Natural and Intuitive Multimodal Dialogue for In-Car Applications: The SAMMIE System

Tilman Becker [1] and **Nate Blaylock** [2] and **Ciprian Gerstenberger** [2] and **Ivana Kruijff-Korbayová** [2] and **Andreas Korthauer** [3] and **Manfred Pinkal** [2] and **Michael Pitz** [4] and **Peter Poller** [1] and **Jan Schehl** [1]

Abstract.
We present SAMMIE, a laboratory demonstrator of an in-car showcase of a multimodal dialogue system developed in the TALK project[5] in cooperation between DFKI/USAAR/BOSCH/BMW, to show natural, intuitive mixed-initiative interaction, with particular emphasis on multimodal turn-planning and natural language generation. SAMMIE currently supports speech-centered multimodal access for the driver to a MP3-player application including search and browsing, as well as composition and modification of playlists. Our approach to dialogue modeling is based on collaborative problem solving integrated with an extended Information State Update paradigm. A formal usability evaluation of a first baseline system of SAMMIE by naive users in a simulated environment yielded positive results, and the improved final version will be integrated in a BMW research car.

1 Introduction

The TALK project investigates issues in multimodal dialogue systems: multilinguality, adaptivity and learning, dialogue modeling and multimodal turn planning. Our approach is based on an extended Information State Update paradigm. Some of these issues are demonstrated in SAMMIE, an in-car showcase developed in cooperation between DFKI/USAAR/BOSCH/BMW. The design of the SAMMIE system is based on a series of user studies performed in different Wizard-of-Oz settings as well as a usability evaluation of a baseline version of the laboratory demonstrator. SAMMIE will be integrated into a test car at BMW later this year.

The SAMMIE system provides a multimodal interface to an in-car MP3 player through speech and haptic input with a BMW iDrive input device, a button which can be turned, pushed dwn and sideways in four directions. System output is provided by speech and a graphical display integrated into the car's dashboard. An example of the system display is shown on the right in figure 1.

The MP3 player application offers a wide range of tasks: The user can control the currently playing song, search and browse the database by looking for any of the fields in the MP3 database (song, artist, album, etc.), search and select playlists and even construct and edit playlists.

SAMMIE supports natural, intuitive mixed-initiative interaction, with particular emphasis on multimodal turn-planning and natural language generation. The system puts the user in control of the interaction. Input can be given through any modality and is not restricted to answers to system queries. On the contrary, the user can provide new tasks as well as any information relevant to the current task at any time. This is achieved through modeling the interaction as a collaborative problem solving process, modeling the tasks and their progression as *recipes* and a multimodal interpretation that fits any user input into the context of the current task. Note that the user is also free in the use of multimodal input, such as deictic references accompanied by pointing gestures ("Play this title" while pushing the BMW iDrive button), and even cross-modal references without pointing as in "Play the third song (on the list)". To support these aspects of dialogue flexibility, we model dialogue context, collaborative problem solving and the driver's attention state by an enriched information state. Table 1 shows a typical interaction with the SAMMIE system, figure 3 shows the current setup for the user environment.

U:	Show me the Beatles albums.
S:	I have these four Beatles albums. [shows a list of album names]
U:	Which songs are on this one? [selects the Red Album]
S:	The Red Album contains these songs [shows a list of the songs]
U:	Play the third one.
S:	[song "From Me To You" plays]

Table 1. A typical interaction with SAMMIE.

The following section describes our system architecture. Section 3 presents our approach to extended multimodal interaction modeling, ontology based modeling and its impact on natural and intuitive dialogues. Section 4 briefly describes our Wizard-of-Oz experiments, the evaluation process and the results. Finally, section 5 summarizes some important lessons learned in development and evaluation of our system.

2 System Architecture

Our system architecture follows the classical approach [5] of a pipelined architecture with multimodal fusion and fission modules encapsulating the dialogue manager. Figure 1 shows the modules and their interaction: Modality-specific recognizers and analysers provide semantically interpreted input to the multimodal fusion module (interpretation manager in fig. 1) that interprets them in the context of the other modalities and the current dialog context. The dialogue manager decides on the next system move, based on its model of the

[1] DFKI, Saarbrücken, Germany, email: First.Last@dfki.de
[2] Saarland University (USAAR), Saarbrücken, Germany, email: {blaylock,gerstenb,korbay,pinkal}@coli.uni-sb.de
[3] Robert Bosch GmbH, Stuttgart, Germany, email: Andreas.Korthauer@de.bosch.com
[4] BMW Group Forschung und Technik, München, Germany, email: Michael.Pitz@bmw.de
[5] TALK (Talk and Look: Tools for Ambient Linguistic Knowledge) www.talk-project.org is funded by the EU as project No. IST-507802 within the 6th Framework program.

tasks as collaborative problem solving, on the current context and also on the results from calls to the MP3 database. The turn planning module then generates an appropriate message to the user by planning the actual content, distributing it over the available output modalities and finally co-ordinating and synchronizing the output. Modality-specific output modules generate spoken output and an update of the graphical display. All modules interact with the extended information state in which all context information is stored.

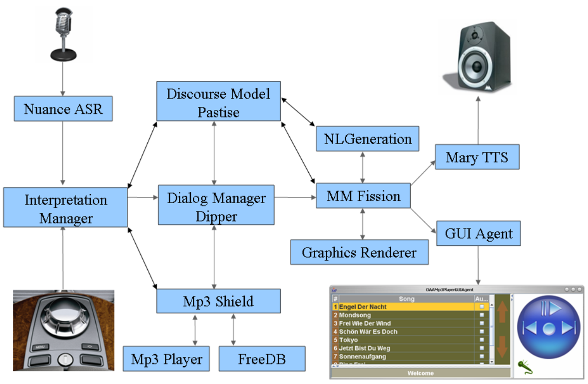

Figure 1. SAMMIE system architecture.

Many tasks in the SAMMIE system are modeled by a plan-based approach. Discourse modeling, interpretation management, dialogue management and linguistic planning, and turn planning in our system are all based on the production rule system PATE[6] [9]. originally developed for the integration of multimodal input. During the implementation of the baseline system, we found that PATE is also adequate for modeling other dialogue system components as it provides an efficient and elegant way of realizing complex processing rules. Section 3.5 elaborates more on PATE and the ontology-based representations it uses. In the following sections we will concentrate on our main areas of research.

3 Modeling Multimodal Interaction

Many dialogue systems that are employed today follow a state-based approach that explicitly models the full (finite) set of dialogue states and all possible transitions between them. The VoiceXML[7] standard is a prominent example of this approach. This has two drawbacks: on the one hand, this approach is not very flexible and typically allows only so-called system controlled dialogues where the user is restricted to choosing their input from provided menu-like lists and answering specific questions. The user never is in control of the dialogue.[8] For restricted tasks with a clear structure, such an approach is often sufficient and has been applied successfully. On the other hand, building such applications requires a fully specified model of all possible states and transitions, making larger applications expensive to build and difficult to test.

[6] PATE is short for (P)roduction rule system based on (A)ctivation and (T)yped feature structure (E)lements.

[7] `http://www.w3.org/TR/voicexml20`

[8] It is possible to build state-based systems without presenting a fixed menu hierarchy to the user. The user might even get the impression that they can interact quite flexible. But of course this creates considerable application complexity and implementation effort for all possible states and transitions.

In SAMMIE, we are following an approach that models the interaction on an abstract level as collaborative problem solving and adds application specific knowledge on the possible *tasks*, available *resources* and known *recipes* for achieving the goals. A planner, based on the PATE rule-interpreter [9] dynamically derives the next move of the system and then plans the details of the system output.

In addition, all relevant context information is administered in a central Extended Information State (EIS) module. This is an extension of the Information State Update approach [13] to the multimodal setting.

3.1 Extended Information State

The information state of a multimodal system needs to contain a representation of contextual information about discourse, but also a representation of modality-specific information and user-specific information which can be used to plan system output suited to a given context. The overall information state (IS) of the SAMMIE system is shown in Figure 2.

```
contextual-info:
[   last-user-utterance:
  :   [ interp               :  set(grounding-acts) ]
      [ modality-requested   :  modality            ]
      [ modalities-used      :  set(msInput)        ]

    discourse-history:
  :   list(discourse-objects)

    modality-info:
  :   [ speech   :  speechInfo  ]
      [ graphic  :  graphicInfo ]

    user-info:
  :   [ cognitive-load  :  cogLoadInfo   ]
      [ user-expertise  :  expertiseInfo ]                    ]
task-info:
[ cps-state         :  c-situation (see below for details) ]
[ pending-sys-utt   :  list(grounding-acts)                ]
```

Figure 2. Structure of the SAMMIE Information State

The contextual information partition of the IS represents the multimodal discourse context, i.e., the information communicated through the different modalities. It contains a record of the latest user utterance and preceding discourse history representing in a uniform way the salient discourse entities introduced in the different modalities (we follow the approach adopted in the SmartKom system [10]; the discourse model employs the three-tiered context representation proposed in [7] where the linguistic layer is generalized to a modality layer). The contents of the task partition are explained in the next section.

3.2 Collaborative Problem Solving

We see a dialogue system as a *conversational agent*—an autonomous agent which can communicate with humans through natural language dialogue. In order to support natural and flexible conversation, we need to model dialogue about the range of activities an agent may engage in, including goal selection, planning, execution, monitoring, replanning, and so forth. To achieve this our dialogue manager is based on an agent-based model which views dialogue as collaborative problem-solving (CPS) [4]. CPS follows a process similar to single-agent problem solving, but the agents negotiate to jointly determine objectives, find and instantiate recipes to accomplish them and execute the recipes and monitoring for success.

The basic building blocks of the formal CPS model are problem-solving (PS) objects, which we represent as typed feature structures. PS object types form a single-inheritance hierarchy, where children inherit or specialize features from parents. Instances of these types are then used in problem solving.

In our CPS model, we define types for the upper level of an ontology of PS objects, which we term *abstract PS objects*. There are six abstract PS objects in our model from which all other domain-specific PS objects inherit: objective, recipe, constraint, evaluation, situation, and resource. These abstract PS objects are used to model problem-solving at a domain-independent level and are taken as arguments by all update operators of the dialogue manager which implement conversation acts [4, 12]. The model is then specialized to a domain by inheriting and instantiating domain-specific types and instances from the PS objects. For example, the domain-specific objective *play-song* in the MP3 domain inherits all attributes from *objective* and adds a has-song slot, which is *single-slot* of type *song*. The operators, however, do not change with domain, which supports reasoning done at a domain-independent level.

3.3 Adaptive Turn Planning

In a multimodal dialogue system, the *fission* component is responsible for realising the planned system response as determined by the dialogue manager through multimodal output in an appropriate combination of the available output channels. This task comprises detailed content planning, media allocation and coordination and synchronization.

In SAMMIE, the fission task is realised by two modules - the *Turn Planner* and the *Output Manager*, whereas user- and modality-specific information which might be necessary for presentation planning can be obtained from another module called *Pastis*. Pastis is designed to provide and store user-, modality- and also discourse-specific information and forms, together with the dialogue manager's embedded information state as the *Extended Information State* of the system.

When the dialogue manager has finished processing the user input, the turn planner (TP) receives a bundle of CPS-specific conversational acts, representing the planned system response on an abstract level. TP then starts planning how to distribute given information over the available modalities, namely speech and graphics, but also determines on which level of detail information is going to be presented. As soon as TP has finished processing, it sends a sorted bundle of output messages, including both speech and graphic messages, to the output manager. The output manager then distributes the messages to the graphics renderer and/or the generation manager and synchronizes their output rendering.

The Turn Planner within the SAMMIE system is responsible for content selection and media allocation. It takes a set of CPS-specific conversational acts generated by the dialogue manager and maps them to modality-specific communicative acts. Therefore, relevant information on how content should be distributed over the available modalities (speech or graphics) can be obtained by

(a) the EIS/discourse module Pastis, which provides information about

  (1) the modality on which the user is currently focused, derived by the current discourse context.

  (2) the user's current cognitive load when system interaction becomes a secondary task (e.g., system interaction while driving). This information is modeled as a state variable with the possible states *low*, *mid*, *high* or *extreme*.

  (3) the user's expertise, which is represented as a state variable with the possible states *power-user* or *beginner*.

  Pastis also contains information about factors that influence the preparation of output rendering for a modality, like the currently used language (German or English) or the display capabilities (e.g., maximum number of displayable objects within a table).[9]

(b) a set of production rules that determine which kind of information should be presented through which modality. The rule set is divided in two subsets, domain-specific and domain-independent rules which together form the system's multimodal plan library. These rules are used to create the presentation plan for a turn, while dynamically taking the information listed in (a) into account.

Beside determining the best possible distribution of information, the turn planner also transforms domain-specific information that will be presented by speech to a representation we call *Reduced Knowledge Representation* (RKR) that can be interpreted by the Linguistic Planner. Such an RKR structure differs from its source structure through its more general, abstract surface structure which can be seen as an intermediate form between a pure ontological representation of a domain specific object and the logical form representation that the linguistic planner needs for deep generation with OpenCCG. Furthermore, an RKR structure specifies exactly the information that has to be presented to the user. Beside deriving the appropriate input for NLG, the turn planner is also responsible for computing and packaging the appropriate information to be presented by the display. Additionally, if there are alternatives on how to graphically realize content in different ways the turn planner needs to decide which one to take.

3.4 Spoken Natural Language Output Generation

Our goal is to produce output that varies in the surface realization form and is adapted to the context. We opted for using both a template-based and a deep grammar-based module in parallel. On the one hand, deep generation provides linguistically more powerful modeling, in particular it allows for more fine-grained and controlled choices between linguistic expressions in order to achieve contextually appropriate output. On the other hand, the template-based module can be developed faster and thus facilitates incremental development. It is also sufficient for classes of system output that do not need fine-tuned context-driven variation, such as simple cases of feedback. Our template-based generator can also deliver alternative realizations, e.g., alternative syntactic constructions *There are 3 songs by Nena* vs. *I found 3 songs by Nena*, referring expressions *Nena* vs. *the artist Nena*, or lexical items, e.g., *song* vs. *track*; however, the choice among alternative templates is made at random. The template-based generator is implemented by a set of straightforward sentence planning rules in the PATE system to build the templates, and a set of XSLT transformations to yield the output strings. Output in German and English is produced by accessing different dictionaries in a uniform way.

The grammar-based generator uses OpenCCG, an open-source natural language processing environment[3]. We have developed a

[9] Note, that for points (2) and (3) we haven't yet fully elaborated the rule based processing of these factors as this part of our ongoing work.

German OpenCCG grammar with basic coverage of German phenomena, and gradually extend it with respect to the phenomena encountered in the SAMMIE-1 and SAMMIE-2 corpora (see section 4).

3.5 Modeling with an Ontology

We use a full model of the application in OWL[10] format as the knowledge representation format in the dialogue manager, turn planner and sentence planner. This model includes the entities, properties and relations of the MP3 domain—including the player, data base and playlists. Also, all possible tasks that the user may perform are modeled explicitly. Note that this is a model that is *user centered* and not simply a model of the application's API. Actually, there is a separate module, the MP3-shield, that maps user tasks into possibly complex interactions with the connected applications, i.e., the database and the MP3 player itself.

The OWL-based model is transformed automatically to the internal format used in the PATE rule-interpreter. PATE employs Typed Feature Structures (TFSs) as basic internal data representation and XML for encoding all incoming and outgoing data as well as knowledge bases (production rules, type definitions). PATE is based on some concepts of the ACT-R 4.0 system [2]. Its main concepts, which PATE makes use of, are the goal-oriented application of production rules, the activation of working memory elements, and the weighting of production rules. In processing TFSs, PATE provides two operations that both integrate data and also are suitable for condition matching in production rule systems, namely a slightly extended version of the general *unification*, but also the discourse-oriented operation *overlay* [1].

Another important feature is the concept of multiple inheritance provided by the type system, as it allows to define different views on ontological concepts. Consider the concept *Song* and the different views our system ontology provides. A *Song* can be seen as a *Browsable-object* which allows generalization within the turn planning library over objects a user can browse. It can also be seen as a *Media-object* or a *Problem-solving-object* which are abstract concepts dialogue management can use for planning and execution. Or it can be seen as a *Mp3-resource* which denotes the domain affiliation of the concept. Thereby PATE provides an efficient and elegant way to create more abstract/generic presentation planning rules.

4 Experiments and Evaluation

To guide system development we have so far conducted two WOZ *data collection* experiments and one *evaluation* experiment with a baseline version of our system. The SAMMIE-1 WOZ experiment involved only spoken interaction, SAMMIE-2 was multimodal, with speech and haptic input, and the subjects had to perform a primary driving task using a Lane Change simulator [8] in a half of the experiment session. The wizard was simulating an MP3 player application with access to a large database of information (but not actual music) of more than 150,000 music albums (almost 1 million songs). The user had to carry out several tasks of two types: searching for a title either in the database or in an existing playlist, and building a playlist satisfying a number of constraints. In order to collect data with a variety of interaction strategies, we used multiple wizards and gave them freedom to decide about their response and its realization. In the multimodal setup in SAMMIE-2, the wizards could also freely decide between mono-modal and multimodal output. There is not enough time for the wizard to properly design the screen output on the fly. Therefore, we implemented modules supporting the wizard by providing several automatically generated screen output options the wizard could select from and inform the user about the database search results using the visual modality.

The following aspects of the setup were designed to elicit interactions more realistically resembling dialogue with an actual system [6]: The wizard and the user did not directly hear each other, instead, their utterances were immediately transcribed; the wizard's utterances were then presented to the user via a speech synthesizer, and parts of the user's utterances were sometimes deleted to simulate acoustic understanding problems and elicit clarifications.

Figure 3. The current setup of the user environment.

To explore the user acceptance, usability, and performance of a first baseline implementation of the SAMMIE multimodal dialogue system we have completed a usability evaluation. The evaluation tested the multimodal interaction of first-time users with the SAMMIE system in a laboratory experiment with a simulated driving task (figure 3 shows the setup). A sample of 20 subjects performed 32 tasks in total out of three scenarios with variation of the different kinds of dialogue modalities (spoken vs. multimodal). The users were asked to perform tasks which tested the system functionality, namely controlling player functions like stop/play/pause, next/previous track, playing a particular/random song/album/playlist, querying the music database (e.g., available songs/albums/artists/playlists) and administration of playlists (create/delete playlists, add/remove songs).

The evaluation analyzed the user's interaction with the baseline system and combined objective measurements like task completion and speech recognition accuracy observed in the experiments and subjective ratings from the test subjects by means of intermediate interviews during the session and by post-experimental questionnaires. The analysis of the experiments yielded an overall task completion rate (TCR) of about 90% for both spoken and multimodal interaction. Note that we allow up to four repeats for a user input. Relatively small differences in TCR between the dialogue modalities but considerable decrease of TCR for the more complex tasks have been observed. This was partly due to a relatively high out-of-grammar rate and consequently word error rate which showed the need to further increase the ASR grammar coverage.

Spoken dialogue was observed as the preferred interaction modality in the experiments. About 70% of the subjects chose speech when they had the free choice and less than 10% changed the modality during the task. Nevertheless, 40% of all subjects would prefer multimodal interaction in the long run when having more practice with the system. The general impression of the SAMMIE system was rated positively by most of the test subjects and changing the interaction modality was simple or very simple for 95% of the users. Also, the content and extent of the system's spoken and graphical output messages were rated mostly positively. But a detailed analysis of the ob-

[10] `http://www.w3.org/TR/owl-features`

jective measurements and subjective ratings also revealed some important issues for significant improvement towards the final version of the SAMMIE system.

The following section reflects some of the lessons learned from this evaluation.

5 Lessons Learned

To summarize the current state of our research and development, this section attempts to present the most important lessons learned to date and the biggest challenges for AI in the domain of natural, intuitive multimodal interaction.

- Reliable and robust ASR: When the system does not recognize the user's utterance, the biggest challenge for the user and the system is to know what went wrong. We are addressing this on two fronts: SAMMIE tries to give feedback about partially understood utterances with appropriate clarification requests. Also, we are using the data collection to expand the language covered by the recognition grammars.
- Natural and understandable speech synthesis: We have used the multilingual capabilities of the Mary TTS [11] to pronounce English song titles correctly, even when embedded in a German utterance. Also, markup of the speech output should allow context dependent prosodic cues such as pauses around elements to be clarified. To address users' demands, we need to increase the naturalness and acoustic quality of the TTS.
- System responsiveness: For command and control type interaction, e.g., stopping the MP3 player, time delays should be in the millisecond range. Thus we have added shortcuts directly from speech recognition to the player application, such that the full dialogue system is updated only in parallel or later. However, the context model must be kept synchronized with the course of the dialogue. Overall, reaction times (for complex input) have to be improved to satisfy the requirements of in-car systems.
- Close to real time system feedback: early on, we have added a multimodal microphone state, signaling to the user whether the system is ready and listening or currently processing input. The microphone shown on the GUI (see figure 1) toggles between red and green, accompanied by characteristic acoustic signals.
- Speech-centered multimodality: The evaluation confirmed that speech is the most important modality for our system, however the graphical and haptic modalities were used and valued by users. Since driving as the primary task leaves fewer and sometimes little attention (in particular, visual attention) for interacting with the SAMMIE system, turn planning must assure that core information is conveyed in speech and only additional information is presented on the display. Such additional information must also be accessible through spoken interaction.
- Adaptive, context-sensitive presentation: Natural, intuitive, multimodal interaction can be achieved through true mixed-initiative, collaborative interaction. Our system dynamically adapts its next move to give "intelligent" replies that (i) make the system's understanding transparent and (ii) ask for clarification when necessary and (iii) that query for more information by dynamically determining the most informative type of information, e.g. album name rather than artist name[11].

The last point represents the ultimate goal of our work: optimize all the above issues in the context of a flexible dialogue paradigm (Extended Information State Update) to achieve natural and intuitive multimodal interaction.

[11] We are currently experimenting with clustering algorithms to adequately summarize large sets of answers for a database query.

6 Conclusion

We presented an in-car multimodal dialogue system for an MP3 application developed in the TALK project in cooperation between several academic and industrial partners. The system employs the Information State Update paradigm, extended to model collaborative problem solving and multimodal context. It supports natural, intuitive mixed-initiative interaction, with particular emphasis on multimodal turn-planning and natural language generation to produce output adapted to the context, including the driver's attention state with respect to the primary driving task. We performed extensive user studies in a WOZ setup to guide the system design. A formal usability evaluation of the system's baseline version in a simulated environment has been carried out with overall positive results. A further enhanced version of the system will be integrated and evaluated in a test car, demonstrating the successful transfer of a range of AI techniques towards a real world application.

REFERENCES

[1] J. Alexandersson and T. Becker, 'Overlay as the basic operation for discourse processing in a multimodal dialogue system', in *Proceedings of the 2nd IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, Seattle, Washington, (August 2001).

[2] J. R. Anderson and C. Lebiere, *The Atomic Components of Thought*, Lea, June 1998.

[3] J. M. Baldridge and G.J. M. Kruijff, 'Multi-Modal Combinatory Categorial Grammar', in *Proc. of the 10th Annual Meeting of the European Chapter of the Association for Computational Linguistics (EACL'03)*, Budapest, Hungary, (April 2003).

[4] N. Blaylock and J. Allen, 'A collaborative problem-solving model of dialogue', in *Proc. of the 6th SIGdial Workshop on Discourse and Dialogue*, eds., L. Dybkjær and W. Minker, pp. 200–211, Lisbon, (September 2–3 2005).

[5] H. Bunt, M. Kipp, M. Maybury, and W. Wahlster, 'Fusion and coordination for multimodal interactive information presentation: Roadmap, architecture, tools, semantics', in *Multimodal Intelligent Information Presentation*, eds., O. Stock and M. Zancanaro, volume 27 of *Text, Speech and Language Technology*, 325–340, Kluwer Academic, (2005).

[6] I. Kruijff-Korbayová, T. Becker, N. Blaylock, C. Gerstenberger, M. Kaißer, P. Poller, J. Schehl, and V. Rieser, 'An experiment setup for collecting data for adaptive output planning in a multimodal dialogue system', in *Proc. of ENLG*, (2005).

[7] S. LuperFoy, *Discourse Pegs: A Computational Analyses of Context Dependent Referring Expressions*, Ph.D. dissertation, University of Texas at Austin, December 1991.

[8] S. Mattes, 'The Lane-Change-Task as a tool for driver distraction evaluation', in *Proceedings of IGfA*, (2003).

[9] N. Pfleger, 'Context based multimodal fusion', in *ICMI '04: Proc. of the 6th international conference on Multimodal interfaces*, pp. 265–272, New York, NY, USA, (2004). ACM Press.

[10] N. Pfleger, J. Alexandersson, and T. Becker, 'A robust and generic discourse model for multimodal dialogue', in *Proceedings of the 3rd Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, Acapulco, (2003).

[11] M. Schröder and J. Trouvain, 'The german text-to-speech synthesis system MARY: A tool for research, development and teaching', in *The Proceedings of the 4th ISCA Workshop on Speech Synthesis, Blair Atholl, Scotland.*, (2001).

[12] D. R. Traum and E. A. Hinkelman, 'Conversation acts in task-oriented spoken dialogue', *Computational Intelligence*, **8**(3), 575–599, (1992). Also available as University of Rochester Department of Computer Science Technical Report 425.

[13] D. R. Traum and S. Larsson, 'The information state approach to dialog management', in *Current and New Directions in Discourse and Dialog*, Kluwer, (2003).