; "A" is the turn number and "B" is the channel number (channel 1 is human; channel 2 is machine). For example:

```
47232 187264 SMA_1_2
205312 249472 ULT_2_1
...
2170240 2190080 SMA_19_2
```

4.5.4 Prosodic Labeling for User State

Input: audio channel

The metalinguistic prosodic annotation is part of the so-called user state annotation. User states are positive or negative attitudes of users towards a certain system in human-computer interaction. The labeled prosodic events are part of the input to the user state classifier that works exclusively on the audio signal.

The files *.trp contain the orthographic transliteration of spontaneous spoken language as described above modified in the following way:

1. Elimination of the following tags:
   - linguistic prosodic features (labels enclosed in '[]')
   - noise (tags beginning with '#' or '<#>' or bracketing with those)
   - pauses ('<P>')
   - overlapped speech (word modifiers '...1@' or '@1...')
   - special comments ('<;...>')
   - tildes for marking names ('~..')
   - words hard to understand ('...%')
2. Added the annotation of the following metalinguistic prosodic features
   - PAUSE_PHRASE: irregular pause on a phrasal level

[17] http://www.phonetik.uni-muenchen.de/Forschung/SmartKom/Konengl/engltrans

- PAUSE_WORD: irregular pause on a word unit level
- PAUSE_SYLL: irregular pause on a syllable level
- LENGTH_SYLL: lengthening of a syllable
- EMPHASIS: emphatic accentuation
- STRONG_EMPH: stronger emphatic accentuation
- CLEAR_ART: clear articulated speech
- HYPER_ART: hyperarticulated speech
- LAUGHTER: speech overlapped by laughter, sigh or the like

More than one label per word is allowed and a feature occurring at several adjacent words is marked at each word by a separate label.

3. Added information about durations in ms, surrounded by curved brackets: {%snd: SOUNDFILENAME_begin_end}, where %snd stands for "sound" and _begin_end are given in msec from the beginning of the sound file SOUNDFILENAME. This segmental information is used in the following way, depending on the annotated label:
   - PAUSE_PHRASE, PAUSE_WORD: duration of the word before the pause plus the pause duration
   - PAUSE_SYLL: duration of the word with a pause between its syllables
   - LENGTH_SYLL: duration of the word with a lengthened syllable
   - EMPHASIS: duration of the word with an emphatic accentuation
   - STRONG_EMPH: duration of the word with a stronger emphatic accentuation
   - CLEAR_ART: duration of the clear articulated word
   - HYPER_ART: duration of the hyperarticulated word
   - LAUGHTER: duration of the overlapped word

The various labels are placed after the word that reveals the annotated phenomena; of course, pause label are placed between the relevant units. See Steininger et al. (2000) for more detailed information about this annotation.

4.5.5 Holistic Labeling of User State

Input: facial video, audio channel

The file type *.ush contains an segmentation and labeling of user-states (interesting emotional and cognitive states) that occurred in the session. The whole session is segmented that is there are no gaps within the segmentation.

For each segment begin (second column) and duration (third column) are given in samples (16 kHz) starting from the begin of recording (0). Each segment is assigned to one of the seven labels (third column) below (label names in square brackets):

- neutral [Neutral]
- joy/gratification (being successful) [Freude/Erfolg]
- anger/irritation [Ärger/Mißerfolg]
- helplessness [Ratlosigkeit]
- pondering/reflecting [Überlegen/Nachdenken]

- surprise [Überraschung/Verwunderung]
- unidentifiable episodes [Restklasse]

The labels are assigned with respect to the impression of the labeler. Not only the facial expression but also the voice quality or other contextual information is considered. Only the use of words with emotional content, but without an emotional expression is *not* considered as an indicator of a respective emotion/user state.

The intensity of a user state is labeled in the forth column for the classes 2–6 by:

- strong [stark]
- weak [schwach]

In parallel to the user-state labels the *.ush files also contain a segmentation of occlusions when the hand or pen occlude parts of the face. This segmentation is not without gaps and is overlapped to the segmentation of user states: [18]

- hand over face [Hand im Gesicht]
- hand over face/mouth [Hand im Gesicht/Mund]
- hand over face/nose [Hand im Gesicht/Nase]
- hand over face/eyes [Hand im Gesicht/Augen]
- pen over face [Stift im Gesicht]
- pen over face/mouth [Stift im Gesicht/Mund]
- pen over face/nose [Stift im Gesicht/Nase]
- pen over face/eyes [Stift im Gesicht/Augen]
- face partly not in the range of recording [Teilweise nicht im Bild]
- other object in face [Objekt im Gesicht]

For more detailed information please refer to Steininger et al. (2002a), Steininger et al. (2002b) or Steininger et al. (2000).

4.5.6 Mimic Labeling of User State

Input: facial video, USM annotation without labels

The file type *.usm contains information on user states that were segmented and labeled *without the audio channel*.

The file structure and the labels used are identical to the *.ush annotation described before. Only the labeling procedure differs: A different labeler watches the video face recording again and assigns new names to the segments. The information that is available for the segmenter from the previous USH labeling process are:

- the neutral segments
- the segments with occlusions of the face
- the time information but not the labels of all other user state events

[18] In the BAS Partitur Format (Schiel et al., 2002) that summarizes all annotations in a common format this segmentation has a different tier name (OCC) than the user state segmentation (USH).

The segmenter is advised to ignore the neutral and occlusion segments and to label the remaining user-state segments anew. He may also assign the borders of the segments anew (if applicable).

Acknowledgments

The work described here was carried out by a rather large group of scientists and students at the Ludwig-Maximilians-Universität München stemming from a variety of scientific fields such as electrical engineering, psychology, phonetics, computer linguistics, computer science, phonology, choreography, and music. Specifically, we would like to thank Silke Steininger, Nicole Beringer, Susan Rabold, Olga Dioubina, Daniel Sonntag, Ute Kartal, Hannes Mögele, and approx. 40 students for their valuable contributions to the SMARTKOM Corpus.

References

N. Beringer, V. Penide-Lopez, K. Louka, M. Neumayer, U. Türk, and C. Grieger. Wizard-of-Oz Display. Technical Document SmartKom 12, Bavarian Archive for Speech Signals (BAS), 2000.

T. Maremaa and W. Stewart. *QuickTime for Java: A Developer Reference*. Morgan Kaufmann, San Francisco, CA, 1999.

F. Schiel, S. Steininger, N. Beringer, U. Türk, and S. Rabold. Integration of Multi-Modal Data and Annotations Into A Simple Extendable Form: The Extension of the BAS Partitur Format. In: *Proc. Workshop "Multimodal Resources and Multimodal Systems Evaluation"*, pp. 39–44, Las Palmas, Spain, 2002.

S. Steininger, O. Dioubina, R. Siepmann, C. Beiras-Cunqueiro, and A. Glesner. Labeling von User-State im Mensch-Maschine Dialog — User-State-Kodierkonventionen SmartKom. Technical Document SmartKom 17, Bavarian Archive for Speech Signals (BAS), 2000.

S. Steininger and U. Kartal. Aufnahmeplan Wizard-of-Oz Aufnahmen SmartKom. Technical Document SmartKom 5, Bavarian Archive for Speech Signals (BAS), 2000.

S. Steininger, F. Schiel, O. Dioubina, and S. Rabold. Development of User-State Conventions for the Multimodal Corpus in SmartKom. In: *Proc. Workshop "Multimodal Resources and Multimodal Systems Evaluation"*, pp. 33–37, Las Palmas, Spain, 2002a.

S. Steininger, F. Schiel, and A. Glesner. Labeling Procedures for the Multi-Modal Data Collection of SmartKom. In: *Proc. 3rd Int. Conf. on Language Resources and Evaluation (LREC 2002)*, pp. 371–377, Las Palmas, Spain, 2002b.

S. Steininger, F. Schiel, and S. Rabold. Annotation of Multimodal Data, 2006. In this volume.

U. Türk. Technisches Setup bei den SmartKom-Aufnahmen. Technical Document SmartKom 7, Bavarian Archive for Speech Signals (BAS), 2000.

Annotation of Multimodal Data

Silke Steininger, Florian Schiel, and Susen Rabold

Bavarian Archive for Speech Signals
c/o Institut für Phonetik und Sprachliche Kommunikation
Ludwig-Maximilians-Universität München, Germany
{steins,schiel,rabold}@bas.uni-muenchen.de

Summary. Do users show emotions and gestures if they interact with a rather intelligent multimodal dialogue system? And if they do, what do the "emotions" and the gestures look like? Are there any features that can be exploited for their automatic detection? And finally, which language do they use when interacting with a multimodal system — does it differ from the usage of language with a monomodal dialogue system that can only understand speech?

To answer these questions, data had to be collected, labeled and analyzed. This chapter deals with the second step, the transliteration and the labeling.
The three main labeling steps are covered: orthographic transliteration, labeling of user states, labeling of gestures. Each step will be described with theoretical and developmental background, an overview of the label categories, and some practical advice for readers who are themselves in the process of looking for or assembling a coding system. Readers who are interested in using the presented labeling schemes should refer to the cited literature — not all details necessary for actually using the different systems are presented here for reasons of space. For information on the corpus itself, please refer to Schiel and Türk (2006).

1 Transliteration of Speech

The transliteration of spontaneous speech is a tricky problem. Spontaneously speaking people tend to produce a lot of events that are not actually speech: Looking at a transcript one sometimes gets the impression that a speaker actually produces more nonlanguage events than actual language events. One can assume that the ratio of well-pronounced speech to reduced or unconventional speech and nonspeech events is different for different situations of spontaneous speech. For example, a person giving a talk pays much more attention to proper pronunciation and comprehensibility and tries to suppress coughs, hesitations and similar events. A person who is talking rapidly to a friend, perhaps joking, perhaps even eating, simultaneously represents the other extreme: One can expect a lot of neologisms, reductions, interruptions, etc. The same will be the case for emotional speech. The situation in SMARTKOM is a case in-between the two of these extremes: The speaker will probably only concentrate to produce well-pronounced speech in a small part of the dialogue and show

lots of the events that are characteristic for everyday, spontaneous speech. A transcription system has to be able to capture all the peculiarities of such casual speech. Fortunately, spontaneous speech has been transliterated for many years, therefore much experience about the to-be-expected problems has been accumulated and a wide variety of transcription systems is available.

We decided to use the Verbmobil conventions to be consistent with the Verbmobil corpus and because our requirements were very similar to the ones in Verbmobil. Thus we adapted the Verbmobil conventions, most notably was the addition of prosodic labels. For the complete conventions please refer to our website [1] or the German version in Beringer et al. (2000).

2 Labeling of the User States

2.1 Overview User-State Labeling

This section covers the most interesting points of the user-state labeling in SMARTKOM. The labeling process comprised three separate procedures: holistic labeling of the data, labeling of the facial expression without audio information and prosodic annotation. The first two steps were specially designed for the requirements in SMARTKOM. The third step was added by request of one of the developer groups and comprised a coding system that was used in Verbmobil. First, we explain which requirements we wanted the coding systems to fulfill and how these were translated into the coding conventions for holistic and facial expression labeling. After that the basic principles and categories of these two steps are outlined. Problems and observations that could prove useful to future emotion-labeling endeavors are discussed afterwards. A brief description of the prosodic annotation concludes the section.

2.2 Requirements of the User-State Labeling

The labeling of emotions or user states[2] in SMARTKOM had to serve two main goals:

1. the training of recognizers
2. the gathering of information on how users interact with a multimodal dialogue system and which user states occur during such an interaction

The questions that have to be solved to detect user states automatically are: Which features of the face and of the voice contribute to an emotional impression? Which of these features can be detected automatically?

If we already knew the answers to these questions, it would make sense to define coding conventions that mark these features in the data (for example, with a formal or structural coding system that labels morphological features), compare FACS by

[1] http://www.phonetik.uni-muenchen.de/Forschung/SmartKom/Konengl/engltrans

[2] The name "emotion labeling" was changed to "user-state labeling" because the targeted episodes in the data comprised not only emotional but also cognitive states.

Ekman and Friesen (1978). But since we are far from answering these questions conclusively, we had to work the other way round.

Our "other way round" technique amounted to the following: We decided to define the user states with regard to the subjective impression that a human communication partner would have, if he would be in place of the SMARTKOM system. A human in a conversation with another human is able to judge which emotion or user state his communication partner shows. Of course, he does not know which emotion is "truly" present in his communication partner. But this is not necessary for him to use the emotions he perceives as clues for managing the dialogue: going on if his partner seems to be in a good mood, stopping and intervening if his partner shows signs of a negative mood and offering help if his partner seems confused. A misjudgment would not be seen as problematic (in most cases), but ignoring an expressed (negative) emotion in many cases would lead to irritation. If a communication partner therefore can detect user-states that are relevant for the communication process — then a labeler should be able to do the same — and the user states that he detects would be the ones that are relevant for the system. It would not be too bad if the detected user state was not present (it would only be bad if this happened too often). Missing a user state (with a function in the dialogue), on the other hand, would be less desirable. Our first requirement therefore was:

- The label system should target user states that were perceived subjectively and were regarded as important for the dialogue. This is a functional[3] approach; we look for the user states that serve a certain function in the dialogue.

In addition to the task of marking the relevant user states to fulfill the two goals mentioned above, the coding system had to satisfy some additional, mostly formal requirements. They overlap to a large part with the requirements for transliteration and gesture labeling:

- The labels should be selective,
- The coding system should be fast and easy to use.
- The resulting label file should facilitate automatic processing (a consistent file structure, consistent coding, nonambiguous symbols, ASCII, parsability) and preferably should be easy to read.
- The main categories and most of the modifiers should be realized as codes and not as annotations, in order to heighten consistency. Annotations (free comments and descriptions that do not follow a strict rule) are more flexible, but codes (predefined labels from a fixed set) increase the conformity between labelers.

These considerations were the starting point for the development of the coding systems for holistic and facial expression labeling, which will be outlined in the next subsection.

[3] "Functional code" or "functional unit" is sometimes defined differently by different authors. We use the term in accordance with Faßnacht (1979) for a unit that is defined with regard to its effect or its context.

2.3 Development of the Coding System for the User States

We had no data on emotions that could be expected in a human-machine dialogue, therefore we picked those that seemed interesting or relevant (for us or for the developers): "anger/irritation," "boredom/lack of interest," "joy/gratification (being successful)," "surprise/amazement," "neutral/anything else." A selection of sessions was labeled with these categories. After the labeling the categories were discussed. "Boredom/lack of interest" was excluded because it could not be distinguished from "neutral." "Neutral" and "anything else" were separated into two different categories because many sequences were found where the users definitely did not show a neutral expression, but no meaningful label could be given. "Anything else" was renamed an "unidentifiable episode." Two new categories were included to describe user states that occurred quite often in the data and are important in the context of human-computer interaction: "helplessness" and "pondering/reflecting." In a second step with the new (final) set of categories all sessions from the pretest were labeled again, and after that all other sessions. Since the goal was to find detectable features, for each category some morphological characteristics were listed. Please note that these descriptions were assembled relatively late in the process of labeling the data. We started in each category only with the "simple" instruction to the labeler: "Mark a sequence as 'joy/gratification' if you have the impression that the subject is in a positive mood, enjoys himself, is visibly content, amused or something similar." Only after a considerable amount of data was labeled were the formal criterions were extracted and written down. The goal was to enhance the consistency of the labeling system but at the same time to be true to the goal to label the subjective impressions, not existing formal categories.

With the holistic labeling system we were relatively sure to be able to label all relevant user-state episodes. However, the developers criticized that the coding system was not very well suited for generating training material for user-state recognition of facial expressions. Because of the holistic approach, the labels included not only information from the facial expression, but also information from the voice and from the context. This was seen as problematic since a facial expression recognizer obtains information only from the facial expressions and a prosody recognizer obtains information only from the voice. Therefore, we included two additional labeling steps: labeling of the facial expression without the audio information and prosodic labeling without the video channel.

For the facial expression labeling a different labeler group watched the videos without audio. The labelers started with a presegmented file (from the holistic labeling). This presegmentation was derived from the holistic labeling where the names of the categories (apart form "neutral") were deleted, while the borders were retained. We decided to start with the holistic step because humans perceive emotions in a holistic way. Not only do we include visual, auditory and semantic information, we even add assumptions about the probable emotional state of our communication partner and then add these assumptions to our perceptions (Ratner, 1989). For this reason it can be assumed that the judgments about perceived emotions would differ in cases where we have a more natural, holistic situation (watching a video, although

in a video, of course, some cues from a real-life situation are missing) and a more unnatural, deprived situation, such as watching only the face without audio (Wallbott, 1988; Munn, 1940; Sherman, 1927). Since we did not use a formal system, but one that had the goal to label subjective categories, the labelers could not be trained to look only for the cues for the category (concentrate only on certain movements, for example). Their task was a holistic one: to label a "positive mood" or a "negative mood." Therefore we assumed that their emotion detection and labeling ability would decrease in a situation with as little information as possible. By putting the holistic step before the nonauditory step we hoped that subtle episodes would not be missed.

Since the facial expression crew had presegmented label material, they knew there should be something and could verify if they actually could decide upon an emotion or not. In some of the cases, they could not perceive an emotion in the segment delivered from the holistic crew. In these cases they recorded "neutral." Conceivably these were the cases where the emotional information was only in the voice or in the context, or the emotion was too weak to be visible in a context-deprived situation. The results of the two steps were linked – the facial expression crew did not start from scratch but got the hint that there "could be something." This generated an expectation that could lead to the perception of an emotion where none would have been seen without this expectation. We decided to rather live with these errors rather than to have a wholly independent video-only labeling step because of practical considerations (the time would have doubled for the user-state labeling), but also for theoretical ones:

1. As already mentioned, without context information judgments about emotion can be blurred (see references above).
2. Because of a low inter-rater reliability for emotional judgments, two wholly independent steps would perhaps have had results that differed so much that no conclusions would have been possible about what is different between holistic and purely visual labeling.

We therefore chose an intermediate step and hoped to get some clues about which material is better suited for training and how the emotional judgments differed with regard to the input information.

Additionally, on request of one of the developer groups, we adopted a formal coding system that was used in Verbmobil (Fischer, 1999) and changed it to suit our needs in SMARTKOM. The labels were chosen according to the requirements for the user-state recognition group in SMARTKOM and were thought to represent prosodic features that are indicative of emotional speech. By the comparison between the holistic labeling and the prosodic labeling we hoped to detect relevant user states in speech. For more information on the usage of prosodic features as indicators of emotional speech, please refer to Batliner et al. (2000). For more information on the development of the user-state label systems, please refer to Steininger et al. (2002a). In the following sections the label procedures for the three different steps are described briefly together with difficulties and proposals for future research.

Fig. 1. Example of the front view that was used for the holistic and the facial expression labeling. The picture was taken from an episode that was labeled as "joy/success" in the holistic labeling step

2.4 Holistic Labeling of User States

The first step during user-state labeling was the so called "holistic labeling". A labeler watched the video of a session and marked each change in the state of the user. The segments that were found in this way were then assigned one of seven labels:

- neutral [Neutral]
- joy/gratification (being successful) [Freude/Erfolg]
- anger/irritation [Ärger/Mißerfolg]
- helplessness [Ratlosigkeit]
- pondering/reflecting [Überlegen/Nachdenken]
- surprise [Überraschung/Verwunderung]
- unidentifiable episodes [Restklasse]

The allocation of the labels was done with regard to the overall, subjective impression and took into account the facial expression, the quality of the voice, the choice of words and the context. However, the sole usage of anger-implicating words that were uttered without any emotional expression were not taken as an indicator for anger.

Additionally the label was given a rating with regard to the intensity of the user-state ("weak" or "strong"). This rating was not absolute, but referred to the general strength of emotion a certain subject showed. An individualized referencing like this poses problems for the training of recognizers. However, there is no way to define weak and strong emotions absolutely. Additionally the impression of weak and strong is dependent on the observer, and he or she also has no absolute sense for the level of a perceived emotion. This has to be taken into account when interpreting results that refer to the weak and strong differentiation. Sequences during which

Fig. 2. Example for an episode that was labeled as "anger/irritation" in the holistic labeling step

the face was partly occluded by the hand/s of the subject were marked, as well as sequences during which the face could be seen only partly in the camera picture. Consistency was enhanced by two correction steps, one of them a final correction by the same corrector for every session. Difficult episodes were discussed.

2.5 Facial Expression Labeling of User States

After the holistic labeling a second labeler group (that was not involved in the holistic labeling) watched the same video without the audio information. The group used the same categories as the holistic crew (without the auditory features). As described above, the labeler was informed about the segment borders and the occurrence of "neutral" segments, which he or she could ignore. Because of the different perception situation the labelers sometimes had a different impression about the borders of an emotional segment. In these cases they could adjust the markers for beginning and end of a segment.

2.6 Categories of Holistic and Facial User-State Labeling

Please note that the categories in the two steps were the same, but in the facial expression labeling the descriptions for features in the voice were not used.

Joy/Gratification (Being Successful)

This label was given if the labeler had the impression that the user was in a positive mood, enjoyed him- or herself, was visibly content, amused or something similar. Amusement that (obviously or probably) stemmed from derision, mockery or

Fig. 3. Example for an episode that was labeled as "helplessness" in the holistic labeling step

something similar was still labeled as joy. We made this decision because sarcasm, derision, etc., are exceedingly hard to detect reliably automatically.

Anger/Irritation

The labeler marked this label if he had the impression that the user was in a negative mood, was visibly not content, was irritated, annoyed, exasperated, angry, disappointed or something similar. Most cases of this category were weak: Full-blown anger almost never showed up. In many cases users showed their anger rather "politely" (sometimes even smiling!).

Helplessness

This label was chosen if the labeler had the impression that the user was helpless, confused or interrogative. The strong cases comprised helplessness and distress: The user did not know what to do and/or had no idea how to go on. The weak cases comprised episodes in the dialogue where the user was puzzled, a bit confused, wanted to know something or had no concrete plan how to go on.

Pondering/Reflecting

If the labeler had the impression that the user was thinking hard, "pondering/reflecting" was chosen. The task of the subject led to the fact that for the most parts of the session he watched the display, concentrated, read or searched the display. This behavior was not labeled as "pondering/reflecting". There had to be visible or audible indicators, like biting the lips.

Fig. 4. Example for an episode that was labeled as "pondering/reflecting" in the holistic labeling step

Surprise

This label was given if the labeler had the impression that the user was surprised. There was a fast movement in the face (in most cases of the eyebrows) or an abrupt vocal reaction in response to an external stimulus.

Neutral

If no emotional or cognitive state in the face or the voice could be detected by the labeler, "neutral" was marked, as well as in cases where an emotional or cognitive state was so faint that the assignment of a label seemed inappropriate.

Unidentifiable Episodes

This label was chosen if the user was neither "neutral" nor any of the other labels could be assigned to the episode. Three cases could be discriminated:

1. grimaces with no emotional content, for example, playing with the tongue in the cheek, twitching muscles, etc. (about 65%)
2. emotional sequences that had no label in our system, for example, "disgust," (about 5%)
3. states that seemed to have an emotional or cognitive meaning, but could not be decided upon by the labelers (about 30%)

For additional information on the user-state labeling please refer to Steininger et al. (2002b). A complete description of the conventions can by found in Steininger et al. (2000).

2.7 Inter-Rater Agreement in Holistic User-State Labeling

To work out the interlabeler agreement, 20 dialogues were labeled twice by different, nonoverlapping groups of labelers. The accuracy and correctness values were calculated as were the single label agreements. The dialogues for the interlabeler agreement probe were chosen by chance. The first labeling was done in the years 2001 and 2002. It took place during the normal processing of the data of the Wizard-of-Oz recordings. The second labeling was done especially for the calculation of the interlabeler agreement. It took place in 2003. The procedure was exactly the same as in the first labeling with the exception that no end correction was done. This was because only one person was responsible for the end correction during normal labeling to keep the consistency high, and this person could not take part in the interlabeler agreement study without influencing the results. All labelers were experienced, with the exception of some of the labelers in the second labeling, who had only medium labeling experience.

Within each category the labels of 20 files (labeling A) were compared to the corresponding labels of their 20 equivalent files (labeling B). Since the labeling was not a discrete process but the segmentation of a continuous data stream, labels from the two files were matched with regard of their time range. The possible hits and mistakes were:

Label A matched label B, same label:	correct
Label A matched label B, different label:	substitution
There was no match for label A in the B-file:	deletion (missing label)
There was no match for label B in the A-file:	insertion (additional label)

The third case could happen if, for example, labeler B marked one long label and labeler A marked the first part of the long label, but inserted a short label afterwards. If the two longer parts matched (with regard to time), they were compared. For the short label a "deletion" was marked, since labeler A perceived a new user state, whereas labeler B did not:

A-File XXXXXX YYY ZZZZZZZZ
B-File XXXXXXXX ZZZZZZZZZZ

The fourth case could happen, for example, if labeler A marked a long label and labeler B marked a label that matched the long A label (with respect to time) but ended earlier as the label in A. Labeler B then marked a short label that had no match (with respect to time) to a label in file A. In this case an "insertion" was marked:

A-File XXXXXXXX ZZZZZZZZZZ
B-File XXXXXX YYY ZZZZZZZZ

Note that a "deletion" during the comparison of file A with file B would be counted as an "insertion" during the comparison of file B with file A. Since neither file could be regarded as the "correct" file because no external reference existed, deletions and insertions actually described the same case: One labeler marked an additional label

that the other labeler overlooked or did not deem strong enough to warrant an extra label.[4]

The percentage of correctness was based on the number of correct labels divided by the total number of labels, multiplied by one hundred. The percentage of accuracy was based on the total number of labels minus substitutions, minus deletions, minus insertions divided by the total number of labels and multiplied by hundred.

The reliability turned up to be not very good: 55% matches were correct and the accuracy was 1.95%. This was caused by twice as many insertions/deletions and substitutions as concordant labels. The large number of insertions/deletions came from the fact that the second label group set 763 labels in 20 dialogues, whereas the other group set only 494 labels to categorize the user states within the same dialogues. When matching segmentally more than one fourth of the segments were judged differently in the two labelings. The categories "joy/gratification" and "pondering/reflecting" had the largest number of concordant user state labels. A comparison of only the labels that were rated "strong"[5] showed an accuracy of 36.6%, with 51% correct matches. This much higher rate was caused by a similar number of assigned labels during labeling A and labeling B (37 vs. 46) and by a higher number of label matches related to substitutions, insertions and deletions. Again, "joy/gratification" and "pondering/reflecting" showed the highest concordance. For detailed information about the inter-rater analysis, please refer to Jacobi and Schiel (2003).

Correct matches differed strongly with respect to label:

neutral	41%
joy/gratification	47%
anger/irritation	14%
helplessness	25%
pondering/reflecting	34%
surprise	8%
unidentifiable episodes	20%

The results of the inter-rater agreement will be discussed after the next section, which describes the inter-rater agreement results for the nonauditory user-state labeling.

2.8 Inter-Rater Agreement Facial User-State Labeling

The procedure for the analysis of the inter-rater agreement for the labeling without audio information was exactly the same as for the holistic labeling (see above). The results were much better than for the holistic labeling group: 59% of the labels were correct and the accuracy was 45.4%. The concordant labeling of "neutral" segments was not included in the calculation because they were in large part predetermined by

[4] Both files were compared to each other, and the correctness and accuracy were computed for both cases and then divided by two.

[5] A label was chosen for the comparison if the rating "strong" was given in at least one of the labelings.

the holistic labeling group. Apart from "joy/gratification," the variance was overall fairly high. If comparing the labels where at least one was rated "strong," the accuracy climbed to 64.9%. "Joy/gratification" and "pondering/reflecting" were in this case fairly reliable with regard to their hit results. The label "helplessness" showed the highest spreading, respectively the worst reliability.

Correct matches differed strongly with respect to label:

joy/gratification	82%
anger/irritation	59%
helplessness	40%
pondering/reflecting	59%
surprise	33%
unidentifiable episodes	55%

For detailed information about the inter-rater analysis please refer to Jacobi and Schiel (2003).

2.9 Problems and Suggestions for the Future

With regard to one of our goals — labeling training material for recognizers — the inter-rater results described above are disappointing. Only two of the categories were matched correctly relatively often, but even the category that was detected best ("joy/gratification") was labeled as something different more than half of the time by the other group. The missing end correction step certainly increased the number of errors, but it cannot explain the whole range of differing judgments. Apart from the missing end correction we assume, the following main reasons for the large difference between the two labeler groups: First, the emotions in the data mostly were subtle emotions. Obviously, it is very difficult to detect and judge such fine changes in facial expressions and voice. This assumption is strengthened by the finding that the emotions that were rated "strong" showed a higher inter-rater agreement.
Second, the labelers in the holistic group had to make two decisions:

1. Is there an emotional segment, respectively, where does an emotional segment start/end?
2. What is the content of the segment?

This doubly ambiguous situation could lead to more errors. In fact, the second group coded 50% more segments than the first group. The groups therefore differed not only in their judgments with respect to the content of a segment but also with respect to the question if there was an emotion to be labeled. Perhaps the second group used a high criterion (willingness to say "signal there" in an ambiguous situation, compare with signal detection theory (Egan, 1975)) and the first group used a low criterion, which means both labeler groups had a different response bias. Third, the categories with the best results ("joy/gratification" and "pondering/reflecting") were the ones with the highest homogeneity with respect to morphological features. Notably the lips curved upwards ("joy/gratification") and chewing on the lips ("pondering/reflecting") could be observed in most of the cases. The other emotions differed

a lot between the subjects (with the notable exception of surprise, which subjectively seemed relatively homogeneous). Probably only emotions that are shown in a relative similar way by different persons evoke a comparable categorization by different observers. Emotions with a wide range of possible realizations (displays), like polite anger, seem to be judged very differently by different observers.

The inter-rater results for the group without auditory information are very interesting if compared to those of the holistic group. Obviously, the task was much easier (assuming that the skill of the two labeler groups was roughly the same, which seems probable because they had equal training). Is the judgment of emotions perhaps easier without (perhaps confusing) context? This would contradict the subjective impression of most labelers that the task was more difficult than with context and also some studies about this subject (Wallbott, 1988; Munn, 1940; Sherman, 1927). We think that the main reason for the better reliability results was that the labelers had to make only one decision: the content decision (is there an emotion, and if yes, which one). The holistic group, as mentioned above, had to make two: one about the boundaries (beginning/end) and one about the content. Actually, many errors in the holistic group were caused by setting differing boundaries ("insertions" and "deletions," see above). In the holistic group 79% of all errors were substitutions/deletions (which actually included two mistakes: differing content and differing segment borders). Substitutions (21% of all errors) included only one mistake: the content. In the facial expression group 57% of all errors were substitutions, 43% were deletions/insertions (which could occur despite of largely predetermined segment boundaries, for example, if a segment was cut in two by the first group and left whole by the second group). Segmentation of a "user-state stream" obviously was as difficult (perhaps even more so) than assigning a label to a segment. Therefore, if a reliable way of segmenting a user-state stream can be found, perhaps even subtle emotions can be labeled relatively reliable.

With these considerations in mind, we propose the following for future attempts to label emotions:

- Try to make sure that all labelers use the same response bias. Since for the labeling of emotions no external validation is possible,[6] this can be achieved only in instructing the labelers to use one of the extremes: "Be certain to label all sequences where you perceive an emotional change," resp. "Only label the sequences where you are absolutely certain that an emotional change has occurred." The second approach would be the more practical, since labelers would ignore many weak emotional changes, which are hard to judge, as we have seen in our data. Of course, the response bias still would not be the same for all labelers, but it can be assumed that the biases would have a smaller variation.

[6] Asking the subjects what they "truly" felt would not help, because first, asking them afterwards results in judgments affected by memory and social response bias, and second, we do not want to label the emotions that the user feels, but the ones that are observed by a communication partner, which is something different.

- Alternatively to the instruction to only label sequences where the labeler is absolutely sure, one could introduce a rating for certainty. In this way data could be collected about the reliability, with respect to the certainty of the judgments.
- With respect to our finding that the labeling of emotions that were rated "strong" had a much higher reliability, one could argue to try to evoke stronger emotions in subjects and then label these. Although this is an interesting approach, it does not solve the general problem that everyday emotions are subtle. If we want to study them we have to accept that they subjectively are categorized very differently and have to look for the reason for this. Another point against evoking stronger emotion is that the validity possibly suffers. Provoked emotions are not the same as strong emotions that show up "by themselves."
- To increase the reliability of segmenting a continuous user-state stream, one could artificially cut the whole stream in small segments of the same length. The task of the labelers then would be to assign a label to each segment. The segments would have to be as small as the shortest possible user state that one wants to include (which can be very short: surprise in our case was the shortest and could comprise as few as five frames). The exact duration should probably be found out empirically.
- A more radical solution to the problem of reliable labels would be to take a step backwards and ask the question if it actually made sense to look for examples of the "big six"[7] like anger or surprise in a human–machine dialogue. The so-called "fundamental" emotions were formulated as categories with regard to evolutionary development (Plutchik, 1980). These emotions are also considered to be fundamental because all other emotions are thought to be somehow derived from them. So in effect they are the "parents" of a huge range of emotional states, which seldom are studied because these substates are even harder to define and to describe than the parents. So does it make sense to look for basic, superordinate concepts that mostly were studied in their "strong" form (that means as "full-blown" emotions) and with the goal to find the categories that exist below social conventions? Perhaps it would be better to look for not-so-well-defined, context-dependent states that serve a certain function within a dialogue. For a computer system it actually is not in the least important to detect if the user is angry, sad, happy or afraid. The computer system has to decide if the user is content (which is not the same as happy), not content (which is not the same as angry) or needs help (for example, other scenarios of course are conceivable). Not only is the state of the user is of interest but at the same time we are interested in the necessary reaction of the communication partner. We observed quite often that subjects were genuinely amused (without sarcasm) but at the same time were not content with the service of the system, so detection of "happiness" or "joy" would not have served our needs. We acknowledged the problem outlined

[7] In the literature on the study of emotions the "big six" are thought by some authors to be the fundamental emotions, probably existing in every culture: happiness, sadness, fear, disgust, anger and surprise (Cornelius, 2000). Authors differ in their opinion about the number of the "basic" emotions, which ranges from six to ten.

above at least a little when we decided not to include "emotion recognition" in SMARTKOM but instead use "user-state recognition." A user state was defined as an emotional or cognitive state, "that is of interest in the communication process." We perhaps did not take the second part of our definition not seriously enough and we concentrated mainly on "standard" emotions, not trying to find out if there were categories that were better suited for our special condition. Of course, it would be great if it was possible to construct universal emotion detectors. But emotional states strongly depend on the context — perhaps not with regard to the feeling (inside) but definitely with regard to the emotional display. And only the emotional display is available for detection. Luckily, emotional displays conform to cultural rules and serve a certain function in human–human interactions. These rules certainly have detectable features since they were developed to help regulate communication. To target these rules and functions for labeling and recognizing user states in human–computer interaction seems to be — after our experience with the SMARTKOM data — a more promising approach than trying to look for consistent features of "basic" emotions. We therefore propose to look for user states that are defined as "emotional or cognitive state of a user that calls for a certain action/reaction by the system and/or contains an important message for the system." One way to do this would be to start with no categories whatsoever and in a first step label only the "functional message" that an observer gets from a displayed user state. In a second step categories could be developed from the segmented material. Another option would be to define which states/messages are important for the system (or to which the system can react). Taking the example above, this could be "content," "not content, but not confused" and "confused/needs help." After that a labeling of the morphological changes in the face, voice quality, body movements, head movements and gestures could be carried out (perhaps realized as modifiers for the functional label). This approach would be similar to the one we chose for the gesture labeling: Do not use existing categories, because you do not know if they make sense in your case or if you lose important information because existing categories are too broad. Look for events with regard to your item of study (gestures, resp. user states) and categorize them according to their "function" in the dialogue. Then mark the observable, formal features of these events.

A lot more could be said with respect to the labeling of emotions/user states. The points above are the the most important considerations from our experience with labeling the data in SMARTKOM. We hope that they will prove themselves useful for future attempts to tackle the task of emotion/user-state labeling and recognition in humann–machine dialogues and everyday situations.

2.10 Prosodic Annotation of Audio with Formal Criteria

The goal of this labeling step was to capture the information that was contained in the voice: A labeler listened to the audio file and marked if any of the labels below occurred, together with their time of occurrence. A word could carry more than one

prosodic marker. The labels were given with respect to the "normal" speaking habits of a subject: If someone habitually articulated clearly or spoke emphatically only the very clearly articulated words or the words with strong emphasis were marked. The categories were:

- PAUSE_PHRASE: Irregular pause on a phrasal level/between units of meaning. Pauses between sentences or between main clause and subordinate clauses were not meant, except if the pause was very long.
- PAUSE_WORD: Irregular pauses between words.
- PAUSE_SYLL: Irregular pauses between syllables of a word.
- LENGTH_SYLL: Lengthening of a syllable; could occur at any syllable of a word.
- EMPHASIS: Emphatic accentuation/strong emphasis on a word or syllable.
- STRONG_EMPH: Very strong emphatic accentuation/very strong emphasis on a word or syllable.
- CLEAR_ART: Clearly articulated speech. Clear articulation could be seen as a weak version of hyperarticulated speech. The speaker used (tried to use) less colloquial speech and less dialect, and the speech emulated that of a news reader.
- HYPER_ART: Hyperarticulated speech; very strong increase of the clear articulation.
- LAUGHTER: Speech overlapped by laughter or sighing.

2.11 Inter-Rater Agreement Prosodic Annotation

The procedure for the analysis of the inter-rater agreement for the prosodic annotation was exactly the same as for the holistic labeling (see above). The agreement between the labelers in this group was very low, with 35% correct and an accuracy of −1.14%. The main reason for the negative accuracy was the huge overall amount of insertions/deletions. For detailed information about the inter-rater analysis please refer to Jacobi and Schiel (2003). Here again it seems to be a big problem for the labelers to decide if a label should be placed or not. This is backed up by the observation that the prosodic group was the one that had the most discussions about how to improve the consistency. Some labelers complained that the label instructions were too vague. They found it very hard to decide if a pause was long enough to be classified as an unusual long pause, and to distinguish emphasis and strong emphasis. Since no absolute measure for the length of pauses, the strength of emphasis, etc., exists, the labelers had to compare a possible instance of a label against the rest of the dialogue of the subject. This could be one reason why the judgments varied so widely. One could try to align the response bias of the labelers ("only mark a label if you are absolutely sure that it is there"), but this would reduce the labeled phenomena drastically. Taking everything together we think that the label categories simply were not very well suited for the material we had to label in SMARTKOM. The coding system was developed for emotional speech. The SMARTKOM corpus includes subtle everyday emotions but not much speech that actually could be called emotional. We suggest the approach outlined under the holistic section for a new attempt to capture significant aspects of subtle emotions in everyday situations.

3 Development of the Gesture Coding System

To define our categories, we started with a heuristic analysis of a number of recordings of human–machine dialogues. The data we used consisted of 70 sessions of about 4.5-min length each. The subjects had the task to "plan a trip to the cinema this evening." The scenario was Public (for more information on the WOZ recordings please refer to Schiel and Türk (2006)). The recorded sessions were analyzed by the following procedure: Two observers separately listed every episode that they could identify as a "functional unit." Then they tried to assign a label to the unit with respect to the observable meaning of the gesture ("meaning" with respect to the communication process). For obvious reasons, it was not attempted to ascertain the true intent with which the gesture was performed but just the intent that could be identified by an observer. Identifiable units with no observable meaning for the communication process were listed as well. Some questions for the list of units were: What is the best way to categorize the observed units? Is there a reference on the display? If so, which one? How can we define beginning and end of a given unit? The observers made their judgments independently. After completing the list, the identified gestures were discussed. Every label that could not be operationalized satisfactorily was removed. Similar labels were combined. Three broad categories emerged, each with several subcategories, called labels. After this, some additional labels were added that were not observed but were thought probable to appear or had to be added in order to complete the categories. For more information on the development of the gesture coding system please refer to Steininger et al. (2001a). A complete description of the labels can be found in Steininger et al. (2001b). In the next section the categories and labels are outlined.

3.1 Overview and Definitions of Gesture Labeling

The basic procedure for the gesture coding was as follows: All gestures were assigned to one of three broad categories according to the observable intention with which they were used: Is the gesture an interaction with the system (an "interactional gesture"), a preparation of an interaction (a "supporting gesture") or something else (a "residual gesture")? The only (interactional and supporting) gestures that were labeled were the ones that entered the so-called "cubus," the field of the display and the room above the display where the SiViT gesture recognizer recorded data (it corresponded roughly to the border of the display). Additionally, gestures that were interesting (but took place outside the "cubus") were coded in the category of residual gestures. Each label was complemented by several modifiers (see below).

3.2 Gesture Label Categories and Modifiers

Three broad categories were labeled:

1. interactional gesture (I-gesture)

2. supporting gesture (U-gesture)[8]
3. residual gesture (R-gesture)

Interactional Gesture

When a subject used a hand/arm movement to request something from the computer this gesture was called "interactional." A second type of interactional gesture was the confirmation of a question from the system. Only for interactional gestures was the stroke marked. We found that most gestures of this type had a specific acceleration/deceleration curve (whereas other gestures were more erratic with respect to velocity changes), often had an overall higher velocity and were mostly produced vertically to the body and the display (whereas other gestures were mostly produced parallel to the display/body). These last observations were not used as labeling criteria.

Supporting Gesture

The "supporting gesture" occurred in the phase when a request was prepared. It signified the gestural support of a "solo action" of the user (like reading or searching), which means an action that was a request or reaction to a request from the system. To distinguish it from an "interactive gesture," at least one of the following prerequisites had to be confirmed as true:

1. The gesture was accompanied by verbal comments from the user that made the preparational character of the underlying cognitive process obvious (for example, "hmm... where is... here?... well...").
2. The gesture was accompanied by a facial expression that made the preparational character of the underlying cognitive process obvious (like lip movements of silent reading, searching eye movements, frowning).

If a gesture that seemed to be an "interactional gesture" was interrupted (because of a change on the display or speech output from the system) it was marked as "supporting gesture," since it contained no request. In order to be able to differentiate a supporting gesture from a residual gesture at least one (or both) of the following two prerequisites had to be given:

1. A linkage between gesture and speech (both streams referred to the same topic).
2. An identifiable focus, i.e., gaze and gesture fell on the same spot.

Residual Gesture

A "residual gesture" did not prepare a request (at least not obviously) and was not a request or confirmation. A "residual gesture" was either an "emotional gesture" or an "unidentifiable gesture." When an unidentifiable gesture was marked, there existed:

1. no linkage between gesture and speech, and/or
2. no identifiable focus, which means gaze and gesture did not fall on the same spot

[8] "U" for the German "Unterstützende Geste".

Label

Below are listed all labels and modifiers.

I-labels:

- I-circle (+), I-circle (−)
- I-point (long +), I-point (long −)
- I-point (short +), I-point (short −)
- I-free (+), I-free (−)
- not recognizable[9]

U-labels:

- U-continual (read), U-punctual (read)
- U-continual (search)
- U-continual (ponder), U-punctual (ponder)
- U-continual (count)
- not recognizable

R-labels:

- R-emotional (+ cubus), R-emotional (− cubus)
- R-unidentifiable (+ cubus)
- not recognizable

Morphology

This modifier signified which hand and which finger was used to perform the gesture and if a pen was used (in some scenarios the subjects had a pen for the display, in some not. For more information on the scenarios please refer to Schiel and Türk (2006). Some peculiarities (like double pointing) were noted as comment (see below).

Reference Word

This modifier signified the word that was spoken in correspondence to an "interactional gesture." It is known that "hand gestures co-occur with their semantically parallel linguistic units" (McNeill, 1992). "Correspondence" therefore meant that the word occurred shortly (50 ms) before, simultaneously or shortly after the gesture and gesture, and word were linked with regard to the content.

Reference Zone

This modifier signified the region of the display to which an "interactional" or "supporting" gesture referred — the middle, one of the four corners, "whole display" or "not recognizable." "Whole display" was labeled if the hand/finger moved through at least two reference zones or in the case of "I-free" gestures, resp. if the gesture indicated the whole display.

[9] It actually could happen that the category could be decided upon but the exact label was not recognizable because a part of the gesture was occluded. However, this was a rare case.

Object

This modifier signified the "interactional gestures," if the targeted object was hit or missed (and if so, on which side it was missed). It could be "hit," "above," "below," "right," "left" or "not recognizable."

Stroke

This modifier signified the "culmination point" of the gesture, which means the "most energetic" or "most important" part of a gesture that is often aligned with the intonationally most prominent syllable of the accompanying speech segment (McNeill, 1992; Kendon, 1980). For pointing gestures it was the period the finger remained on the indicated object.

Beginning, End

The beginning of a gesture was determined with the help of the following criteria: (1) hand entered cubus, (2) end of previous gesture or (3) hand moved after pausing within the cubus for a certain time. The end of a gesture was determined with the help of similar criteria: (1) hand left cubus, (2) beginning of following gesture, (3) hand stopped moving within the cubus.

Comment

Comments about the stroke, beginning, end or peculiarities that could not be noted anywhere else. Comments referring to the whole file were marked in the header; comments that referred to a single label were marked as a modifier for the label.

3.3 The Gesture Labels

I-Label

- I-circle (+), I-circle (−): A continuous movement with one hand that specified one or more objects on the display. The display was touched (+)/not touched (−).
- I-point (long +), I-point (long −): A selective movement of one hand that specified an object on the display, with the display being touched (+)/not touched (−). The stroke of the movement (which means the pointing of the hand at the object) was of extended duration (20 frames or more).
- I-point (short +), I-point (short −): As I-point (long +/−), but the stroke of the movement (which means the period during which the hand was pointed at the object) was not extended/was very short (up to 19 frames).
- I-free (+), (−): A movement of the hand/the hands which took place above the display (+) or beside the display/between body an display (−) and signified a request of the user (for example, to go to another page). This gesture could vary considerably with respect to its morphology.

U-Label

- U-continual (read): A continuous movement of the hand above or on the display. At least during a part of the movement the subject obviously read, which means the text was read aloud or the lips moved.
- U-point (read): Like U-continual (read), but with a selective movement of the hand (not a continuous one).
- U-continual (count): A continuous movement of the hand above or on the display. At least during a part of the movement the subject obviously counted, which means numbers were uttered or formed with the lips.
- U-continual (search): A continuous movement of the hand above or on the display that moved over a large part of the display. The hand moved through at least two reference zones. From the context it was obvious that the subject was looking for something or preparing a request, that means through the spoken words and/or a following interactional gesture (selection).
- U-point (ponder): A selective movement of the hand to a spot on the display. The hand did not move.
- U-continual (ponder): As U-point (ponder) but with small movements of the hand.

R-Label

- R-emotional (+): A movement of one or both hands within the cubus with an obviously emotional or otherwise interesting content.
- R-emotional (-): Like R-emotional (+), but outside of the cubus.
- R-unidentifiable (+): A movement of one or both hands within the cubus that could not be assigned to any of the other labels, for example, the hand lying on the display.

3.4 Inter-Rater Agreement Gesture Labeling

The procedure for the analysis of the inter-rater agreement for the gesture labeling was exactly the same as for the holistic labeling (see above). Comparing the label categories (I-, U-, R-gestures) the inter-rater reliability was good: 85% correct matches were counted, and the overall accuracy was 52,3%. The most reliable label was "I-gesture." Comparing all labels, not only the categories, the reliability remained good. Here 90% of the matches were correct; however, the accuracy dropped to 45,1%. Since the most important distinction — the one between an interactional gesture (I-gesture) and a supporting gesture (U-gesture) — showed very few confusions, we were content with the results. Most confusions took place between U-gestures and R-gestures, which possibly results from the fact that gestures that help prepare a request or accompany a thought process can sometimes be short, casual and have no distinct acceleration/deceleration curve like the I-gestures. In this they resemble R-gestures that happen without a goal and are mostly subconscious (for example, scratching, impatient drumming). Another reason for the confusions could be that

Table 1. Inter-rater agreement for gesture labeling (%)

Label	I-gesture	U-gesture	R-gesture	None
I-gesture	76	3	3	18
U-gesture	1	54	0	45
R-gesture	9	34	44	13
None	38	38	25	0

an important criterion for distinguishing U- and R-gestures was the gaze: U-gestures were defined as being in the focus of attention, seen in the gaze that followed the gesture (for the most part) or a link between gesture and speech. The last could be asserted with relative ease, but sometimes it was difficult to judge where the g aze of a subject was directed to. Therefore, with more suitable camera angles the reliability of the coding system could perhaps be improved.

The insertions/deletions are unfortunate because the instructions for marking a label or not marking a label are very clear and easily observable (entering the display for the most part). That in spite of this relevant sequences are missed (or irrelevant coded) shows that even a well-defined label system suffers problems with regard to consistency. The insertion/deletion rate could probably reduced with more training and with a more thorough correction.

3.5 Considerations for Future Attempts of Labeling Gestures

The gesture coding system described above was especially designed for labeling gesture in a human–machine dialogue. It categorizes gestures according to their observable function for the user in the communication process with the machine. We think that such coding systems (functional and tailored to the context) are much needed in addition to general taxonomies (McNeill, 1992; Ekman, 1999; Friesen et al., 1979) and structural, formal or micro-coding systems (like "FACS" from Ekman and Friesen (1978)). Additionally, we assume the following two major points to be important for future attempts to label gestures and study gestural aspects of communication:

1. A very important aspect of gestures is their dynamic properties. Until now these properties have largely been ignored in taxonomies (with the exception of notation of dance (Laban, 1956)). This is a pity, since it is very likely that the dynamic properties carry important information about the meaning of a gesture. They likely are good candidates for automatic detection since dynamic properties often are subconscious and therefore are less influenced by social conventions and are less context-dependent. As described above, we found that intentional gestures ("interactional gestures") often seemed to have a distinct acceleration–deceleration curve, whereas supporting gestures showed relatively erratic velocity changes. To study such differences accurate measurements of the movements would be necessary — which is not too difficult a problem, since there exist many good techniques for measuring trajectories of gestures in space

(Mulder, 1994). However, additionally a wa y to categorize the dynamic properties will have to be found and this could be a tricky problem. Nonetheless, we strongly hope that research along this road will be forthcoming.

2. Taxonomies above the structural level are often very similar (that means they can be translated into each other with relative ease). There seems to exist an underlying assumption that one (functional) taxonomy is enough to describe all kinds of human gestures that show up during communication. We agree that it is desirable to have one "master taxonomy" that is broad enough to cover all relevant gestural phenomena to have a reference and to make studies comparable. However, we think that we additionally need taxonomies that are especially tailored to the different situations where gestures can show up. Not only does speech vary greatly from situation to situation, gestures will look very different during a talk, a phone call, an argument, during talking to a machine like in our data, or during describing an animated cartoon as in one very well known gesture corpus (McNeill, 1992). If one uses a taxonomy that was developed with data from a certain situation, one cannot expect to cover all gestural phenomena adequately. For example, if we had used the category "deictic gesture" (pointing movements) we would have missed the most important aspects of the gestures in our human–machine dialogues: A deictic gesture can have the meaning of a selection ("Ok, I want this"), but in some cases people point at something while they are still making up their minds if they really want the thing indicated ("Hmm, should I go to this cinema") or point to figure something out ("Uh — this restaurant is near my place — or isn't it ?"). These two kinds of deictic gestures not only have a different meaning, they are executed differently as well. If using a standard taxonomy we would have missed these differences. Another example for needing context-dependent studies of gestures are ideographs (sketch a direction of thought). Ideographs normally take place in front of the thorax. We assume that the "supporting gestures" we describe are such "thought-sketching" gestures. Since the subjects think about the things presented on the display, they move their hands in this area (and not in front of their body as in a human–human interaction). We suppose that a display encourages the subjects "to think with the hands" (that means trace a thought with their hands). This can be compared to the "thinking aloud" one sometimes uses to help make up one's mind. In a communication situation where the general physical conditions are different the gestures will change — this is obvious, but until now it has not initiated a gathering of data from such diverse situations.

If we truly want to find out what happens between two speakers on the nonverbal tier, we have to keep looking with a much broader scope than used until now.

4 Conclusion

The challenge we faced at the start of the project was to develop labeling systems for multimodal data that were tailored to the task, that were fast and that were suited for

the training of recognizers. We decided not to label on the morphological level but to define the labels with regard to the intent of the user, respectively his observable communication goal (in the case of user-state labeling and gesture labeling). Since we do not yet know which features of the gestures or the user states carry the vital information, we cannot resort to a low-level coding of morphological features, even if this would greatly improve the reliability of categories. For future endeavors of transliterations of spontaneous speech in human–computer interaction, we think that a very important point that has to be solved is the coding of so-called off-talk (Oppermann et al., 2001). For the development of machines that do not need a command word to switch on and off, the discrimination of commands to the computer and talk that is not aimed at the computer such a coding, is essential. Our system allows for such coding but it seems prudent to try to enhance the validity and reliability of off-talk labelings: subjective impression for these judgments plays a greater role than for most other categories of transliteration conventions for spontaneous speech. The system we described for gestural coding is reliable and able to pinpoint the important aspects of gesture events in a dialogue. It can easily be expanded for other contexts and requirements. However, we will need studies that target the dynamical properties of gestures if we want to come to a more complete understanding of the meaning and appearance of the gestural components of communication. In the case of the user-state labeling, the low inter-rater agreement of our nonformal coding system showed that the optimal way of labeling such cloudy categories as emotions or user states still has to be found. We made suggestions how to improve the reliability of a functional coding system without resorting to formal categories. Together with additional collections of emotional speech data — which in itself poses no trivial problem — they should lead to a better understanding of the connections between speech and emotion. Since even human observers have problems in deciding if an emotion is there and if it is, which one (in everyday situations where only subtle emotions are shown, not full-blown emotions like during an argument or in a TV show), feature detection alone will probably not suffice, but will have to be combined with context information and/or information from other recognizers and/or models of the user. If this will be proven as true, one last critical remark about the endeavor of teaching computer systems to detect emotions will have to be made: If it turns out that not only humans need context information to judge emotions but machines need it too, then perhaps they will only get really good at the job if we train them to expect certain emotions in certain situations for certain persons, like humans tend to do. However, we then may have the problem of a system that is projecting emotions onto a communication partner – a well-known phenomenon in human communication. With this option to look forward to, we strongly advise not to enhance the emotion detection capabilities of computers, but the ability to judge the requests from the user correctly (which is not the same), both the ones uttered verbally and the ones given by nonverbal means, be they gestural or through signs of disapproval, confusion or approval. That is to say, let us keep in mind what the computer system should do: fulfill a request. Computer systems that show polite considerateness to my — real or assumed — mood would be a marvel, but interacting with one could get as tedious as conversing with an overpolite human.

Acknowledgments

We give our thanks to the SMARTKOM group of the Institute of Phonetics in Munich that provided the Wizard-of-Oz data. Many thanks to B. Lindemann and M. Pätzold for their contribution to the development of the gesture coding system and to all labelers, who not only coded the data but made important contributions to the development of the label systems.

References

A. Batliner, K. Fischer, R. Huber, J. Spilker, and E. Nöth. Desperately Seeking Emotions Or: Actors, Wizards, and Human Beings. In: *Proc. ISCA Workshop on Speech and Emotion*, Belfast, Irland, 2000.

N. Beringer, S. Burger, and D. Oppermann. Lexikon der Transliterationen. Technical Document SmartKom 2, Bavarian Archive for Speech Signals (BAS), 2000.

R.R. Cornelius. Theoretical Approaches to Emotion. In: *Proc. ISCA Workshop on Speech And Emotion*, Belfast, Ireland, 2000.

J. Egan. *Signal Detection Theory and ROC Analysis*. Academic, New York, 1975.

P. Ekman. Emotional and Conversational Nonverbal Signals. In: L.S. Messing and R. Campbell (eds.), *Gesture, Speech and Sign*, pp. 45–55. Oxford University Press, 1999.

P. Ekman and W.V. Friesen. Facial Action Coding System (FACS). A Technique for the Measurement of Facial Action. *Consulting Psychologists Press*, 1978.

G. Faßnacht. *Systematische Verhaltensbeobachtung*. Reinhardt, Munich, Germany, 1979.

K. Fischer. Annotating Emotional Language Data. Verbmobil Report 236, DFKI, 1999.

W.V. Friesen, P. Ekman, and H.G. Wallbott. Measuring Hand Movements. *Journal of Nonverbal Behavior*, 1:97–112, 1979.

I. Jacobi and F. Schiel. Interlabeller Agreement in SmartKom Multi-Modal Annotations. Technical Document SmartKom 26, Bavarian Archive for Speech Signals (BAS), 2003.

A. Kendon. Gesticulation and Speech: Two Aspects of the Process. In: M.R. Key (ed.), *The Relation Between Verbal and Nonverbal Communication*, The Hague, The Netherlands, 1980. Mouton.

R. Laban. *Principles of Dance and Movement Notation*. Macdonald & Evans Ltd., London, UK, 1956.

D. McNeill. *Hand and Mind: What Gestures Reveal About Thought*. University of Chicago Press, Chicago, IL, 1992.

A. Mulder. Human Movement Tracking Technology. Hand Centered Studies of Human Movement Project 94-1, School of Kinesiology, Simon Fraser University, Burnaby, Canada, 1994.

N. Munn. The Effect of Knowledge of the Situation Upon Judgment of Emotion From Facial Expressions. *Journal of Abnormal and Social Psychology*, 35:324–338, 1940.

D. Oppermann, F. Schiel, S. Steininger, and N. Beringer. Off-Talk — A Problem for Human-Machine-Interaction. In: *Proc. EUROSPEECH-01*, Aalborg, Denmark, 2001.

R. Plutchik. *Emotion: A Psycho-Evolutionary Synthesis*. Harper and Row, New York, 1980.

C. Ratner. A Social Constructionist Critique of Naturalistic Theories of Emotion. *Journal of Mind and Behavior*, 10:211–230, 1989.

F. Schiel and U. Türk. Wizard-of-Oz Recordings, 2006. In this volume.

M. Sherman. The Differentiation of Emotional Responses in Infants. *Journal of Comparative Physiology*, pp. 265–284, 1927.

S. Steininger, O. Dioubina, R. Siepmann, C. Beiras-Cunqueiro, and A. Glesner. Labeling von User-State im Mensch-Maschine Dialog — User-State-Kodierkonventionen SmartKom. Technical Document SmartKom 17, Bavarian Archive for Speech Signals (BAS), 2000.

S. Steininger, B. Lindemann, and T. Paetzold. Labeling of Gestures in SmartKom - The Coding System. In: I. Wachsmuth and T. Sowa (eds.), *Gesture and Sign Languages in Human-Computer Interaction, Int. Gesture Workshop 2001 London*, pp. 215–227, Berlin Heidelberg New York, 2001a. Springer.

S. Steininger, B. Lindemann, and T. Paetzold. Labeling von Gesten im Mensch-Maschine Dialog — Gesten-Kodierkonventionen SmartKom. Technical Document SmartKom 14, Bavarian Archive for Speech Signals (BAS), 2001b.

S. Steininger, F. Schiel, O. Dioubina, and S. Rabold. Development of User-State Conventions for the Multimodal Corpus in SmartKom. In: *Proc. Workshop "Multimodal Resources and Multimodal Systems Evaluation"*, pp. 33–37, Las Palmas, Spain, 2002a.

S. Steininger, F. Schiel, and A. Glesner. Labeling Procedures for the Multi-Modal Data Collection of SmartKom. In: *Proc. 3rd Int. Conf. on Language Resources and Evaluation (LREC 2002)*, pp. 371–377, Las Palmas, Spain, 2002b.

H. Wallbott. Faces in Context: The Relative Importance of Facial Expression and Context Information in Determining Emotion Attributions. In: K. Scherer (ed.), *Facets of Emotion*, Mahwah, NJ, 1988. Lawrence Erlbaum.

Multimodal Emogram, Data Collection and Presentation

Johann Adelhardt, Carmen Frank, Elmar Nöth, Rui Ping Shi, Viktor Zeißler, Heinrich Niemann

Friedrich-Alexander Universität Erlangen-Nürnberg, Germany
{shi,adelhardt,batliner,frank,noeth,zeissler,
niemann}@informatik.uni-erlangen.de,

Summary. There are several characteristics not optimally suited for the user state classification with Wizard-of-Oz (WOZ) data like the nonuniform distribution of emotions in the utterances and the distribution of emotional utterances in speech, facial expression, and gesture. In particular, the fact that most of the data collected in the WOZ experiments are without any emotional expression gives rise to the problem of getting enough representative data for training the classifiers. Because of this problem we collected data in our own database. These data are also relevant for several demonstration sessions, where the functionality of the SMARTKOM system is shown in accordance with the defined use cases.

In the following we first describe the system environment for data collection and then the collected data. At the end we will discuss the tool to demonstrate user states detected in the different modalities.

1 Database with Acted User States

Because of the lack of training data we decided to build our own database and to collect uniformly distributed data containing emotional expression of user state in all three handled modalities — speech, gesture and facial expression (see Streit et al. (2006) and for an online demonstration refer to our website [1]). We collected data of instructed subjects, who should express four user states for recording. Because SMARTKOM is a demonstration system it is sufficient to use instructed data for the training database.

For our study we collected data from 63 naive subjects (41 male/22 female). They were instructed to act as if they had asked the SMARTKOM system for the TV program and felt content, unsatisfied, helpless or neutral with the system feedbacks. Different genres such as news, daily soap and science reports were projected onto the display for selection. The subjects were prompted with an utterance displayed on the screen and were then to indicate their internal state through voice and gesture, and at the same time, through different facial expressions.

[1] http://www5.informatik.uni-erlangen.de/SmartKom/SkBook.html

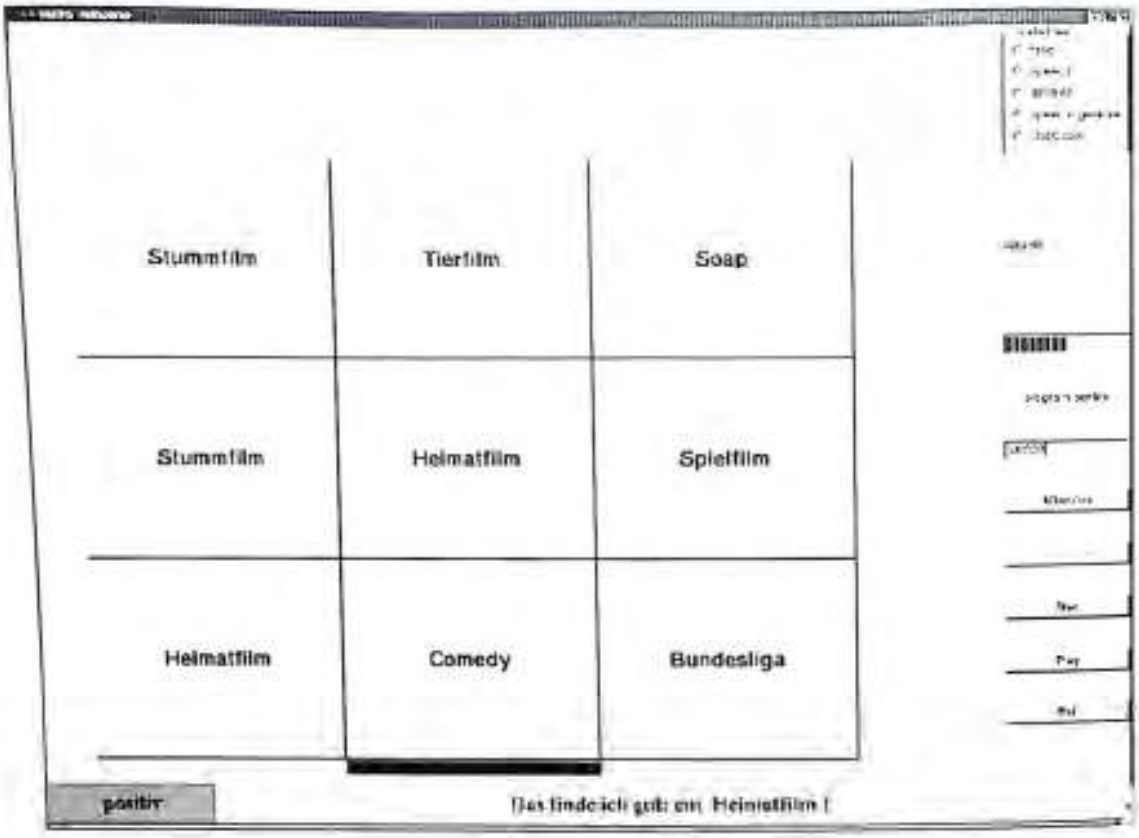


Fig. 1. Screen used for recording the user states in our local environment; several genres are shown together with a state that should be played. The *upper right corner* shows which modalities are currently recorded

1.1 Data Collection Environment

For the data collection we developed a special environment which we used directly with the local SMARTKOM demonstrator of the Institute of Pattern Recognition (LME). The screen of the graphical user interface of our experiment is shown in Fig. 1.

It shows in the center a 3×3 matrix of several genres, from which the subject had to choose one, together with a user state that should be expressed by the subject. This user state is shown in the lower left corner of the screen. At the middle bottom, the utterance, which the user had to speak at the current turn, is shown. Between the bottom of the 3×3 matrix of genres and the generated user utterance a colored bar is shown, which informs the user that the system is recording his speech.

On the right of the screen several features for handling the local data collection environment are shown. In the upper right corner five kinds of data collection can be chosen. Four of them refer directly to the modalities and to the combination of modalities used in SMARTKOM. The fifth one, UseCases, was used to collect data

Fig. 2. Sample data: from *left to right: hesitant, angry, joyful*, and *neutral*

related to so-called UseCases, which were used for the demonstration of the SMART-KOM system.

In the middle and the lower right part of the screen control buttons for camera and microphone and the field for the identification label of the experiment are shown. These were handled by the supervisor of the experiment with mouse and keyboard.

Data collection was done separately for all three modalities and for the combination of them. Because of the lack of annotated emotional data, the facial expression was recorded throughout, although the subjects were unaware of it. During the utterance the user had to select the genre by gesture in the centered choice matrix mentioned above.

Data collection was done in several sessions so the users could get familiar with the system and with the way to express their emotions naturally facing the camera and/or microphone in an incremental way. In the first session we recorded only the facial expression as the subject had to express the related emotion by speaking the utterance. In the second session we also recorded the speech of the subject together with facial expression. In the third session gesture and facial expression were recorded, while in the final session all three modalities were recorded as indicated with the tag *"speech_gesture"* (facial expression automatically included as mentioned above) in the upper right corner of the screen.

1.2 Collected Data

Facial expression, gesture and speech were recorded simultaneously in the experiment; this made it possible to combine all three input modalities afterwards. The user states were equally distributed. The test subjects read 20 sentences per user state. The utterances were taken in a random order from a large pool of utterances. About 40% of them were repetitions of TV genres or special expressions, which did not actual

depend on the given user state, like *"tolles Programm!"* (*"nice program!"*). In other words, we chose expressions one could produce in each of the given user states. (Note that a prima facie positive statement can be produced in a sarcastic mode and by that, turned into a negative statement.) All the other sentences were multiword expressions, where the user state could be guessed from the semantics of the sentence. The subjects should align to the given text, but minor variations were allowed.

From all collected data we chose 4848 sentences (3.6 hours of speech) with good signal quality and used them for further experiments. For the experiments with prosodic analysis, we randomly chose 4292 sentences for the training set and 556 for the test set.

For the facial analysis video, sequences of ten subjects were used. These subjects where selected because their mouth area was not covered by facial hair or the microphone. As training images, we used image sequences of these subjects without wearing the headset. In the images of the test sequences, there is a headset. Some of the training images can be seen in Fig. 2.

For gesture analysis there are, all in all, 5803 samples of all three user states (note that there are only three user states for gesture as mentioned in Shi et al. (2006)), and 2075 of them are accompanied by speech. As we are interested in the combination of all three modalities, we concentrate on this subset. Of this sample subset, 1891 are used for training and the other 184 are used for testing. Since the samples were recorded according to the user states categories in facial expression and speech, we merged the data of the corresponding user states *neutral* and *joyful* into a general user state category *determined* for gesture. The data we collected for the multimodal emogram (MMEG) are described in Zeißler et al. (2006), Frank et al. (2006) and Shi et al. (2006).

2 Multimodal Emogram (MMEG)

During the system development there always exists a problem if the input and output happen to take place simultanenously. It is also unwise to registrate any kind of output at the end of the processing queue without knowing the steps between input and output. In particular, if there are several steps for processing with statistical methods, as is the case in SMARTKOM with user states, it is absolutely necessary to know the intermediate results. The *multimodal emogram*[2] is a tool to show the results of user state processed in three modules: prosody, facial expression and gesture analysis.

The advantages of the evaluation tool are the following:

- quick presentation of recognizer results in the running environment
- systematic evaluation
- demonstration of functionality without the full system environment
- possibility to show the user the difficulty to act "as if" he were in the corresponding user state

[2] http://www5.informatik.uni-erlangen.de/SmartKom/SkBook.html

Fig. 3. Presentation of the results of the user state analysis in three modalities: speech, gesture and facial expression

For evaluating the system it is absolutely necessary to know the final result, but it is also important to know the results of the stages between input and output. In the case of error during the early development stage the developers are able to localize system components that produce errors.

In our case this is the presentation of the user states as recognized by the three modules. Their output is taken by the tool and is shown in a presentation where the user states are aligned parallel to the modalities. The presentation shows in this manner the detected user state of each module/modality at a certain time. So the recognition result of each single modality can be analyzed also for itself and at run-time.

The MMEG provides a compressed and transformed presentation of the user utterance in form of user states in quasi–real time. Just as the sonagram shows the spectral energy and the formants of the speech, the multimodal emogram shows user states as they occur in speech, facial expression and gesture. At the moment there are some problems in presentation concerning time alignment, synchronization and real-time behavior. But the idea of presentation generation in real time is realized, and it relies only on technical aspects such as the dependency on the word lattice or the power of the CPU. Another positive side effect of the evaluation tool is the possibility to show the functionality of the components processing the user state without running the full system.

For the robust classification of the user state it is necessary to analyze the input of three different modalities. Therefore, these single results have to be combined in a fusion component, which is performed by the *media fusion module* in the SMART-

KOM environment. By considering the result of the fusion, it is possible to know the contribution of each modality to the final recognized user state. By knowing the results of each system component, it is possible to give a clue to the current user state and thus to increase the system performance in several tuning steps.

Beyond evaluation the multimodal emogram delivers very valuable information of the analysis in the system. So it is possible to adapt it to several conditions:

- A user state is not recognized: the tool shows the results of the different modalities; with the help of these results it is possible to find out at run time why the user state is not recognized,
- Testing of new side conditions like new light conditions or new microphones.
- Training of system conformation behaviour for presentation tasks.

For the tuning task it is helpful to show the contribution and result of each modality. Like the sonagram shows the different frequencies in the speech signal, the multimodal emogram shows the results of user state expressed in speech, facial expression and gesture.

The user states are presented with the help of colored bars as shown in Fig. 3. For each modality there is one bar. We have four colors for the presentation:

- red: angry/anger
- green: joyful/joy
- yellow: hesitant/undecided
- blue: neutral

For the recognition of user states in gesture there is the restriction that only three user states are defined: angry, hesitant, and joyful together with neutral as the third class.

References

C. Frank, J. Adelhardt, A. Batliner, E. Nöth, R.P. Shi, V. Zeißler, and H. Niemann. The Facial Expression Module, 2006. In this volume.

R.P. Shi, J. Adelhardt, A. Batliner, C. Frank, E. Nöth, V. Zeißler, and H. Niemann. The Gesture Interpretation Module, 2006. In this volume.

M. Streit, A. Batliner, and T. Portele. Emotion Analysis and Emotion Handling Subdialogs, 2006. In this volume.

V. Zeißler, J. Adelhardt, A. Batliner, C. Frank, E. Nöth, R.P. Shi, and H. Niemann. The Prosody Module, 2006. In this volume.

Empirical Studies for Intuitive Interaction

Iryna Gurevych* and Robert Porzel

European Media Laboratory GmbH, Heidelberg, Germany
iryna.gurevych@eml-r.villa-bosch.de
robert.porzel@eml-d.villa-bosch.de
* Current affiliation: EML Research gGmbH

Summary. We present three types of data collections and their experimental paradigms. The resulting data were employed to conduct a number of annotation experiments, create evaluation *gold standards* and train statistical models. The data, experiments and their analyses highlight the importance of data-driven empirical laboratory and field work for research on intuitive multimodal human–computer interfaces.

1 Introduction

Research on dialogue systems in the past has focused on engineering the various processing stages involved in dialogical human–computer interaction (*HCI*), e.g., robust automatic speech recognition, intention recognition, natural language generation or speech synthesis (Allen et al., 1996; Bailly et al., 2003; Cox et al., 2000). Development of such dialogue technologies involves experimental work.

Many issues in the semantics and pragmatics of dialogue can be formulated as empirical topics. Therefore, the analysis of sufficient amounts of data is necessary to study a specific phenomenon or to train statistical models. Given a certain hypothesis, the researcher has to find out whether there is empirical evidence for the phenomenon in question, and how frequent and important it is. Also, rigorous large-scale evaluations of the algorithms are a driving force of the engineering progress. Such evaluations are only possible if sufficient amounts of annotated data are available. In this case, the output of the computer program (an *answer*) is compared to the gold standard (a *key*) defined by the annotated data (Romary et al., 2003).

Alongside these efforts, the characteristics of computer-directed language have also been examined as a general phenomenon (Darves and Oviatt, 2002; Wooffitt et al., 1997; Zoeppritz, 1985). In the following sections we describe new paradigms for collecting dialogue data. These data can be employed for a variety of necessary examinations that improve the performance and reliability of conversational dialogue systems.

Fig. 1. Operator in hidden operator tests

Fig. 2. Subject in hidden operator tests

2 Collection and Usage of Hidden Operator Test Data

The data collected in the first experiment was employed in research on empirical topics, such as semantic postprocessing of the speech recognizer's outputs and assigning domains to speech recognition hypotheses. We designed a set of annotation schemata and performed experiments with human subjects [1] resulting in a set of annotated corpora. The annotated data was, then, employed in order to:

- test whether the annotators are able to annotate the data reliably;
- produce a gold standard for automatic evaluations of the algorithms;
- create data models based on human annotations;
- produce training datasets for statistical classifiers.

2.1 Data Collection Setup

The data collection was conducted following an iterative data collection approach (Rapp and Strube, 2002). This approach has been developed within the EMBASSI research project[2] and employes the *hidden operator test* paradigm. In this experimental setup, the SMARTKOM system was simulated by an operator with the help

[1] Referred to as "subjects" or "users" in what follows.
[2] http://www.embassi.de

Table 1. Descriptive corpus statistics

	Dataset 1	Dataset 2
Number of dialogues	232	95
Number of utterances	1479	552
Number of SRHs	2.239	1.375
Number of coherent SRHs	1511	867
Number of incoherent SRHs	728	508

Table 2. Multiple domain assignments to a single SRH (in %)

	Dataset 1		Dataset 2	
Number of domains	Annotator 1	Annotator 2	Annotator 1	Annotator 2
1	90.06	87.11	90.77	88.7
2	6.94	11.27	9.11	11.19
3	3.01	1.28	0.16	0
4	0	0.35	0	0

of predefined dialogue scripts (Fig. 1). Twenty-nine subjects were prompted to say certain inputs in 8 dialogues, resulting in 1479 turns (Fig. 2).

Because of the controlled experimental setting and since prompts appeared on a per-turn basis, each user turn in the dialogue corresponded to a single intention, e.g., route request or sights information request. All user turns were recorded in separate audio files. The audio files obtained from the hidden operator tests were then sent to two differently configured versions of the SMARTKOM system. The data describing our corpora is given in Table 1.[3] The first and the second system's runs are referred to as *Dataset 1* and *Dataset 2*, respectively.

The corpora obtained from these experiments were further transformed into a set of annotation files, which can be read into a GUI-based annotation tool, MMAX (Müller and Strube, 2003a,b). This tool can be adopted for annotating different levels of information, e.g., semantic coherence and domains of utterances, the best speech recognition hypothesis in the N-best list, as well as domains of individual concepts. The two annotators were trained with the help of an annotation manual. A reconciled version of both annotations resulted in the gold standard. In the following, we describe a set of annotation experiments performed on the basis of the *Hidden Operator Test* data and present their results. Also, we sketch some applications of the annotated corpora for empirically grounded research.

2.2 Annotation Experiments

2.2.1 Semantic Coherence and Domains of SRHs

In this annotation experiment, the task of annotators was to classify the speech recognition hypotheses (SRHs) as either *coherent* or *incoherent*. We defined semantic coherence as well formedness of an SRH on an abstract semantic level (Gurevych et al., 2002). The SRHs from dataset 1 were randomly mixed in order to prevent them from being annotated in the discourse context. The resulting Kappa statistics (Carletta, 1996) over the annotated data yielded $\kappa = 0.7$. This indicates that human annotators can distinguish between coherent samples and incoherent ones.

In the second step, the coherent SRHs from both dataset 1 and dataset 2 were labeled with respect to their domains. We restricted the annotation to coherent SRHs

[3] We also include the distribution of coherent and incoherent speech recognition hypotheses, which was computed on the basis of annotation experiments described in Sect. 2.2.

Table 3. Class distribution for domain assignments (in %)

Domain	Dataset 1		Dataset 2	
	Annotator 1	Annotator 2	Annotator 1	Annotator 2
Electronic program guide	14.43	14.86	12.13	11.73
Interaction management	15.56	15.17	10.02	9.24
Cinema information	5.32	8.7	4.01	3.43
Personal assistance	0.31	0.3	0	0
Route planning	37.05	36	44.2	46.31
Sights	12.49	12.74	21.94	21.08
Home appliances control	14.12	11.22	7.59	8.1
Off-talk	0.72	1.01	0.11	0.1

only, as automatically assigning domains to SRHs should only be done if the utterance is a coherent one. The annotations resulted in two corpora of coherent speech recognition hypotheses labeled for at least one domain category. The percentage of SRHs with one or more domain attributions can be found in Table 2. It shows that the vast majority of hypotheses (ca. 90%) could be unambiguously classified in a single category. The class distribution is given in Table 3. The majority class is the *Route Planning* domain, while some of the categories, e.g., *Personal Assistance*, *Off Talk*, occur rarely in the data. Table 4 presents the kappa coefficient values computed for individual domain categories in dataset 1.[4] P(A) is the percentage of agreement between annotators. P(E) is the percentage that we would expect them to agree by chance. The annotations are generally considered to be reliable if $\kappa > 0.8$. This is true for all classes except those which are under-represented in this dataset (see Table 3).

2.2.2 Best SRH in the N-Best Lists

For this study a markable, i.e., an expression to be annotated, is a set of SRHs (N-best list) related to a single user's utterance. The number of markables in *Dataset 2* employed here corresponds to the number of utterances 552. The annotators saw the SRHs together with the transcribed user utterances. For each utterance a single SRH had to be labeled as the best one. The guidelines for selecting the best SRH were:

- How well the respective SRH captures the intention contained in the transcribed user's utterance.
- If several SRHs capture the intention equally well, the actual word error rate had to be considered.

The kappa coefficient — often applied to measure the degree of interannotator agreement — is not appropriate in this experiment, as its calculation is class-based. But the number of SRHs underlying the best SRH selection per utterance does vary.

[4] The kappa coefficient was not computed for dataset 2, but we expect the results to be very similar. For dataset 2 we employed a different agreement measure described in a separate experiment.

Table 4. Kappa coefficient for separate domains

	P(A)	P(E)	Kappa
Electronic program guide	0.9743	0.7246	0.9066
Interaction management	0.9836	0.7107	0.9434
Cinema information	0.9661	0.8506	0.7229
Personal assistance	0.9953	0.9930	0.3310
Route planning	0.9777	0.5119	0.9544
Sights	0.9731	0.7629	0.8865
Home appliances control	0.9626	0.7504	0.8501
Off-talk	0.9871	0.9780	0.4145

Therefore, we computed the percentage of utterances, where the annotators agreed on the correct solution resulting in 95.35% of inter-annotator agreement. This number suggests a rather high degree of reliability for identifying the best SRH by humans.

2.2.3 Best Conceptual Representation and Domains of SRHs

The best *conceptual representation* and the domains of coherent SRHs from dataset 2 were annotated. Due to lexical ambiguity of individual words, each SRH can be mapped to a set of possible interpretations called *conceptual representations* (CR). E.g., the German word *kommen* can be mapped either to the ontological concept `MotionDirectedProcess` or to the concept `WatchPerceptualProcess`. The algorithms operating on the ontology convert each utterance to a set of conceptual representations. As a consequence, they must be disambiguated automatically within the running system as well as manually for evaluation purposes.

Eight hundred sixty-seven SRHs used in this experiment are mapped to 2853 CRs, i.e., on average each SRH is mapped to 3.29 conceptual representations. The annotators' agreement on the task of determining the best CR resulted in ca. 88.93%. For the task of domain annotation, we computed two kinds of agreements. The first measure is the absolute agreement $Prec_{mark}$, when the annotators agreed on all domains for a given markable, i.e., an SRH. This resulted in ca. 92.5%. The second measure $Prec_{dom}$ denotes the agreement on individual domain decisions (6936 overall) and amounted to ca. 98.92%.

2.2.4 Domains of Ontological Concepts

In the last experiment, ontological concepts were annotated with zero or more domain categories.[5] The percentage of ambiguous domain attributions is given in Table 5. We extracted 231 concepts from the lexicon, which is a subset of ontological concepts occurring in our data. The annotators were given the textual descriptions of all concepts. These definitions are supplied with the ontology. Again, we computed

[5] Top-level concepts, e.g., Type, Event, are typically not domain-specific. Therefore they will not be assigned any domains.

Table 5. Domain assignments to concepts (in %)

Number of domains	Annotator 1	Annotator 2
0	4.76	16.45
1	60.17	44.16
2	22.51	18.18
3	5.63	10.82
4	1.73	7.79
5	2.16	1.3
6	0.43	0.43
7	0.87	0
8	1.73	0.87

two kinds of interannotator agreement. In the first case, we calculated the percentage of concepts for which the annotators agreed on all domain categories $Prec_{mark}$, resulting in ca. 47.62%. In the second case, the agreement on individual domain decisions $Prec_{dom}$ (1848 overall) was computed, ca. 86.85%.

A comparison between domain annotations of SRHs and ontological concepts (Table 6) suggests that annotating utterances with domains is an easier task for humans than annotating ontological concepts with the same information. A reason for this state of affairs might be that the high-level meaning of the utterance is defined through the context, whereas the meaning of an ontological concept is much less clear in the absence of context.

2.3 Usage of the Annotated Data

In Sect. 2 we discussed the reasons for creating large corpora with annotated data. Such data is necessary to perform empirical research in the automatic language processing. At first, a research hypothesis is formulated. Then, it is translated into an annotation scheme and validated empirically by measuring the reliability of human annotations based on the corresponding annotation scheme. Additionaly, human performance indicates the *upper boundary*, also called *ceiling*, in evaluating computer programs.

The data from hidden operator tests were used for the design and evaluation of some language processing algorithms. Following the common distinction in the evaluation practices, we differentiate between *intrinsic* and *extrinsic* evaluation. For example, the semantic coherence scoring algorithm was evaluated intrinsically for scoring sets of ontological concepts in terms of their semantic coherence (Gurevych et al., 2003). In this kind of evaluation, the output of the program is compared directly with the gold standard produced by the annotators. Extrinsic evaluation measures how a particular program contributes to the overall performance of the system. In our case, we employed the system in order to determine the best SRH in the output of the automatic speech recognizer (Gurevych and Porzel, 2003).

A further application of the annotated data was a set of experiments directed at automatically assigning the domains to speech recognition hypotheses (Rüggenmann

Table 6. Interannotator reliability for domain annotations (in %)

	$Prec_{mark}$		$Prec_{dom}$	
	Concepts	SRHs	Concepts	SRHs
Agreement	47.62	92.5	86.85	98.92

and Gurevych, 2004). The algorithm for determining the domains of conceptual representations and their underlying speech recognition hypotheses employs a knowledge source called the *domain model*. Two kinds of domain models were produced on the basis of the annotated data. The first domain model was derived by means of statistical analysis of speech recognition hypotheses in dataset 1 annotated for their domains. The second domain model was established through the direct annotation of concepts with respect to domains in the corresponding annotation experiments. Finally, the quality of both domain models and the domain recognition algorithm was evaluated intrinsically on the basis of annotated dataset 2.

3 Collection and Usage of the Wizard and Operator Test Data

The main goal of the second data collection and its experimental setup was to enable precise analyses of the differences in the communicative behaviors of the various interlocutors, i.e., human–human, human–computer and computer–human interaction. The setup of the experiment was, therefore, designed to enable the control of various factors. Most important factors were the technical performance (e.g., latency times), the pragmatic performance (e.g., understanding vs. nonunderstanding of the user utterances) and the communicative behavior of the simulated system. They were to be adjustable to resemble the state of the art dialogue systems, such as SMARTKOM. These factors can, of course, also be adjusted to simulate potential future capabilities of dialogue systems and test their effects.

3.1 Data Collection Setup

For conducting the experiments a new paradigm for collecting telephone-based dialogue data, called *Wizard and Operator Test* (WOT) — described by Porzel and Baudis (2004) — was employed. This procedure represents a simplification of classical end-to-end experiments, as it is — much like *Wizard-of-Oz* (WOZ) experiments — conductable without the technically very complex use of a real conversational system. As postexperimental interviews showed, this did not limit the feeling of *authenticity* regarding the simulated conversational system by the human subjects. The WOT setup consists of two major phases that begin after subjects have been given a set of tasks to be solved with the telephone-based dialogue system:

- In Phase 1 the human assistant is acting as a wizard who is simulating the dialogue system, much like in WOZ experiments, by operating a speech synthesis interface.

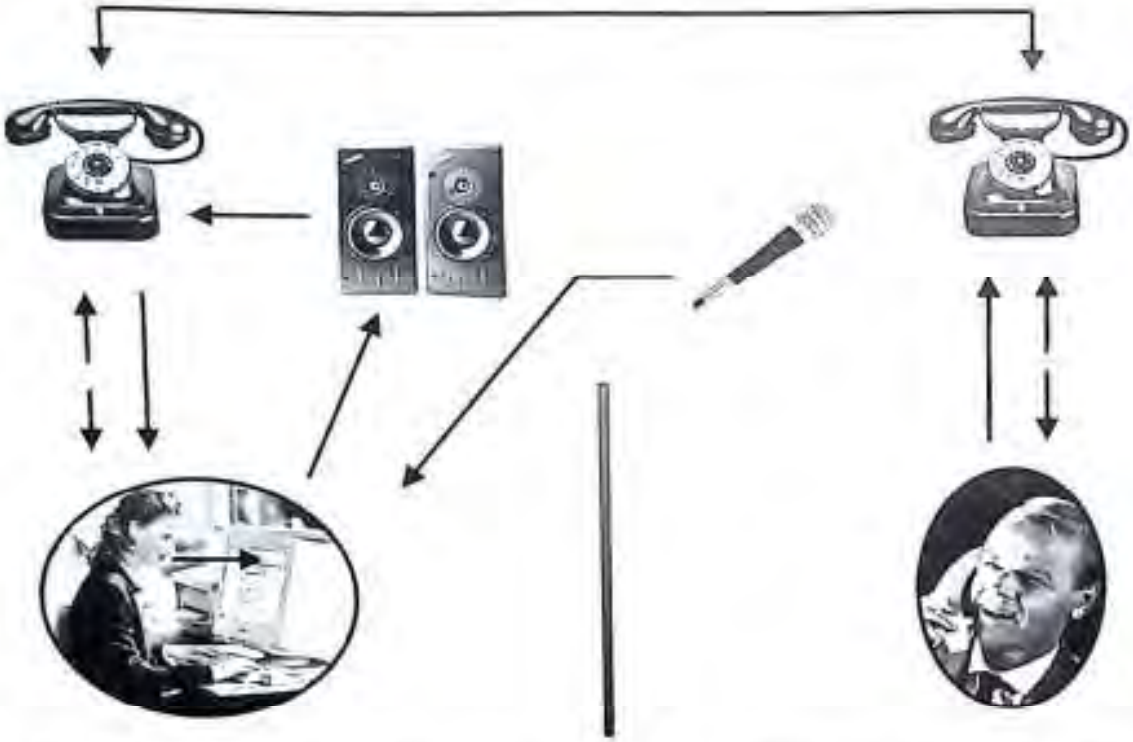


Fig. 3. Communication in phase 1 goes from synthesized speech out of the loudspeakers into the operator room (*left side*) telephone to the subject room (*right side*), and in phase 2 directly via the telephone between operator and subject

- In Phase 2, which starts immediately after a system breakdown has been simulated by means of beeping noises transmitted via the telephone, the human assistant is acting as a **human** operator asking the subject to continue with the tasks.

During the experiment the subject and the assistant were in separate rooms. Communication between both was conducted via telephone, i.e., for the user only a telephone was visible next to a radio microphone for the recording of the subject's linguistic expressions. As shown in Fig. 3, the assistant/operator room featured a telephone as well as two computers — one for the speech synthesis interface and one for collecting all audio streams. Loudspeakers were also present for feeding the speech synthesis output into the telephone and a microphone for the recording of the synthesis and operator output. With the help of an audio mixer all linguistic data were recorded time-synchronously and stored in one audio file. The assistant/operator acting as the computer system communicated by selecting fitting answers for the subject's request from a prefabricated list, which were returned via speech synthesis through the telephone. Beyond that, it was possible for the assistant/operator to communicate over telephone directly with the subjects when acting as the human operator.

The experiments were conducted with an English setup, with subjects and assistants at the International Computer Science Institute in Berkeley, USA, and with

Table 7. Descriptive corpus statistics of the English data (dataset E) and the German data (dataset G)

	Dataset E	Dataset G
Number of dialogues	22	22
Dialogue average length (min.)	5:53	4:57
Human–operator average length (min.)	2:30	1:52
Human–computer average length (min.)	3:23	2:59

a German setup, with subjects and assistants at the European Media Laboratory in Heidelberg, Germany. Both experiments were otherwise identical, with 22 sessions recorded in each. At the beginning of the WOT, the test manager told the subjects that they were testing a novel telephone-based dialogue system that supplies touristic information on the city of Heidelberg. In order to avoid the usual paraphrases of tasks worded too specifically, the manager gave the subjects an overall list of 20 very general touristic activities, such as *visit museum* or *Museum besuchen*, from which each subject had to pick six tasks that had to be solved in the experiment. The manager then removed the original list, dialed the system's number on the phone and exited from the room after handing over the telephone receiver. The subject was always greeted by the system's standard opening: *Welcome to the Heidelberger tourist information system. How can I help you?* After three tasks were finished — whether successful or not — the assistant simulated the system's breakdown and entered the line by saying *Excuse me, something seems to have happened with our system, may I assist you from here on*, and finishing the remaining three tasks with the subjects. Table 7 gives an overview of the data collected in the WOT experiments. Overall, the subjects featured approximately proportional mixtures of gender (25 male, 18 female), age (12< >71) and computer expertise. The average dialogue (German and English) consisted of 12.3 turns (composed of an average of 8.3 turns per dialogue in Phase 1 and 16.3 turns in Phase 2).

3.2 Measurements on the Data

These data enable analyses of the datasets that are language-specific as well as cross-linguistic analyses on the entire data, such as presented by Porzel and Baudis (2004). In their analyses pauses (i.e., silences over 1 second) not caused by system latency times, overlaps in speech, dialogue structuring signals (e.g., *well*, *yes* or *OK*) and feedback-channeling signals (Yngve, 1970) were in part manually and in part automaticall tagged and measured, yielding the results shown in Table 8.

As the primary effects of the human-directed language exhibited by today's conversational dialogue systems, the data collected in these experiments clearly show that:

- Dialogue efficiency decreases significantly even beyond the effects caused by latency times.

Table 8. Measurements performed on the English data (dataset E) and on the German data (dataset G)

	Dataset E	Dataset G
Dialogue average turns	14.33	10.28
Human–operator average turns	21.25	11.35
Human–computer average turns	7.4	9.2
Pauses	115	89
Pauses human–operator	21	10
Pauses human–computer	94	79
Overlaps	92	56
Overlaps human–operator	88	49
Overlaps human–computer	4	7
Dialogue-structuring signals	292	317
Signals human–operator	202	225
Signals human–computer	90	112
Feedback particles	43	153
Feedback human–operator	43	135
Feedback human–computer	0	18

- The human interlocutor ceases in the production of feedback signals, but still attempts to use his or her turn signals for marking turn boundaries, which, however, remain ignored by the system.
- The increases in the number of pauses is caused by waiting and uncertainty effects, which are also manifested by missing overlaps at turn boundaries.

4 Collection and Usage of the Field Operative Test Data

In an initial data collection we asked American native English speakers to imagine that they were tourists in Heidelberg, Germany, equipped with a small personal computer device that understands them and can answer their questions (Porzel and Gurevych, 2002). Among tasks from hotel and restaurant domains subjects also had to ask for directions to specific places. In the corpus we find 128 instances of instructional requests out of a total of roughly 500 requests from 49 subjects. The types and occurrences of these categories are in Table 9.

As can be seen by looking at category B, *Where interrogatives* are pragmatically ambiguous, i.e., they request either spatial localizations or spatial instructions. In order to perform an empirical study examining this phenomena under realistic conditions we performed the *field operative test* described below.

4.1 Data Collection Setup

Based on the results from these observations we conducted a *field operative test* (FOT) in which field operatives asked people on the street using interrogatives of

Table 9. Request types and occurrences

Type	Example	Occurrences
(A) How interrogatives	*How do I get to the Fischergasse*	38
(B) Where interrogatives	*Where is the Fischergasse*	37
(C) What/which interrogatives	*What is the best way to the castle*	18
(D) Imperatives	*Give me directions to the castle*	12
(E) Declaratives	*I want to go to the castle*	12
(F) Existential interrogatives	*Are there any toilets here*	8
(G) Others	*I do not see any bus stops*	3

Table 10. Request types and occurrences

Factors	Values
The goal object	The castle, city hall, school, discotheque, cinema, bank (ATM), clothing store
The time of day	Morning, afternoon, evening
The proximity to the goal object	Near , medium, far
The approximate age group	Young, middle, old
The gender of the subjects	Male, female
The weather conditions	Warm, cold, rainy, dry
The operative's means of transportation	On foot, bicycle
The baggage situation of the question	With, without luggage
The accessibility of the goal object	Open, closed

type A, B, C, D, E, and F (Table 9). They asked, e.g., `Where interrogatives`, such as *Excuse me, can you tell me where the cinema Europa is*, or `How interrogative`, such as *How do I get to the castle*, or `Existential Interrogatives`, such as *Is there an ATM here*. The passerby's responses were not recorded as that would have required their permission and thwarted a *natural* response. We logged and varied several factors shown in Table 10.

We asked 366 subjects, and their responses were then immediately categorized and logged according to their type, i.e., spatial descriptions, instructions, questions, etc., and their specific features, e.g., suggested means of transport (by bus, foot, etc).

4.2 Usage of the Field Data

This set of collected and categorized data was used to train classifiers and extract decision trees via standard machine learning algorithms. For example, as reported by Porzel and Gurevych (2003), the results of generating and rules applying a c4.5 learning algorithm (Winston, 1992) show that:

- If the object is currently closed, e.g., a discotheque or cinema in the morning, almost 90% of the *Where interrogatives* are answered by means of localizations, a few subjects asked whether we actually wanted to go there now, and one subject gave instructions.

- If the object is currently open, e.g., a store or ATM machine in the morning, people responded with instructions, unless — and this we did not expect — the goal object is near and can be localized by means of a reference object that is within line of sight.

5 Conclusion

We described three experimental paradigms for the collection of data employed in the dialogue system design. We showed that issues in dialogue processing can be formulated as empirical problems, and translated to annotation schemata and annotated corpora. The corpora can be employed as training data for statistical models and as a basis for symbolic models, such as domain models. A further advantage of our approach is a straightforward possibility for thorough evaluations of the developed algorithms. This is especially important for improving the quality and reliability of intuitive dialogue systems.

Acknowledgments

This work was partially funded by the German Federal Ministry of Education and Research (BMBF) as part of the SMARTKOM project under Grant 01 IL 905 C0 and by the Klaus Tschira Foundation. The authors would like to thank Dr. Michael Strube, Nicola Kaiser, Stefani Nellen, Florian Hillenkamp, Klaus Rüggenmann and Christof Müller for their help in designing and conducting the experiments.

References

J.F. Allen, B. Miller, E. Ringger, and T. Sikorski. A Robust System for Natural Spoken Dialogue. In: *Proc. 34th Annual Meeting of the Association for Computational Linguistics*, pp. 62–70, Santa Cruz, CA , June 1996.

G. Bailly, N. Campbell, and B. Mobius. ISCA Special Session: Hot Topics in Speech Synthesis. In: *Proc. EUROSPEECH-03*, pp. 37–40, Geneva, Switzerland, 2003.

J. Carletta. Assessing Agreement on Classification Tasks: The Kappa Statistic. *Computational Linguistics*, 22(2):249–254, 1996.

R.V. Cox, C.A. Kamm, L.R. Rabiner, J. Schroeter, and J.G. Wilpon. Speech and Language Processing for Next-Millenium Communications Services. *Proc. IEEE-2000*, 88(8):1314–1337, 2000.

C. Darves and S. Oviatt. Adaptation of Users' Spoken Dialogue Patterns in a Conversational Interface. In: *Proc. ICSLP-2002*, Denver, CO, 2002.

I. Gurevych, R. Malaka, R. Porzel, and H.P. Zorn. Semantic Coherence Scoring Using an Ontology. In: *Proc. Human Language Technology Conference / North American Chapter of the Association for Computational Linguistics Annual Meeting 2003*, pp. 88–95, Edmonton, Canada, 2003.

I. Gurevych and R. Porzel. Using Knowledge-Based Scores for Identifying Best Speech Recognition Hypotheses. In: *Proc. ISCA Tutorial and Research Workshop on Error Handling in Spoken Dialogue Systems*, pp. 77–81, Chateau-d'Oex, Switzerland, 28–31 August 2003.

I. Gurevych, R. Porzel, and M. Strube. Annotating the Semantic Consistency of Speech Recognition Hypotheses. In: *Proc. 3rd SIGdial Workshop on Discourse and Dialogue*, pp. 46–49, Philadelphia, PA, July 2002.

C. Müller and M. Strube. Multi-Level Annotation in MMAX. In: *Proc. 4th SIGdial Workshop on Discourse and Dialogue*, pp. 198–207, Sapporo, Japan, 4–5 July 2003a.

C. Müller and M. Strube. A Tool for Multi-Level Annotation of Language Data. In: *Proc. Diabruck: 7th Workshop on the Semantics and Pragmatics of Dialogue*, pp. 199–200, Wallerfangen, Germany, 4–6 September 2003b.

R. Porzel and M. Baudis. The Tao of CHI: Towards Felicitous Human-Computer Interaction. In: *Proc. Human Language Technology Conference / North American Chapter of the Association for Computational Linguistics Annual Meeting 2004*, Boston, MA, 2004.

R. Porzel and I. Gurevych. Towards Context-Adaptive Utterance Interpretation. In: *Proc. the 3rd SIGdial Workshop on Discourse and Dialogue*, pp. 90–95, Philadelphia, PA, 2002.

R. Porzel and I. Gurevych. Contextual Coherence in Natural Language Processing. In: P. Blackburn, C. Ghidini, R. Turner, and F. Giunchiglia (eds.), *Modeling and Using Context*, LNAI 2680, Berlin Heidelberg New York, 2003. Springer.

S. Rapp and M. Strube. An Iterative Data Collection Approach for Multimodal Dialogue Systems. In: *Proc. 3rd Int. Conf. on Language Resources and Evaluation (LREC 2002)*, pp. 661–665, Las Palmas, Spain, 2002.

L. Romary, M. Strube, and D. Traum. Best Practice in Empirically-Based Dialogue Research. In: *Proc. Diabruck: 7th Workshop on the Semantics and Pragmatics of Dialogue*, Wallerfangen, Germany, 4–6 September 2003.

K. Rüggenmann and I. Gurevych. Assigning Domains to Speech Recognition Hypotheses. In: S. Bangalore and H.K.J. Kuo (eds.), *Proc. HLT-NAACL 2004 Workshop: Spoken Language Understanding for Conversational Systems and Higher Level Linguistic Information for Speech Processing*, pp. 70–77, Boston, MA, 2–7 May 2004. Association for Computational Linguistics.

P.H. Winston. *Artificial Intelligence*. Addison-Wesley, Reading, MA, 1992.

R. Wooffitt, N. Gilbert, N. Fraser, and S. McGlashan. *Humans, Computers and Wizards: Conversation Analysis and Human (Simulated) Computer Interaction*. Brunner-Routledge, London, UK, 1997.

V. Yngve. On Getting a Word in Edgewise. In: *Papers From the 6th Regional Meeting of the Chicago Linguistic Society*, Chicago, IL, 1970.

M. Zoeppritz. Computer Talk? Technical Report 85.05, IBM Scientific Center Heidelberg, 1985.

Evaluation of Multimodal Dialogue Systems

Florian Schiel

Bavarian Archive for Speech Signals
c/o Institut für Phonetik und Sprachliche Kommunikation
Ludwig-Maximilians-Universität München, Germany
schiel@bas.uni-muenchen.de

Summary. In this chapter we will give a brief overview about what different methods of evaluation were applied to the SMARTKOM prototypes and which of them resulted in utilizable results and why others did not. Since there are no established benchmarking methods yet for multimodal HCI systems and very few debatable methods for monomodal dialogue systems our work within the SMARTKOM prototype evaluations was rather an exploration of new methods than the simple application of standard routines.

1 Introduction

The evaluation of a human-computer interface (HCI) is not a very well defined term. In general, we understand the term "evaluation" as a reproducible experiment or procedure involving human subjects[1] and the HCI which results in some kind of qualitative or quantitative statement about the quality of the HCI system or parts of it. Again, the term "quality" in this context is not very well defined either.

Therefore the "evaluation of a multimodal dialogue system" can have many different realizations with regard to

- which parts (modules, modalities) of the HCI are involved
- type and number of subjects
- which tasks the subjects have to solve with the HCI
- what data are measured during evaluation
- what kind of questionnaire is used for the subjects
- the analysis of the recorded data
- the combination of subjective and objective results

For SMARTKOM, being a multimodal system with three different technical scenarios[2] and many different task domains, the possible space of evaluations gets even larger.

[1] Referred to as "subjects" or "users" in what follows.
[2] Public, Mobile and Home.

The framework of this book does not allow us to list all the results en detail. The interested reader may refer to some of our publications (Beringer et al., 2002c,b) or contact the authors for the numerous project internal reports.[3]

In the following sections we first will explain the logistical and structural methods of the different SMARTKOM evaluations carried out and then give a short overview of each individual evaluation, and list and discuss the most prominent results or the possible reasons for failure.

2 Methods of Evaluation

The SMARTKOM evaluations can roughly be classified into the broader categories "subjective" and "objective" as well as into "accompanying" and "benchmark" evaluations.

2.1 Accompanying Versus Benchmark Evaluation

In the SMARTKOM project we applied two logistically different types of evaluation while applying the same evaluation techniques to both:

"Accompanying" evaluation denotes an ongoing evaluation during the design process of the prototypes. The motivation for this is quite clear: Flaws in the design should be detected as early as possible and removed. The ideal situation would be to have an ongoing evaluation for all three technical scenarios Public, Home and Mobile for the total period of development. Due to financial and time constraints this was not possible. Therefore only two accompanying evaluations were carried out during the SMARTKOM project.

The results of an accompanying evaluation may theoretically be used to quantify and document the progress of development, because the prototype may change (hopefully for the better) during the evaluation phase. Since we could carry out only two evaluations during development, the only progress documented here is that from development phase to the final prototype.

"Benchmark" evaluation is our term for an evaluation of a "frozen" prototype; a prototype is delivered to the evaluation team and may not be updated during the evaluation phase (which may last for more than a month). In SMARTKOM seven benchmark evaluations were carried out on the subfinal prototypes[4]; three subjective, three objective and one long-term subjective evaluation.

2.2 Subjective Versus Objective

All serious HCI evaluations involve experiments with human subjects as well as the measurement of system parameters. The former usually take place in form of

[3] In German only.

[4] After the delivery of the "final" prototype 4.2, another prototype version 5.1 was assembled. The evaluation of version 5.1 in an outdoor setting is described in Sect. 3.3.

questionnaires or interviews before/after the user has tested the system, while the latter are sampled from the system itself during the test runs with the user. The reason for this often-found combination is simply that we do not know which combination of measurable parameters of a HCI system provides an adequate performance score for the system. Or, with other words, the overall performance or quality of a complex HCI system cannot be measured like, for instance, the word accuracy of a speech recognition system.

For example, it is quite easy to measure the delay between the end of a user input and the beginning of the corresponding system output. However, it is not clear a priori that a shorter measured period improves the overall success of the dialogue between user and machine. And if it does, is it an important or less important parameter?

Some people argue that the total is the sum of all components, i.e., if we can objectively measure the performance of each module of a HCI system, then the overall performance should be the sum of all single performances. Since Verbmobil (Burger et al., 2000) we know that this is not even true for a system with 100% optimal harmonized interfaces between the different modules, because one single defective module can crush the entire processing chain of a temporal linear structured information processing system.

Therefore both, the subjective evaluations of subjects and the measurable parameters of a life session, are the basis for the advanced analysis of the quality of the HCI system. In the following we will discuss fundamental techniques for both methods.[5]

2.2.1 Subjective: Speaker Questionnaires/Interviews

Subjective evaluations about a HCI system may be obtained by a questionnaire or — better, but much more time-consuming — by an interview. For all subjective evaluations the rule applies: what you ask, is what you get. We therefore conducted a pretest (Kartal et al., 2002a) in the early phase of SMARTKOM to come up with a reasonable and usable catalogue of questions to the subjects.

In general, we distinguish the following types of questions:

- Questions with categorized answers (*codes*). This includes *multiple choice*, *ranking* (usually on a scale of 7) and answers that allow *defined quantities* only (e.g., the height in centimeters).
- *Cross questions* that are asked in the same way before and after the experiment to detect changes caused by the experience.
- Questions that may be answered in free texts (*free questions*).
- *Subject questions* that are not directly related to the experiment but characterize the background of the subject (e.g., about computer experience).

The subjective evaluation should provide answers to the following general questions (Beringer et al., 2002a):

- How do subjects evaluate the efficiency of SMARTKOM to solve practical tasks?

[5] Other terms often used for "subjective" and "objective" are "ergonomic" and "technical" or "modular."

- How do subjects evaluate input and output modalities and the dialogue?
- What kind of (subjective) problems occurred?
- How do users accept the system?
- How do subjects accept the biometric identification of SMARTKOM?
- Who do the subjects think might use such a system?

For the SMARTKOM evaluation this concept led to a first catalogue of 62 questions (Questionnaire 1 (Kartal et al., 2002a)) in the pretest that was conducted with participants of the Wizard-of-Oz recordings (Schiel and Türk, 2006): [6]

- 2 cross questions about how the user feels about emotion detection
- 7 subject questions (codes)
- 5 PARADISE questions (codes, see below)
- 8 general ergonomic questions (ranking)
- 10 ergonomic questions (free questions)
- 6 + 6 specific evaluations of SMARTKOM techniques (ranking)
- 11 questions about potential user groups (ranking)
- 7 questions about the usability of certain SMARTKOM domains (ranking)

Categorized answers can be easily summarized (mean and median). Free questions require a rather tedious manual processing, often combined with the finding of new categories. The interpretation of results is tricky, especially when the number of subjects is low.

As for the technique, we used HTML-based questionnaires that entered the answers directly into a simple database. We also tested face-to-face interview techniques but did not pursue these due to the costs.

After the analysis of the pretest an improved version was compiled as follows (Questionnaire 2, e.g., Kartal et al. (2002b)):

- 2 cross questions about how the user feels about emotion detection (ranking)
- 13 subject questions (codes)
- 11 general ergonomic questions (ranking)
- 8 evaluations of special SMARTKOM techniques (ranking)
- 11 questions about potential user groups (ranking)
- 5 questions about the usability of certain SMARTKOM domains
- 1 free question about problems and suggestions
- 1 free question about the communication in general
- 1 free question about additional input/output techniques
- 1 free question about the preferred identification technique
- 7 evaluations of the input/output modalities (ranking)
- 1 free question about what is most annoying when using SMARTKOM
- 1 question about how much the subject would pay for a system like that
- 1 free question about the most positive and most negative action of SMARTKOM
- 14 control questions with a different ranking system (ranking)

[6] Note that the participants in the WOZ recordings believed they were communicating with a real technical system.

This results in a catalogue of 78 questions total that was used in all accompanying SMARTKOM evaluations (Sect. 3.1).

For the final benchmark evaluations of all three prototypes (Sect. 3.2) this questionnaire was boiled down to ten free and 31 ranking questions (Questionnaire 3). All questions that were interpreted ambiguously by the majority of the subjects were removed; some additional questions were added. The ten free questions are as follows (abbreviated):

- Describe your expectations of the SMARTKOM system.
- Was the operating of SMARTKOM unambiguous?
- Did you use a functionality that the supervisor did not mention in her instruction?
- How did SMARTKOM help you to operate the system?
- Did you make mistakes? Which ones?
- How did SMARTKOM help you to avoid such mistakes?
- Did SMARTKOM come up with correct answers although you made a mistake?
- How did SMARTKOM fulfil your requests/wishes?
- Was the output (voice and display) always clearly understandable?
- Could you finish your task in an appropriate period of time?

Please refer to Beringer et al. (2003a) for a detailed listing.

2.2.2 Objective: Technical Evaluation

The technical or objective evaluation exploits two sources: the internal parameters of the HCI system itself, e.g., the recognized string of words produced by the speech recognition module, and labelings and segmentations of the recorded modalities during the test which must be produced by a labeler group after the experiment. The latter are needed to provide measurable parameters such as *word accuracy*, *gesture recognition* and *user state recognition*. This implies, of course, that for the evaluation it is not sufficient to run the prototype alone, because the prototype itself will not record, for instance, the video stream of the facial camera. In our setting we augmented the prototype system by the required recording equipment, such as recording of voice input and output, facial video, infrared video (SiViT stream) and system display output. Also, the labeling of the recorded data is a very time-consuming process, which explains that some results of the benchmark evaluations were published long after the termination of the project.

In total, 110 parameters were derived from the running system as well as from the labelings of the recorded data streams. Please refer to Beringer et al. (2002d) for a complete listing. As expected, the majority of the parameters turned out to contribute none or only very little to an overall performance score. Consequently, the number of analyzed parameters was reduced to 30 in the final benchmark evaluations to achieve more utilizable results.

2.2.3 Task Success: End-to-End Evaluation

The subjective and objective measures of an evaluation may be combined into a general score representing the overall performance. However, there are situations where

both measures are inadequate to judge the primary success of a human–machine dialogue, namely whether the given task has been solved efficiently or maybe not at all. For lack of a better term, we call this abstract concept "task success." It should be clear that the task success cannot be easily derived from the subjective or objective measures: a dialogue might be successful even if the majority of technical measures scored rather low or the user gave a bad evaluation because she had a different idea on how to solve the task, and vice versa. The task success is therefore judged by objective guidelines based on the recorded input and output information streams only. Because of this "black box" approach[7] we often call this type of evaluation "end-to-end."

In PARADISE (Walker et al., 1997), for instance, this third pillar is the normalized task success κ computed from the attribute value matrix (AVM). In PROMISE (see below) κ was not computable, because SMARTKOM tasks are much more fuzzy than the strictly aimed task of accessing a certain information from a database system.

2.3 PARADISE and PROMISE

As said before, the optimal evaluation strategy combines subjective and objective results as well as the task success of a series of experiments in a way that an overall quality score can be computed from the measured data, which is comparable across different domains and desirably even across different technical scenarios. In Walker et al. (1997), Walker and her colleagues proposed the PARADISE method to calculate comparable performance scores for voice dialogue agents. This method was applied successfully to several different monomodal dialogue systems (Walker et al., 1998). When trying to apply the PARADISE method to SMARTKOM we ran into several problems. The most prominent are:

- SMARTKOM uses multiple asynchronous modalities in combination, and therefore it is not always possible to assign labels of correctness to single recognition results.
- In typical SMARTKOM dialogues the tasks are much more fuzzy than in a simple database access task and also may be solved in very many different ways, simply because SMARTKOM has no strict dialogue control and allows the user to ask almost everything at all times.

In Beringer et al. (2002b) we proposed an adaptation of PARADISE called "Procedure for Multimodal Interactive System Evaluation" (PROMISE) that should work with a system with more freedom such as SMARTKOM. In a nutshell, PROMISE replaces the κ calculated from the AVM by a much broader "task success" score that may have only two possible values $+1$ and -1, while the multilinear regression analysis between subjective judgments and measured cost functions was replaced by

[7] What happens in the "black box" is not relevant, only the outcome of the dialogue.

a simple regression analysis (t-test).[8] By applying PROMISE we could — at least on a formal level — avoid the systematical problems of PARADISE when applied to SMARTKOM.

2.4 Task Success Versus Request Success

The task success of PROMISE is, of course, a rather ambivalent concept, because it is based on a whole recording session (which typically lasts 10 min). Therefore it was not surprising that the labeling in the end-to-end evaluation turned out to be a difficult task: in many cases it was almost impossible to assign the labels "success" (+1) or "failure" (−1) to a session. After the accompanying evaluations were completed we took another step and defined a more fine-grained value called "request success" that is assigned in the end-to-end evaluation to pairs or groups of turns between the subject and the system according to the following rules:

1. Each dialogue is automatically segmented into blocks.
   Each block starts with a regular user request. A regular user request must address the system in a clear and cooperative way. Irregular requests may include incomplete turns, hesitations, off-talk, ungrammatical sentences, inconsistent multimodal inputs, not understandable utterances; these are discarded from the analysis including the possible following system reply. Each block must end with a system reply.[9]
2. Each block is then evaluated with 1.0, 0.5 or 0.0 points.
   - 1.0 points : The user gets a complete and correct reply to her request. For example:
     `USR: ''Show me the TV program!''`
     `SMA: ''Here are the shows running at the moment.''`
     (displaying the correct program of all channels)
   - 0.5 points : The user gets a complete and correct reply to her request, but the system provides *additional and unnecessary* information so that the user has to filter out the requested items. For example:
     `USR: ''Which TV shows are on ARD tomorrow?''`
     `SMA: ''Here are the shows that will run tomorrow.''`
     (displaying the program of all channels)
   - 0.0 points : All other cases. For most SMARTKOM sessions this results in basically three possibilities:
     - no reply at all
     - a wrong or incomplete reply
     - an error message

[8] Of course, this implies that only judgments of the type *ranking* may be used in the PROMISE method, because a t-test correlation is only meaningful across a certain range of possible values.

[9] Please keep in mind that a user request may be just a gesture, while a system reply may also be just a change of display.

3. The total request success per session is calculated.
   Since there exist (rare) cases where in one block more than one user request and system answer take place,[10] there are two possible ways to average the individual judgments: simple average (sum all points and divide by number of requests) or blockwise average (average over requests within each block, then average over all block values). Although both methods show very similar results, it turned out that the blockwise average reached more significant correlations with technical measures. Therefore we decided to use the blockwise averaging.

This procedure results in an average request success value between 0.0 and 1.0 for each session; 0.0 denotes a complete failure, while 1.0 would — theoretically — suggest that all cooperative requests of the subject were satisfied perfectly.

3 SmartKom Evaluations

We are now ready to list the results of the individual SMARTKOM evaluations. In this contribution we will only present the most prominent results.[11]

3.1 Accompanying Evaluations — Home, Mobile

The two accompanying evaluations took place from 1 April 2002 to 31 June 2002 (scenario Home) and from 1 Aug to 31 Aug 2002 (scenario Mobile). Both a full ergonomic and a technical evaluation were carried out.

3.1.1 Technical Setup Home

For the Home scenario the studio was converted into a living room-like environment with sofa, low chair, TV and VCR. The graphical output was displayed on a portable Web pad Fujitsu Siemens, Stylistic 3500 X Pen Tablet (Windows NT, 256 MB) that the subject held on her knees. A pen was used for gesture input. The input microphone was a standard low-cost headset[12] connected to the Web pad. The display and audio modules were running on the Web pad; the Web pad was connected via standard cable network with the remaining prototype cluster (2 x Linux 900/1800 MHz, 1280 MB RAM total).

[10] This happens when the user does not wait for the answer to her first request but utters another request:
```
USR: ''What shows are on TV tonight?''
USR: ''Show me the program of today!''
SMA: ''Sorry, I did not understand you.''
SMA: ''Here are the shows running at the moment.''
```

[11] Please refer to Kartal et al. (2002b); Beringer et al. (2002d); Kartal et al. (2003); Beringer et al. (2003d,a,c,b); Jacobi et al. (2004); Schiel (2004) for a more detailed version of the evaluation results.

[12] Approved by system group.

3.1.2 Ergonomic Accompanying Evaluation Home

Prototype	Subjects	Male/Female	Naive/Experts	Quest.	Remarks
2.1	27	14m / 13f	11n / 10e	2	2 L2 speakers

General Results

The overall picture was rather negative. Subjects criticized long waiting periods between request and answer, nonexistent user guidance and that they felt left alone by the system. The topic of biometric identification seemed to be of high importance to almost all subjects, but with very mixed messages. Nearly all technical aspects of the SMARTKOM prototype have been evaluated negatively. The only exception was the "persona," although some users doubted that actions like blowing bubble gum or counting fingers were really necessary.

It turned out that naive users were not able to use the system on their own; even experts rarely completed the given task. Very often the long processing periods between request and answer led to delayed answers that were misinterpreted by the subjects.

A detailed discussion of all results can be found in Kartal et al. (2002b).

Most Prominent Problems (Number of Judgments in Brackets)

"SMARTKOM does not understand me" (13)
"System too slow" (12)
"Pointing gestures, help function, volume control do not work" (8)
"Bad/not working user guidance; actions of 'persona' are hard to understand" (12)
"Did not know what to say" (2)
"System crashed" (1)

Most Prominent Suggestions

"Improve speech recognition" (10)[13]
"Improve speed" (16)
"Improve pen input" (4)
"More functionalities" (2)
"More feedback about what the system is doing" (10)
"Better help" (2)

3.1.3 Technical Accompanying Evaluation Home

Prototype	Subjects	Parameters	Naive/Experts	Remarks
2.1	21	110	11n / 10e	–

[13] This probably did not refer exclusively to the speech recognition module alone, but to the complete "speech understanding."

General Results

The overall picture was rather negative, quite in correspondence to the ergonomic evaluation. Recognizers of all modalities scored rather low; there was not one correct treatment of off-talk and barge-in. The processing time between request and answer was extremely high; as a follow-up it almost uniformly happened that new requests were uttered before the answer to the first request was given. The correct answers to correct recognitions were very few. The dynamic help module was activated very often, but the impact to the dialogue was not effective. The only positive measurements were the output gesture of the persona (75% correct) and the syntax of the speech output (100%). For a detailed listing of all measurements please refer to Beringer et al. (2002d).

The PROMISE value could not be calculated reliably because the correlations between subjective judgments and measurements were in most cases not significant.

3.1.4 Technical Setup Mobile

The Mobile scenario was evaluated in the same studio. An evaluation in the field was not possible because of technical reasons. The subject was asked to stand or walk within a certain area where two video cameras could capture the user actions. The subject was holding the PDA (Compaq IPAQ, embedded Linux, graphical output displayed via VNC) in her left hand and the pen in her right. Input microphone was a headset Sennheiser Evolution Wireless E100 connected directly to the Windows 2000 machine of the prototype cluster (1 x Windows 2000 1800 MHz and 2 x Linux 900/1800 MHz, 1536-MB RAM total).

3.1.5 Ergonomic Accompanying Evaluation Mobile

Prototype	Subjects	Male/Female	Naive/Experts	Quest.	Remarks
3.1	16	7m / 9f	11n / 5e	2	2 L2 speakers

General Results

The overall picture was rather negative; in comparison to the previous evaluation in scenario Home it was even worse. Subjects criticized that in most cases a real interaction was not happening. Processing periods between request and answer were extremely high so that most subjects had doubts whether the system had already crashed. Seven subjects claimed that no other display output happened than the greeting of the persona.

It turned out that naive users were not able to use the system on their own; even experts rarely completed the given task. Very often the long processing periods between request and answer led to delayed answers that were misinterpreted by the subject. A detailed discussion of all results can be found in Kartal et al. (2002b).

Most Prominent Problems

"No communication at all" (6)
"System did not understand me" (6)
"System did not react to anything" (7)
"System too slow" (1)

Most Prominent Suggestions

"Better use single commands instead of sentences" (5)
"Better user guidance: signal when to speak, display should indicate what can be done next" (5)
"System should react faster so that you get a feeling of a real dialogue" (3)
"Input should be improved" (1)
"Better gesture input" (1)

3.1.6 Technical Accompanying Evaluation Mobile

Prototype	Subjects	Parameters	Naive/Experts	Remarks
3.1	18	110	12n / 6e	–

General Results

We found a low performance of all recognizers. Off-talk and barge-in were not handled correctly. Very long processing times occurred between request and answer; this caused double requests that were processed in parallel. Very few correct system replies were encountered; also very few correct input recognitions. For a detailed listing of all measurements please refer to Beringer et al. (2003d).

Unfortunately, in many sessions very few interactions between subject and system took place. Consequently, we had only very few usable measurements; in some cases the experiment was terminated prematurely because the system seemed to have crashed.

The PROMISE value could not be calculated reliably because the correlations between subjective judgments and measurements were in most cases not significant.

3.1.7 Summary Accompanying Evaluations

The results of both accompanying evaluations were fed back into the design process of SMARTKOM. Because of the negative experiences with naive users we decided to recruit only users that were familiar with SMARTKOM in the forthcoming benchmark evaluations. Also, the question arose whether a new and unusual interaction technique such as used in SMARTKOM should not be evaluated during a longer time span, because we learned from the first evaluations that most concepts and interaction patterns of SMARTKOM have to be learned or adapted to. Unfortunately a long-term evaluation clashes with the fundamental claim of every evaluation, namely that the

results should be statistically valid across different subjects and therefore needs at least 10–20 subjects per evaluation experiment. Our budget and time constraints did not allow us to blow up the three final benchmark evaluations so that every subject could test the system over a longer period. We therefore decided to stick with the plan and evaluate short sessions of a larger group of subjects while performing one or two long-term evaluations in parallel.

3.2 Benchmark Evaluations — Home, Public, Mobile

The three benchmark evaluations took place 10–20 March 2003 (Home), 11–13 June 2003 (Public) and 4–11 July 2003 (Mobile). A full ergonomic evaluation as described in Sect. 2.2.1 was carried out for each scenario. However, since we recruited only "experts" for the benchmark, evaluations, we changed the instruction phase to the following:

- The supervisor instructed the subject verbally, gave the task description and reminded the subject to wait for a system response before uttering the next request.
- Subjects were asked to start the session after a green light signal given from the operator and to conclude the session when the green light turned off.
- Subjects were shown a short video (5 min) demonstrating the proper usage of SMARTKOM (scenario Home).
- Subjects could perform a short training dialogue of 5 min.
- After the clarification of last questions the evaluation session of 10-min length was started.

During the session the supervisor did not interact with the subject.

After the negative experiences in the accompanying evaluations, the technical evaluation was reduced from 110 to a set of 30 reliable parameters (Jacobi et al., 2004). The main reason for this was the observation that many desirable parameters could not be extracted reliably from the system's log files and therefore did not show the desired correlations to the subjective judgments.

3.2.1 Technical Setup Home

The studio was equipped in a living room-like fashion with sofa, low chairs, TV and VCR. The subject sat in a low chair directly in front of a low table on which the display output of SMARTKOM was projected.[14] The facial video camera was active but the subject's face was not visible when leaning back ("lean-back mode," see task). Microphone input was via a wireless headset Sennheiser E100 connected to the Windows 2000 machine of the prototype cluster (1 x Windows 2000 1800 MHz and 2 x Linux 2800 MHz, 4608-MB RAM total). The subject was not supposed to stand up during the session. The task was: "Inform yourself about the TV program (tonight, tomorrow etc.), get details about certain shows, actors. Try to control your TV and VCR. Try deactivating and reactivating the display ("lean-back mode")."

[14] The Web pad was not used in the final benchmark evaluation because the hardware was not available. Therefore the subject could also use her finger to point.

3.2.2 Ergonomic Benchmark Evaluation Home

Prototype	Subjects	Male/Female	Naive/Experts	Quest.	Remarks
4.2	20	10m / 10f	0n / 20e	3	3 L2 speakers

General Results

The overall results were much more promising than in the previous accompanying evaluation. Improvements were judged mainly in speech understanding, gesture recognition, dialogue handling and shorter processing times. Still a problem seemed to be the missing 'user guidance' and some very long processing periods where the subjects were unsure about the state of the system.

Most Prominent Answers to Free Questions

Subjects criticized in unison that SMARTKOM did not help to learn the operation. About half of the subjects were satisfied with the system; others claimed that SMARTKOM did not understand (4), was too slow (2), did not recognize English movie titles (1), had no "back" button (2), had no emotion detection (1). Most of the subjects showed a high level of self-awareness with regard to their own mistakes. They listed 18 cases where they made a mistake. Unfortunately, SMARTKOM didn't seem to help in correcting those mistakes. The output was judged being more or less OK; the meaning of the "red lamp" was unclear to most of the users. Some subjects criticized the pronunciation of foreign words.

Most Prominent Rankings

The following aspects of SMARTKOM were ranked positively: clearly arranged display, speech understanding, speech input, SMARTKOM is easy to understand, task simple to solve. Negative rankings occurred in: speech output, emotion detection, good support, speed, fast reactions, help function, competence.

3.2.3 Technical Benchmark Evaluation Home

Prototype	Subjects	Parameters	Naive/Experts	Remarks
4.2	20	30	0n / 20e	-

As for the recognizer results, the speech recognition module scored much better than in previous prototype versions (47.1% accuracy), but not as good as in Public (57.8%) and Mobile (63.9%). User state recognition is still low (16.2%) but significantly better than in the Public scenario (8.4%). Gesture recognition reached 38.7% correctness. Crosstalk and off-talk was much lower than in previous recordings, probably due to the fact that the subjects were instructed more strictly. The average duration of a user speech input was significantly longer (2070 msec) than in the other two scenarios (approx. 1760 msec). It is unclear why, possibly because the domain TV allows more complex input sentences than other domains.

The request success (Sect. 2.4) scored an average of 0.48 per session, which was right on the total average (0.47). However, request success values varied from 0.8 to 0.27 per session, which shows that the successful usage of SMARTKOM very much depends on the user.

3.2.4 Technical Setup Public

The subject stood in front of the standard Public setup with the display projected on the graphic tableau, SiViT unit and facial camera. Microphone input was via a wireless headset Sennheiser E100 connected to the Windows 2000 machine of the prototype cluster (1 x Windows 2000 1800 MHz and 2 x Linux 2800 MHz, 4608-MB RAM total). The subject was not supposed to leave the range of the facial camera, but due to body and head movements the face of the subject was not always fully visible in the facial video stream. The task was: "Inform yourself about the cinema program in Heidelberg tonight, get details about certain movies, actors. Find the way to the selected cinema (maps). Try to make a reservation for tickets."

3.2.5 Ergonomic Benchmark Evaluation Public

Prototype	Subjects	Male/Female	Naive/Experts	Quest.	Remarks
4.2	11	6m / 5f	0n / 11e	3	2 L2 speakers

General Results

The overall results for the Public benchmark were very similar to the Home benchmark evaluation. Since there has not been an accompanying Public evaluation we cannot give any detailed improvements. Speech understanding, gesture recognition and dialogue handling seemed to work reasonably well. Two subjects claimed that SMARTKOM actually allowed them to solve the task as fast as with traditional techniques (phone, Internet).

Subjects criticized the missing "user guidance," some very long processing periods so that subjects were unsure about the state of the system and that the error message "I did not understand you!" was not very helpful because it did not relay any information about what was hard to understand.

Most prominent answers to free questions

Subject criticized in unison that SMARTKOM did not help to learn the operation. Five of the subjects were satisfied with the system; others claimed that SMARTKOM did not understand (4), was too slow (1), did not recognize gestures (1), had no "back" button (2) and that the reservation of movie tickets did not work at all (1). Most of the subjects showed a high level of self-awareness with regard to their own mistakes. They listed eight cases where they probably made a mistake. Unfortunately, SMARTKOM did not seem to help in correcting those mistakes. The output was judged as being more or less OK. Five subjects criticized the prosody of the speech synthesis; three subjects said that sometimes the meaning of the output was unclear; 1 subject claimed that speech and display were not in sync when the persona was speaking.

Most Prominent Rankings

The following aspects of SMARTKOM were ranked positively: display output, has a clearly arranged display, correct answers in most of the cases, speech input, SMARTKOM is easy to understand, task simple to solve, SMARTKOM is easy to operate. Negative rankings occurred in: emotion detection, gesture recognition, one input sufficient to reach goal, help function, fast reaction to gesture input, competence.

3.2.6 Technical Benchmark Evaluation Public

Prototype	Subjects	Parameters	Naive/Experts	Remarks
4.2	11	30	0n / 11e	–

The speech recognition module scored in the middle range (57.8% accuracy), better than in Home (47.1%), but not as good as in Mobile (63.9%). User state recognition is very low (8.4%) compared to Home (16.2%). This is surprising because one might think that the more rigid hardware setup of the Public scenario (fixed position of user, fixed background texture) would help the user state recognition. Gesture recognition scored 48.6%, which is significantly better than in Home. In four cases the user interrupted his own gesture or speech input caused by a system output (other scenarios: no occurrence). This might indicate a slightly worse dialogue handling in the Public domains. Crosstalk and off-talk was much lower than in previous evaluations, probably due to the fact that the subjects were instructed more strictly.

The request success (Sect. 2.4) scored an average of 0.45 per session which was slightly below the average (0.47). However, request success values varied from 0.62 to 0.23 per session, which shows that the successful usage of SMARTKOM very much depends on the user.

3.2.7 Technical Setup Mobile

For the Mobile scenario the studio was cleared out as much as possible to give subjects room to walk around. Two cameras were mounted to capture user actions. The subjects were asked to stand or walk within a marked area of the studio during the test. No background noise was played back. The subject was holding the PDA (Compaq IPAQ, embedded Linux, graphical output displayed via VNC) in her left hand and the pen in her right. Input microphone was a headset Sennheiser Evolution Wireless E100 connected directly to the Windows 2000 machine of the prototype cluster (1 x Windows 2000 1800 MHz and 2 x Linux 900/1800 MHz, 4608-MB RAM total). No facial video camera was used and therefore no user state recognition was available. The subject was not supposed to give the PDA out of her hands during the test. The task was: "Inform yourself about touristic sites in Heidelberg, get details about them and look at a map. Try to get directions for a walk (navigation). Also get information about parking sites."

3.2.8 Ergonomic Benchmark Evaluation Mobile

Prototype	Subjects	Male/Female	Naive/Experts	Quest.	Remarks
4.2	10	5m / 5f	0n / 10e	3	1 L2 speaker

General Results

The overall results were more promising than in the previous accompanying evaluation. Improvements were judged mainly in speech understanding and faster reaction to gesture input. On the other hand, the display output was judged much lower than in the other two scenarios. A problem still seemed to be the missing "user guidance" and some very long processing periods where the subjects were unsure about the state of the system. In a total comparison of all answers the scenario Mobile showed the lowest user satisfaction.

Most Prominent Answers to Free Questions

Eight subjects criticized that SMARTKOM did not help to learn the operation. About half of the subjects were satisfied with the system; others claimed that the display was confusing (1), SMARTKOM was too slow (2), did not recognize gestures (1), had an unsatisfactory feedback (2), had no emotion detection (1). Six of the subjects showed a high level of self-awareness with regard to their own mistakes. Five subjects claimed that they solved the task with the help of SMARTKOM in the same amount of time as with traditional techniques.

Most Prominent Rankings

The following aspects of SMARTKOM were ranked positively: pen input, speech understanding, speech input, speech output, gesture recognition (pen). Negative rankings occurred in: emotion detection, good support, speed, fast reactions, clear dialogue, help function and competence.

3.2.9 Technical Benchmark Evaluation Mobile

Prototype	Subjects	Parameters	Naive/Experts	Remarks
4.2	10	30	0n / 10e	-

As for the recognizer results, the speech recognition module scored the best result of all evaluated prototypes (63,9%). Crosstalk and off-talk was much lower than in previous recordings, probably because the subjects were instructed more strictly. The number of turns and the number of words[15] were significantly lower than in the other evaluations; it is not quite clear whether this is a good or bad tendency.

The request success (Sect. 2.4) scored an average of 0.49 per session, which was slightly above the total average (0.47). But again request success values varied from 0.70 to 0.25 per session, which shows that the successful usage of SMARTKOM very much depends on the user.

[15] Both user and system.

3.2.10 Pearson Correlation Subjective — Objective

To achieve significant correlations we ran the analysis across all three scenarios. As for the Pearson correlation between the subjective judgments (ranking only) and the measured parameters, we found only three value pairs with an error probability of less than 0.01. On this basis we did not deem the PROMISE approach applicable to this data.

A total of 17 value pairs were found with an error probability of less than 0.05. For a complete listing and discussion of these correlations please refer to Jacobi et al. (2004). Although some of these pairs (4) did not seem to make any sense, there are some interesting observations:

- A larger average processing time in the speech recognition module is not necessarily an indicator for a bad performance. Users who operate SMARTKOM successfully use larger and more complex requests than others. Consequently the speech recognition module needs more time to process them. This does not hurt the subjective positive judgment of the user.
- Some correlation pairs indicate that frequent interrupts of user actions by the system[16] lower overall performance.
- Unspecified error messages lower overall performance. Also, some correlations indicate that error messages should not be given as speech output alone.
- An unstructured display with too much information causes the users to produce off-talk, which in turn may lower the overall performance.
- Complex requests combined of speech and gestures are valued higher than simple requests, granted that they can be answered successfully.
- Frequent off-talk indicates problems in the interaction and dialogue handling.
- A funny acting or strangely designed persona triggers off-talk, which might lower overall performance.

3.2.11 Pearson Correlation Subjective — Request Success

As for the Pearson correlation between the subjective judgments (ranking only) and the request success per session, we found six value pairs with an error probability of less than 0.01. This suggests that the request success is a rather good overall performance indicator even for a multimodal HCI system.

Since the request success cannot determined automatically from the labeling and segmentation of the session data, it would be quite desirable to predict the average request success from the technical parameters. We therefore trained a simple linear model of technical parameters in the previously found correlation ranking, and tested the amount of explained variance (R^2) with increasing number of parameters (see Fig. 1). The measured parameters in the ranking of their descending correlation are as follows:

[16] For instance, when the user is about to point on the display and the display changes simultaneously.

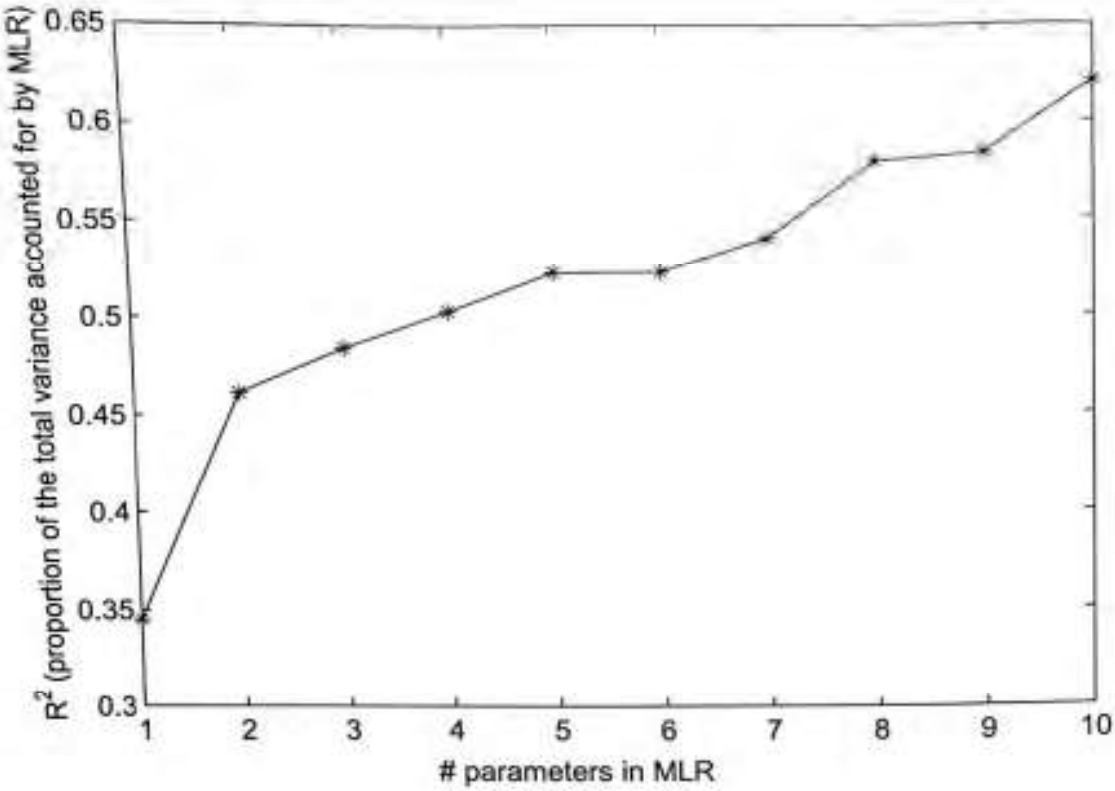


Fig. 1. R^2 explained variance in dependency of measured parameters that were fed into the linear predictor

1. Number of error messages	* −0.29
2. Average reaction time of SMARTKOM	* +0.38
3. Number of audio turns in dialogue	
4. Number of user turns in dialogue	
5. Speech recognition accuracy	* +0.23
6. Number of cross-talked user words	
7. Total number of words in dialogue	* +0.84
8. Total duration of dialogue	* −0.62
9. Total number of cross-talked words	
10. Number of SMARTKOM words in dialogue	* −0.90

Figure 1 shows that 60% of the variance of the request success may be explained with only six measured parameters (marked with "*" in the above listing together with their respective model weight). Although this is not perfect, it might be an alternative solution in cases where neither time nor money are available for the manual labeling and segmentation of request blocks.

3.3 Outdoor Evaluation—Mobile

3.3.1 Introduction

Between December 2003 and March 2004, SMARTKOM-Mobile (version 5.1) was tested in an outdoor setting in Heidelberg.[17] The purpose of this outdoor evaluation was generally similar to that of the benchmark evaluations described in Sect. 3.2, but the specific questions addressed were somewhat different:

1. Can people use SMARTKOM-Mobile effectively in an outdoor setting after a few minutes of instruction?
   Potential problems that could be caused by the outdoor setting include speech recognition problems due to ambient noise and distraction of the users by events and noise in the environment. In accordance with the results of the accompanying evaluations described in Sect. 3.1, it was assumed that previously inexperienced users would require a few minutes of instruction on the tasks that they were about to perform.
2. How do users rate SMARTKOM-Mobile overall, relative to other methods that they might use to accomplish the same tasks?
   Ultimately, users' acceptance of a system like SMARTKOM-Mobile will depend largely on whether they consider it more desirable than other available methods (e.g., conventional tourist guidebooks or more traditional computer-based systems).

Because of the differences in the environment in which the study was conducted and in the nature of the prototype used, the objective and subjective measures collected during this evaluation study were in part different from those of the studies reported in the previous section.

The study was prepared and conducted after the end of the main SMARTKOM project, so that the most advanced prototype from that project could be used as a starting point for the prototype used in the study. Consequently, the analysis of a sizable proportion of the data collected in this study had not been completed at the time of the writing of this book chapter; further results will be reported in later publications.

3.3.2 Method

After indoor pilot testing, the study was conducted in a pedestrian zone in the center of Heidelberg. The system was deployed on two laptops with 1800-GHz processors and with 1-GB RAM each, running Linux and Windows, respectively. A Compaq IPAQ 3630 served as the subjects' handheld device. It communicated with the laptops via a wireless LAN with 2-MBit bandwidth.

[17] The prototype was prepared and the tests were conducted by H. Aras, J. Häußler, M. Jöst and R. Malaka of EML, with consultation from A. Jameson of the Evaluation Center for Language Technology Systems of DFKI's project Collate.

The 8 subjects (6 female, 2 male) were recruited from groups that had no experience with SMARTKOM or other dialogue systems. Their average age was 27.1 years (standard deviation: 4.7 years). Two subjects were not native speakers of German. The subjects had virtually no experience with handheld computing devices (e.g., PDAs or organizers on cell phones) or with automobile navigation systems. Most of them occasionally used mapping systems on the Internet, and all claimed considerable general computer experience.

Subjects used SMARTKOM-Mobile to perform three tasks:

1. Controlling the display of a map: Having all objects of a certain type (e.g., cafés) displayed on the map; zooming in and out
2. Planning a route: Having the system compute and display a walking route between two locations in Heidelberg
3. Requesting information about sights: Having the system display information (a photo plus text) about one or more sights in Heidelberg

In all three tasks, commands to the system were given via speech; in the second and third tasks, pointing gestures could be used as well. The study began with a training phase in which the subject performed one instance of each of these three tasks. For each task, the user was shown a sheet of paper that indicated the actions that could be performed and some typical utterances (some of which were combined with gestures) that the system would understand. The user was asked to perform the actions indicated, using utterances like the ones listed on the sheet of paper.

During the test phase, for each task the subject was once again instructed to perform the actions associated with the task, but they had to recall the appropriate utterances from memory and/or construct similar utterances themselves.

3.3.3 Objective Results

The request success rate for the test phase was computed as described in Sect. 2.4. The overall request success rates for the three tasks were 0.59, 0.64 and 0.78, respectively, the overall rate being 0.65. In some cases, after an unsuccessful request, the subject repeated the request, often in a slightly different way, and ultimately achieved success; in other cases, success was never achieved. Because of the various differences between this study and the previous evaluations (aside from any relevant improvements in the prototype itself), it would not be meaningful to compare these success rates precisely with those found in previous evaluations. But in absolute terms, they do show that reasonable request success rates can be achieved in an outdoor setting with users whose experience with the system has been limited to a few minutes of instruction.

3.3.4 Subjective Ratings and Comments

After performing each task, the subject was asked to name his or her preferred alternative method for performing the same task. For example, for the task "Requesting information about sights," most subjects named the classical tourist guide book; for

the task "Controlling the display of a map" they tended to mention digital alternatives such as Web-based mapping systems. For each of five rating scales, the subject was then asked to compare the methods of SMARTKOM Mobile ("speech and pointing gestures") to the preferred alternative, on a 5-point scale ranging from −2 (e.g., "much less convenient") to +2 (e.g., "much more convenient"). The five scales referred to the criteria of convenience, speed,[18] intuitiveness , efficiency and overall preference. Averaging over all criteria and subjects, the average ratings for the three tasks were +0.5, +1.1 and +1.0 for the three tasks, respectively. That is, subjects showed a consistent preference for the methods offered by SMARTKOM-Mobile relative to the best alternative that they knew. This positive result does not imply that the SMARTKOM-Mobile prototype used for the study, with its long response times and far-from-perfect request success rates, would be preferred to the currently available methods; but it does show that the subjects were on the whole convinced of the suitability of SMARTKOM-Mobile's methods for tasks such as the ones that they performed.

After performing all three tasks, the subjects were asked about possible positive aspects of the character Smartakus. Of the eight subjects, four agreed with the statement that he was "entertaining" and four indicated that he enhanced the understanding of internal system states (such as being busy). But only two subjects found him "informative," and none agreed with the statement that he supported the progress of the dialogue. The freely reformulated comments likewise indicated that Smartakus was generally seen as a nice but—if necessary—dispensable extra element.[19]

The most important negative aspects of the system that the subjects mentioned in their freely formulated comments were the low processing speed and the limited accuracy of the speech understanding. In addition, when subjects were asked to specify situations in which they would choose to use a system like SMARTKOM-Mobile, two subjects consistently indicated an unwillingness to use it when other people were present, even though they were happy to use it when they were alone. Even if this attitude turns out to be limited to a small portion of the potential user group, it does point to a serious issue in connection with application domains such as tourism, in which users are often members of groups.

Concerns that could be addressed through relatively straightforward changes in the design and the hardware include: (a) the small size of the display, which was associated with limited legibility and amount of displayed information; (b) the relatively high speed with which textual information about individual sights was scrolled down the screen; (c) and the lack of an opportunity to obtain different information about sights than the information presented by default.

[18] In connection with this criterion, subjects were instructed to ignore the relatively long response times of the prototype used for the study, which ranged from several seconds for simple utterances to 20–30 seconds for complex requests, and to focus on the basic nature of the interaction.

[19] These ratings may have been affected by the fact that, in the prototype, the lip movements of the character were slow and not well synchronized with the speech output.

The positive aspects that were mentioned include: (a) the simplicity and intuitive usability of the interface—in particular, the ability to do without icons and menus; and (b) the speed and accuracy of the interaction via pointing gestures

3.4 Long-Term Evaluations–Public, Home

The motivation for a long-term evaluation was the trivial observation that the operating humans need to be trained to new technologies. This is true, for instance, for the basic desktop paradigm, where certain actions are always initiated with the left or right mouse button, but also true for simple interactions such as using a cellular phone. On the other hand, in the SMARTKOM evaluations we often found that subjects complained that the recording time for the session was limited to 10 min. They claimed that they were "... just getting the grip on it ..." etc. Therefore we wanted to see how judgments and dialogue efficiency changed for the better when the same user was able to use SMARTKOM more often and without any time limit.

The only long-term evaluation of the SMARTKOM prototype took place in Sept 2003 with only one person.[20] It comprised ten experiments with differing length, differing domains (tasks) but always the same technical setup Public.

Sex	Experience	Age	Smoker	Glasses	Beard	Weight	Height
male	high	40	no	no	no	75kg	186cm
Jewelry	Piercing	Handedness	Education	L1	L2	Bilingual	Culture
no	no	right	academic	German	English	no	German

3.4.1 Technical Setup Long-Term Evaluation

The technical setup of the prototype was identical to that of the Public benchmark evaluation (Sect. 3.2.4). Total recording time was 6.14 hours. Eight sessions were transcribed in a basic transliteration, which is a subset of the official SMARTKOM transliteration standard resulting in 810 turns. Gestures and user states were not labeled and segmented, because without the time limit of 10 min it was not possible to record the corresponding video signals. There was a minimum interval of 2 hours between each session. In most sessions no supervisor was present during the recording. No technical evaluation was carried out for the long-term evaluation.

3.4.2 Ergonomic Long Term Evaluation

The subject started the session and decided when to finish. Right after each session the user completed a questionnaire with seven free general questions, one coded

[20] Since this activity was not covered by the SMARTKOM budget we could not afford more than one study.

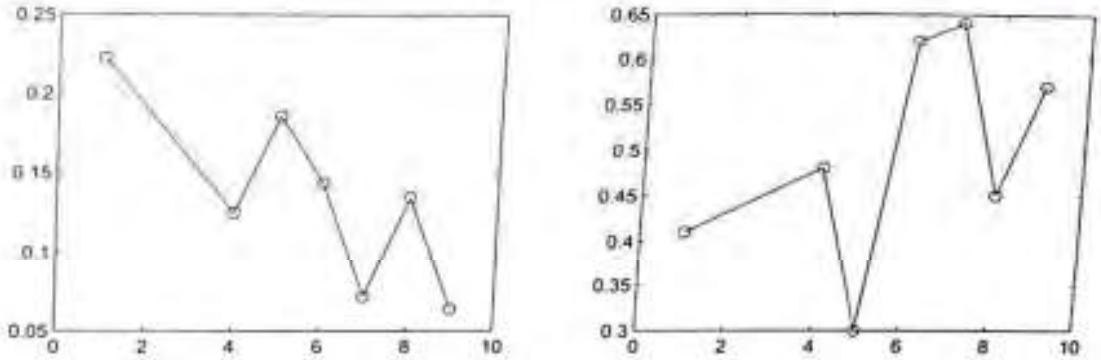


Fig. 2. Number of error messages per turn (*left*) and average request success (*right*) over long-term evaluation sessions.

question (domains) and five free questions per domain. For instance, in a session that comprised the topic TV+VCR, Cinema and Navigation this resulted in $7+3*15=52$ questions. A full listing of all answers as well as the transcripts can be found in Schiel (2004).

Adaptation

As expected, the user adapted his behavior quickly to the specific SMARTKOM dialogue technique. In Fig. 2 we show the average number of error messages of the system (left) and the average request success per session (right) against the total time the subject spent communicating with SMARTKOM. If we ignore the last session for the time being,[21] there is definitely a positive tendency over time, because the subject learned to operate SMARTKOM efficiently (error messages decrease, request success increases).

Most Prominent Positive Judgments

- Dialogue flow was structured if the user patiently waited for each answer.
- The subject was never bored.
- Reaction times were judged adequately.
- Speech understanding was very good if the headset level was adjusted carefully.
- Speech output was very good, but not with names.
- Locations on display and spoken input were associated correctly.
- Time concepts were recognized correctly (e.g., "right now," "tonight, "Friday night", etc.)
- Complex spoken inputs with three different types of information were recognized.

[21] In this last session the subject insistently tried to program the VCR although this functionality was not implemented in the testbed. Since it did not work, the subject tried about four different approaches to get the programming done. Naturally this trial-and-error technique resulted in a very high percentage of error messages.

- Switch on and off of VCR and TV worked.
- Change channel by pointing to a logo worked.
- Dialing the phone by using short names worked.
- Navigating on the maps worked by speech commands.

Most Prominent Negative Judgments

- Help function was not effective. It is not clear what the system calls "alternatives" and how the user should react to them.
- Anaphora were not resolved correctly ("Switch to that channel!")
- No explicit help function.
- Some system outputs came in two separated blocks with a silence interval in between; this can lead to cross-talk with the next user request.
- SMARTKOM seems to have a very limited dialogue memory. After asking for the TV program of Friday, the request: "What about the evening?" was answered by showing the program of tonight.
- Level of microphone was critical; only the slightest change in position may cause the speech recognition to break down.
- Reservation for cinema tickets did not work because the pointing finger always triggered more than six seats on the display.
- Sometimes SMARTKOM misunderstood an input ("ProSieben") even if the subject pointed to a logo of ProSieben at the same time.
- Using the phone was tricky because
  - the input of numbers by pointing was too slow
  - input of a number by voice did not work
  - after scanning a fax the user couldn't input the addressee
- Navigating in Heidelberg worked only if both the starting point and the end point of the desired route were entered in one sentence.

Most Prominent Suggestions for Improvement

- Context-dependent help function that can be called any time. After the help the dialogue should return to the point where help was called.
- Level dynamics of the microphone input more robust; it would be better to use speech input with directed microphone instead of headset.
- Output of the system should never come in more than one block with a silence interval in between.
- User should be able to interrupt system output any time.
- Dialogue handling: If the system gives a wrong answer, the user should be able to repeat the input. Currently this is only possible when the system issues an error message.
- Error messages should contain a diagnosis about the nature of the error (especially if the error is caused by an external device).

4 Conclusion

During the late design phase of SMARTKOM two accompanying ergonomic and technical evaluations were carried out, partly with naive subjects. Although the results provided valuable input to improve the prototypes, the results were statistically invalid and the PROMISE approach could not be applied.

Then the prefinal prototype 4.2 was evaluated in all three technical scenarios using an improved subjective and objective evaluation scheme. These benchmark evaluations showed a progress in almost every aspect of the prototype functionalities. The changes in the ergonomic and technical evaluation techniques led to more substantial results, which were easier to interpret than in the previous accompanying evaluations. In most sessions real dialogues took place between the user and the system, which allowed us to make constructive suggestions for further improvement.

The subjective judgments showed that the basic combined multimodal interaction between human and machine was working successfully. However, it is still a long way to an absolutely smoothly running interaction. The users made many recurring suggestions about how to better the dialogue flow. The overall impression was that the technical quality of the single modules for multimodal input and output was sufficient; the problem now lies more in the dialogue handling, error treatment and the ergonomic user guidance than in the recognizer techniques.

The PROMISE approach was not applied to the data, because the number of significant correlations was too small. On the other hand, the proposed request success value seemed to be in good correlation with the summarized subjective judgments and had more significant correlations with technical parameters. The request success was in the same range for all three scenarios (0.47 on a scale from 0 to 1).

The measured technical parameters of the system show an ambivalent picture. Although some values (e.g., speech recognition accuracy) seem to be low at first glance — especially when compared to accuracies from speech recognition tasks which range between 70–80% for a comparable task (Burger et al., 2000) — this is obviously not a severe problem when the module runs in a well-defined framework together with other recognizers (as shown in the correlation values between technical parameters and user judgments). Insofar one of the claims of SMARTKOM namely that multimodal interaction will be more robust than monomodal seems to be true.

Finally, a long-term evaluation was carried out with a single user. The results confirmed our hypothesis that a new technology like SMARTKOM needs to be adapted to. The user quickly learned how to operate SMARTKOM successfully and managed to use all capabilities of the system while decreasing the error rate over time. We therefore strongly recommend to use long-term evaluations for complex HCI systems in the future.

Acknowledgments

This contribution would not have been possible without the work of the following members of the SMARTKOM team (in alphabetical order): H. Aras, C. Beiras-Cunqueiro, N. Beringer, S. Hans, J. Häußler, I. Jacobi, A. Jameson, M. Jöst, U.

Kartal, G. Kouam-Wotchung, M. Libossek, R. Malaka, H. Mögele, M. Mozul, D. Sonntag, S. Steininger, J. Tang, S.V. Tiedemann, U. Türk and L. Wang.

References

N. Beringer, I. Jacobi, and S. von Tiedemann. Auswertung der ergonomischen Evaluation des SmartKom Prototypen -Szenario Home- Befragungszeitraum 10. Mrz 03 bis 20. Mrz 03. Memo SmartKom 16, Bavarian Archive for Speech Signals (BAS), 2003a.

N. Beringer, I. Jacobi, and S. von Tiedemann. Auswertung der ergonomischen Evaluation des SmartKom Prototypen -Szenario Mobil- Befragungszeitraum 04. Juli 03 bis 11. Juli 03. Memo SmartKom 18, Bavarian Archive for Speech Signals (BAS), 2003b.

N. Beringer, I. Jacobi, and S. von Tiedemann. Auswertung der ergonomischen Evaluation des SmartKom Prototypen -Szenario Public- Befragungszeitraum 11. Juni 03 bis 13. Juni 03. Memo SmartKom 17, Bavarian Archive for Speech Signals (BAS), 2003c.

N. Beringer, U. Kartal, M. Libossek, and S. Steininger. Gestaltung der End-to-End Evaluation in SmartKom 2.0. Technical Document SmartKom 19, Bavarian Archive for Speech Signals (BAS), 2002a.

N. Beringer, U. Kartal, K. Louka, F. Schiel, and U. Türk. PROMISE: A Procedure for Multimodal Interactive System Evaluation. In: *Proc. Workshop "Multimodal Resources and Multimodal Systems Evaluation"*, pp. 77–80, Las Palmas, Spain, 2002b.

N. Beringer, K. Louka, V. Penide-Lopez, and U. Türk. End-To-End Evaluation of Multimodal Dialogue Systems — Can We Transfer Established Methods? In: *Proc. 3rd Int. Conf. on Language Resources and Evaluation (LREC 2002)*, pp. 558–563, Las Palmas, Spain, 2002c.

N. Beringer, H. Mögele, I. Jacobi, C. Beiras-Cunqueiro, L. Wang, U. Türk, G. Kouam-Wotchung, M. Mozul, and U. Kartal. Auswertung der End-to-End Evaluation des SmartKom Prototypen (technische Evaluation, Evaluationstool, PROMISE) -Szenario Home-, Aufnahmezeitraum 01. April 02 bis 31. Juni 02. Memo SmartKom 13, Bavarian Archive for Speech Signals (BAS), 2002d.

N. Beringer, H. Mögele, I. Jacobi, C. Beiras-Cunqueiro, L. Wang, U. Türk, G. Kouam-Wotchung, M. Mozul, and U. Kartal. Auswertung der End-to-End Evaluation des SmartKom Prototypen (technische Evaluation, Evaluationstool, PROMISE) -Szenario Mobil-, Aufnahmezeitraum 01.August 02 bis 31. August 02. Memo SmartKom 15, Bavarian Archive for Speech Signals (BAS), 2003d.

S. Burger, K. Weilhammer, F. Schiel, and H.G. Tillmann. Verbmobil Data Collection and Annotation. In: W. Wahlster (ed.), *Verbmobil: Foundations of Speech-to-Speech Translation*, pp. 537–549, Berlin Heidelberg Germany, 2000. Springer.

I. Jacobi, H. Mögele, S. von Tiedemann, and U. Türk. Auswertung der End-to-end Evaluation des SmartKom Prototypen (technische Evaluation, PROMISE)

- Szenario Home, Public, Mobil. Memo SmartKom 19, Bavarian Archive for Speech Signals (BAS), 2004.

U. Kartal, N. Beringer, and S. Steininger. Auswertung der ergonomischen Evaluation der WOZ-Aufnahmen im Projekt SmartKom. Memo SmartKom 11, Bavarian Archive for Speech Signals (BAS), 2002a.

U. Kartal, N. Beringer, S. Steininger, and D. Sonntag. Auswertung der ergonomischen Evaluation des SmartKom Prototypen -Szenario Home-, Befragungszeitraum 01. April 02 bis 31. Juni 02. Memo SmartKom 12, Bavarian Archive for Speech Signals (BAS), 2002b.

U. Kartal, N. Beringer, S. Steininger, and D. Sonntag. Auswertung der ergonomischen Evaluation des SmartKom Prototypen -Szenario Mobil-, Befragungszeitraum 01.August 02 bis 31. August 02. Memo SmartKom 14, Bavarian Archive for Speech Signals (BAS), 2003.

F. Schiel. SmartKom – Langzeitevaluation. Memo SmartKom 23, Bavarian Archive for Speech Signals (BAS), 2004.

F. Schiel and U. Türk. Wizard-of-Oz Recordings, 2006. In this volume.

M.A. Walker, D.J. Litman, C.A. Kamm, and A. Abella. PARADISE: A Framework for Evaluating Spoken Dialogue Agents. In: *Proc. 35th ACL*, Madrid, Spain, 1997.

M.A. Walker, D.J. Litman, C.A. Kamm, and A. Abella. Evaluating Spoken Dialogue Agents With Paradise: Two Case Studies. *Computer, Speech and Language*, 12 (3), 1998.

Cognitive Technologies

T. Frühwirth, S. Abdennadher:
Essentials of Constraint Programming.
IX, 144 pages. 2003

J. W. Lloyd:
Logic for Learning.
Learning Comprehensive Theories from Structured Data.
X, 256 pages. 2003

S. K. Pal, L. Polkowski, A. Skowron (Eds.):
Rough-Neural Computing.
Techniques for Computing with Words.
XXV, 734 pages. 2004

H. Prendinger, M. Ishizuka (Eds.):
Life-Like Characters.
Tools, Affective Functions, and Applications.
IX, 477 pages. 2004

H. Helbig:
Knowledge Representation and the Semantics of Natural Language.
XVIII, 646 pages. 2006

P. M. Nugues:
An Introduction to Language Processing with Perl and Prolog.
An Outline of Theories, Implementation, and Application with Special Consideration of English, French, and German.
XX, 513 pages. 2006

W. Wahlster (Ed.):
SmartKom: Foundations of Multimodal Dialogue Systems
XVIII, 644 pages. 2006