The BabbleTunes System — Talk to Your iPod!

Jan Schehl, Alexander Pfalzgraf, Norbert Pfleger and Jochen Steigner
Deutsches Forschungszentrum für Künstliche Intelligenz GmbH
Stuhlsatzenhausweg 3
66123 Saarbrücken, Germany
{schehl,pfalzgraf,pfleger,steigner}@dfki.de

ABSTRACT

This paper presents a full-fledged multimodal dialogue system for accessing multimedia content in home environments from both portable media players and online sources. We will mainly focus on two aspects of the system that provide the basis for a natural interaction: (i) the automatic processing of named entities which permits the incorporation of dynamic data into the dialogue (e. g., song or album titles, artist names, etc.) and (ii) general multimodal interaction patterns that are bound to ease the access to large sets of data.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: input devices and strategies, natural language, prototyping

General Terms

Algorithms, Design

1. INTRODUCTION

Today, media players provide access to a large number of titles. However, the rather strict navigation hierarchies offered by current monomodal user interfaces are often inadequate for browsing through large amounts of data. Research in the area of human-computer interaction showed that multimodal interfaces combining classical interaction metaphors with speech-based interaction paradigms can overcome these drawbacks by decreasing the complexity of the overall interaction [2]. With the BabbleTunes system we present an interactive multimodal dialogue system that enables the intuitive, multimodal operation of an iPod or similar devices. Unique about the system is that it provides direct access to the entire music collection and that the user is able to refer to all kinds of named entities during all stages of the interaction. Moreover, the system also supports the full-range of multimodal interaction patterns like deictic or cross-modal references.

ICMI'08, October 20–22, 2008, Chania, Crete, Greece.

The interactive system is based on the ODP framework (see section 2) which is a generic framework for the implementation of multimodal dialogue systems, and greatly facilitates the development of such systems. An important feature of BabbleTunes is the speech-based access to all musical contents: new and often multilingual content can be accessed through an automatic preprocessing step which generates phonetic transcriptions of foreign-language entities (like titles and names) that can be processed by the German speech-recognizer and synthesizer. Section 3 provides an overview of this preprocessing step. In section 4 we will discuss the central interaction patterns that realize an intuitive access to large sets of data.

2. THE ODP FRAMEWORK

ODP (Ontology-based Dialogue Platform) defines both a generic modeling framework and run-time environment for multimodal dialogue applications supporting advanced dialogue phenomena (see section 4). For dialogue authoring, ODP imposes a model-based design approach formalized in terms of ontological concepts and relations. On the technical level, ODP enforces a programming model for interfacing with external components both in the presentation and backend layer (see figure 1). In this context, ODP provides interfaces to relevant third-party software, e.g., for speech recognition and speech synthesis. It is also easily extendable for additional functional components, e.g., for personalization purposes.

The run-time environment of ODP is based on a flexible middleware platform—an enhanced version of [5]—which connects system components following a hub-and-spoke architecture. Dialogue processing is accomplished by means of an ontology-based production rule system [4]. The system components of ODP are completely implemented in the Java programming language and thus operable on a number of system platforms.

2.1 ODP's Architecture

Conceptually, ODP's architecture comprises a number of functional components that deal with tasks like modality-specific interpretation, context-based interpretation, interaction and task management, target control, presentation management and modality-specific generation. All functional components are generally application-independent and are configured by respective models. The design of ODP is based on experience gained from several research and industrial projects and therefore aims at bridging the gap between these two worlds by addressing the following pri-

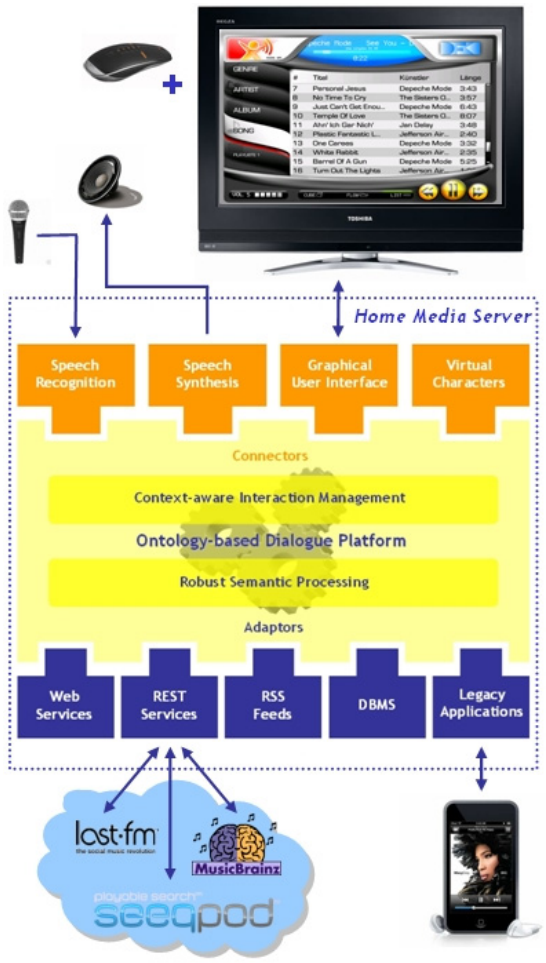

Figure 1: ODP enables intuitive home entertainment: In the I/O layer, the BabbleTunes system relies on microphone and speakers for verbal interaction and a graphical user interface presented on an LCD screen and controlled by an air mouse. ODP connects Apple's *iPod* and a set of Web 2.0 services for delivering intuitive access and added-value in the music domain.

mary goals: (i) scalability, robustness and performance, (ii) customizability (e.g., application tailoring, internationalization, personalization, configuration), and (iii) support for conversational dialogue. Envisioned enhancements for ODP include the establishment of an integrated development environment (IDE) for dialogue applications, the support of embedded operating platforms such as automotive systems and the support of dynamic reconfigurability by means of online upgrades.

2.2 Accessing Local and Web-based Functionality and Data in ODP

ODP's conceptual and technical design foresees the integration of heterogeneous *target applications* on a semantic level such as legacy applications or Web-based services (cf. figure 1). In the case of the BabbleTunes system, an Apple iPod is connected via the proprietary COM (Component Object Model) interface of Apple iTunes. The stock of MP3 metadata accessible via iTunes is lifted to an ontological level in a pre-processing step (cf. section 3), thereby becoming the system's central knowledge base. Hence, during run-time only iTunes' MP3 playing functionality is accessed.

Besides providing multimodal access to music data on the user's iPod, the BabbleTunes system extends local music data with online information from the music domain. To this end, we incorporated the capabilities of a set of Web-based services:

Audioscrobbler provides access to the community data of *last.fm* which records the listening behavior of its users and delivers, e.g., information on the similarity of artists. (`http://www.audioscrobbler.net/`)

MusicBrainz provides large amounts of community-edited metadata for the music domain with remarkably broad coverage and quality. (`http://musicbrainz.org/`)

SeeqPod is a search engine for streamable audio data in the Web. (`http://www.seeqpod.com/`)

Amazon offers access to its underlying product database including, for example, information on user ratings for albums. (`http://aws.amazon.com/`)

Information provided by these services is requested in the user's demand and consistently incorporated into the local knowledge of the system following the attitude of Web 2.0 for "remixing" available data. User requests such as *"Show me similar artists to 'U2'."* or *"Recommend a rock artist."* are answered by querying different online sources and combining their results on the semantic level.

The aforementioned services are connected via their REST interfaces to the run-time framework of ODP by means of respective *adaptors*. These adaptors specify means to invoke a service and how to translate ontological data to service-specific data formats and vice versa. At run-time, a concrete service or a composition thereof is selected on the basis of the semantic description of the user's request in a rule-based fashion. In this context, we follow an approach of design-time service composition in contrast to the automatic composition method presented in [6]. This is justified by a rather small set of available services and respective combinations and outranges run-time composition in terms of performance.

3. PROCESSING OF NAMED ENTITIES

Within the domain of a music player, a large variety of multilingual terms can be a subject matter of the processing. Since the BabbleTunes system was designed to operate in German, we focused on the challenge to make English items accessible within the system. As the BabbleTunes system also uses speech output, the challenge of processing multilingual data concerns speech recognition (ASR) and speech synthesis (TTS).

In order to make English items addressable by a German based language model of an ASR, the first task is to identify those items within the music database that stem from English. To this end, a Java-interface to an n-gram based text categorization tool was employed (*JTextCat;* `http://www.jedi.be/JTextCat/`) that can be trained on domain-specific data. To make the statistical-based text categorization more robust, an additional algorithm was employed that used German and English speech synthesis dictionaries to support the categorization by the language guesser.

The context of a specific named entity was also considered for categorization. For example, if there was no clear decision for a specific item (i.e., the title of a song) to be of English or German origin, the categorization of other titles from the same album or the same artist were included into the decision process. For the speech recognition, we use a grammar-based language model which was evaluated in the TALK project [2] and in which various commands accessing the functionality of a media player are implemented as sentence patterns. Dynamic data like song names, album names, artist names and genres that depend on the composition of a specific database, were collected within specific sub-grammars and linked the main language model. Having identified an English item within the database, we need to make this item addressable by the German speech recognizer by generating a pronunciation dictionary that maps the pronunciation of English items to the phoneme set of the German ASR. To this end, a transcription for a specific item that was identified as English was retrieved from the FreeTTS US English lexicon. As the phoneme set of the German ASR does not include English phones, we employed a mapping between the transcriptions from the lexicon and the phoneme set of the speech recognizer to generate appropriate transcriptions for a specific item (see Table 1).

English Phone	Mapped Phone	Example
ð	d	brother
θ	d	thread
w	v	web
ɹ	ʁ	remote
æ	ɛ	trap

Table 1: Example mappings of English phones.

The resulting transcriptions for the English items were not only used for speech recognition but also for the speech synthesis as the speech output of the BabbleTunes system is also German based. Additionally to the mappings on the phoneme-level, we implemented a domain-specific heuristic to ease the speech-based access to certain named-entities in the music domain. If a song title contains additional information in brackets, like for example 'All good things (feat. Chris Martin)', our algorithm not only expands the abbreviation 'feat.' to 'featuring' but also allows the user to omit this part of the named entity in his speech input.

4. INTERACTION PATTERNS

The BabbleTunes system supports a wide variety of multimodal interaction patterns and pursues a collaborative interaction paradigm with the ultimate goal to ease the access to large databases. Furthermore, the system employs a set of domain-adapted graphical patterns for displaying and accessing large amounts of data. In this section we will first provide a brief sketch of the primary scenario the system is designed for and then discuss central aspects of the multimodal interface.

4.1 The Scenario

The BabbleTunes system has been designed for home environments and envisions the users sitting in their living room on their sofa and operating the system using an air mouse like device with a built-in microphone. In this scenario, the GUI of the system is displayed on the central TV screen (see figure 1). The system permits to browse through the player's database and to control the basic functions like playing songs, creating and manipulating playlists as well as accessing online information via Web services.

4.2 Resolution of Multimodal References

The BabbleTunes system supports a wide variety of different types of multimodal references that enable the user to virtually access any data that is displayed on the screen. As discussed in section 3, a key aspect of the system is the possibility to access songs, albums, artists, etc. by their names. An example for this type of referencing would be *"Play 'One Day' by 'The Verve'."* However, the system also supports additional means to reference individual objects:

Deictic References *"Play this song."* [+ pointing gesture using the air mouse]

Exophoric References *"Play the fourth album."*

Anaphoric References *"Add this song to playlist 1."* [in the context of a song currently being played]

Implicit References *"Add 'Zoo Station'."* [in the context of a newly created or recently referenced playlist]

4.3 Disambiguation via the Display Context

Another crucial aspect of successful human-computer interaction is the incorporation of the physical context into the interpretation process. On the one hand, this context is needed in order to resolve references like *"the first"* or *"the next song"*. On the other hand, this context is also needed for disambiguation purposes. Previous work showed that users tend to employ potentially ambiguous references if the referenced object can be easily determined by considering the physical context [3]. E. g., if a music database comprises two or more songs by the name 'One', it is obvious that the one currently displayed on the screen should be played if the user just requests *"Play 'One'."* However, even though virtually all entities that are available in the field of the participant's view can serve as potential antecedents, not all of them are equally accessible (see, for example, [1]).

4.4 Constraint-based Browsing

On the graphical side BabbleTunes applies a novel constraint-based browsing metaphor which combines multi-tab and list browsing interaction patterns (see figure 2), where each tab represents one search category, e.g., artist or genre. Beside gesture-based browsing the multi-tab pattern is also employed for visual feedback on speech-based database requests.

Concerning browsing the user does not depend on a particular navigation hierarchy but can select any given *category tab* as a starting point. If, for example, the *album* tab is pressed, a list representation of all albums is shown. When the user selects one of the displayed albums, the system marks the *song* tab as active (colored white) and adds, where possible, information about the displayed album to relevant category tabs. Given the browsing situation as depicted in Figure 2, where the system presents the songs of the album *'Achtung Baby'* by *'U2'*, the system would present all known albums by *'U2'* if the user pressed the album tab (currently labeled with *'Achtung Baby'*).

Figure 2: Constraint-based browsing that employs multi-tab and list browsing.

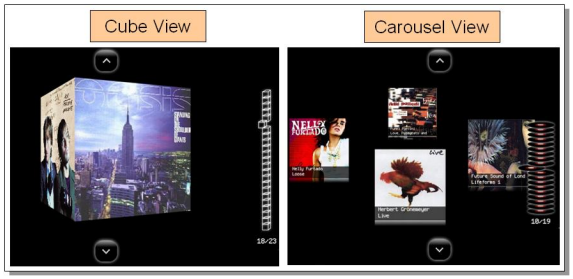

Figure 3: Alternative views on albums which can be selected by the user.

Beside list representation of result sets the system also employs two alternative views on albums, a rotating cube and a movable carousel, which can be selected by a button panel on the bottom of the GUI. As illustrated in Figure 3, both views make use of the album artwork provided by the iTunes interface. Note, that the objects focused within this views can be referenced by speech (e.g., *"Play this album."*).

4.5 Clustering Result-Sets

In cases where the user makes an underspecified request to search for a single object to be played, the BabbleTunes system applies a generic clustering mechanism which partitions a given result-set into several sub-sets based on relevant properties. Consider as an example the interaction illustrated in Figure 4, where the user requests to play a *rock* song which matches 144 songs in the database, which are represented by a list of artists interpreting all these songs. The number in brackets assigned to each row represents the number of songs interpreted by each artist. Thereby, the system is able to collaboratively and incrementally support the user in narrowing down the result-set until the searched-for object is found.

5. CONCLUSION

In this paper we introduced the BabbleTunes system, a full-fledged multimodal dialogue system that enables the operation of MP3-players in home environments. In particular, we discussed two important aspects of the system: (i) the automatic integration of named entities into the dialogue and (ii) the key interaction patterns supported by the system. The system has been presented at the CeBIT 2008 where a number of visitors took the opportunity to test the system even with their own iPod. After a short period of orientation

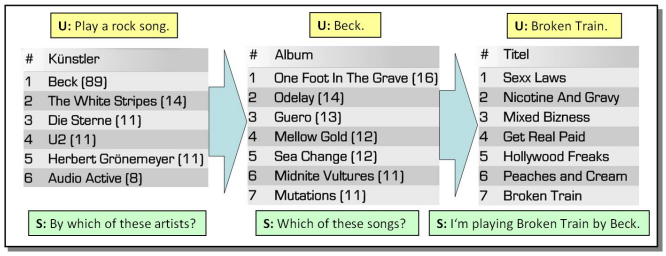

Figure 4: Interaction example illustrating the deployed clustering mechanism.

(i. e., the push-to-activate approach used by the BabbleTunes system for opening the microphone), most users showed a remarkable rate of successful interactions. Besides the home scenario, the system has also been employed in automotive scenarios: The current version of the system has, for example, been integrated into a Mercedes R-Class vehicle in order to provide access to an iPod. It is planned to integrate additional in-car functions like the air-conditioning into the system. Future work will also comprise a qualitative evaluation of the system with naïve users and the extension of the system so that it will be able deal with additional types of media files like movies or photographs.

6. ACKNOWLEDGMENTS

The project was funded by means of the German Federal Ministry of Economy and Technology under the promotional reference 01MQ07012 and by means of the EU 6th Framework Program under grant FP6-033502 (i2home). The authors take the responsibility for the contents.

7. REFERENCES

[1] D. K. Byron, T. Mampilly, V. Sharma, and T. Xu. Utilizing Visual Attention for Cross-Modal Coreference Interpretation. In *Lecture Notes in Computer Science: Proc. of Context-05*, pages 83–96, Heidelberg, Germany, 2005. Springer.

[2] A. Korthauer, F. Steffens, and H. Mutschler. D6.4 Final report on multimodal experiments - Part I: Evaluation of the final in-car system. Technical report, The TALK Project, 2007.

[3] W. Maass. *Von visuellen Daten zu inkrementellen Wegbeschreibungen in dreidimensionalen Umgebungen: Das Modell eines kognitiven Agenten*. Akademische Verlagsgesellschaft, 1996.

[4] N. Pfleger and J. Schehl. Development of Advanced Dialog Systems with PATE. In *Proc. of the Interspeech 2006/ICSLP*, pages 1778–1781, Pittsburgh, PA, 2006.

[5] N. Reithinger and D. Sonntag. An Integration Framework for a Mobile Multimodal Dialogue System Accessing the Semantic Web. In *Proc. of the 9th Eurospeech/Interspeech*, Lisbon, Portugal, 2005.

[6] D. Sonntag, R. Engel, G. Herzog, A. Pfalzgraf, N. Pfleger, M. Romanelli, and N. Reithinger. *SmartWeb Handheld - Multimodal Interaction with Ontological Knowledge Bases and Semantic Web Services*, volume Artifical Intelligence for Human Computing, pages 272–295. Springer, 2007.