

A Robust and Generic Discourse Model for Multimodal Dialogue

Norbert Pfeleger and Jan Alexandersson and Tilman Becker*

DFKI GmbH

Stuhlsatzenhausweg 3

D-66123 Saarbrücken, Germany

{pfeleger, janal, becker}@dfki.de

Abstract

We present an implemented, well-tested discourse model used in several mono- as well as multimodal dialog systems. We provide a detailed description of the underlying representation and show how such challenging phenomena like cross-modal anaphoric expressions and ellipses are processed.

1 Introduction

An important characteristic of dialog systems processing input from several modalities (e.g., speech and gesture) is the great extent of uncertain data such systems are faced with. In addition to the recognition modules producing many different hypotheses, semantic ambiguities arise while analyzing those hypotheses. The resolution of those ambiguities and the selection of the correct hypothesis are important tasks in order to achieve a successful man-machine interaction.

The interpretation of spontaneous speech without accessing some kind of discourse history will often lead to ambiguous results. Especially the interpretation of vague, reduced, or partial expressions, e.g., elliptical or anaphoric expressions, must consider a discourse history. In this paper we present a context model for multimodal dialogs partially extending the ideas presented in [LuperFoy, 1991; Salmon-Alt, 2000]. It builds a unified representation of multimodal user and system contributions and hence supports the resolution of cross-modal anaphoric expressions in a simple and generic manner.

Our context model is part of long time effort: to develop a generic discourse processing module [Pfeleger, 2002] contributing to the DFKI core dialog backbone for multimodal dialog systems.

The paper is organized as follows: In the next section we sketch our dialog backbone and its main data flow. Section 3 describes the challenges for a multimodal discourse module.

*The research within SMARTKOM presented here is funded by the German Ministry of Research and Technology under grant 01 IL 905. The responsibility for the content is with the authors. Thanks to Stephan Lesch and Massimo Romanelli for help with implementation issues and to the two anonymous reviewers who pointed at several unclear passages in an earlier version of this paper.

We give a detailed description of our implementation in section 4 and 5 and show how the module functions in section 6.

2 The Dialog backbone

The topic of this paper describes a part of a larger, long term goal: to develop a reusable multimodal dialog backbone. Until now, the complete backbone or parts of it have successfully been deployed in more than five different projects ranging from mono-modal typed input/output (text-only) to multimodal (speech, gesture and facial expression) input/output. The examples presented below are taken from our largest effort: the SMARTKOM system. Within the SMARTKOM project, we use our backbone to realize three scenarios:

Public The public scenario is an intelligent telephone booth where, in addition to normal telephone, the user can send fax and email and browse, e.g., a movie database.

Home The home scenario comprises device control, e.g., TV and VCR as well as browsing an EPG database. In the home scenario, the system has two modes: “lean back” and “lean forward”. In the former, the user has no visual contact with the display, so the system has to adapt presentation to the modality speech.

Mobile The mobile scenario comprises of car travel and tourist assistance. In this scenario the user interface is running on a small PDA with limited processing power and display size.

In all three scenarios, the user communicates with an animated agent (Smartakus) who is guiding the user. All in all, the current version of the system comprises 43 functionalities.

The discourse modeler has evolved over several years and has, additionally to the discourse types in SMARTKOM, been successfully used for modeling and processing negotiation dialogues within the VERBMOBIL project [Wahlster, 2000; Kipp *et al.*, 1999; Reithinger *et al.*, 2000].

Our architecture (see Figure 1) resembles a standard architecture for dialog systems. However, the module which is normally called dialog manager is split into two modules - the *action planner* and the *discourse modeler*. During processing, hypotheses from the different analyzers are brought together in the modality fusion. Each analysis component, including the discourse modeler, computes a score for each

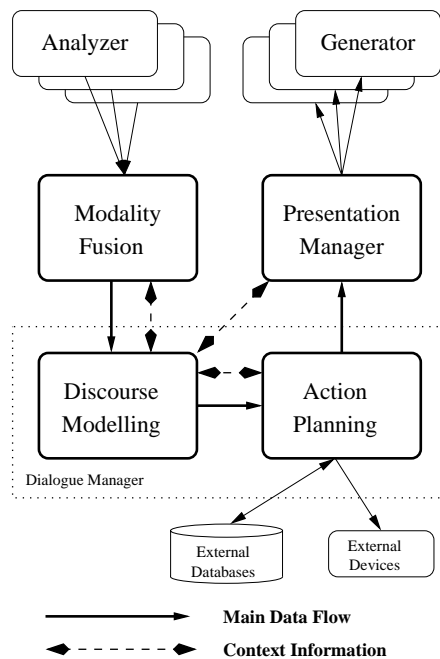


Figure 1: Architecture of the backbone

hypothesis. The score mirrors different aspects of the analysis chain. Whereas the score of the language analysis module [Engel, 2002] mirrors how well the path through the word lattice could be analyzed and mapped onto the semantic representation, the score of the discourse modeler is describing, e. g., how well the hypothesis fits the context. In our backbone, the discourse modeler ranks and selects, based on the scoring from all analysis components, the most probable hypothesis which is then passed on to the action planner. The action planner eventually accesses connected services and/or devices before it passes presentation goals on to the presentation manager and publishes expectations regarding the following user input. The expectations are important since they guide the interpretation process of the discourse modeler. This is especially important for elliptical phenomena [Löckelt *et al.*, 2002]. The presentation manager plans the output and presents it to the user depending on the available modalities.

For communication within the backbone we use a common representation consisting of syntactic information and instances of our domain model. In SMARTKOM, our ontology (see [Gurevych *et al.*, 2003]) is developed using OIL [Bechhofer *et al.*, 2001] and contains a complete description of our domain including (typed) processes and objects (abstract and real), e. g., [Russel and Norvig, 1995; Baker *et al.*, 1998]. Throughout this paper we will use the notion of *application object* for a “top object” in the ontology. Application objects are composed by *subjects* and can be thought of as complete descriptions of some action, e. g., browse a particular database, or switch on a particular device. Subjects on the other hand are roughly atomic objects, e. g., numbers, names and time expressions. We view the instances of the domain model as typed feature structures, which allows for manipulation with unification and unification-like opera-

tions. Since the discourse modeler functions as the central repository for discourse information, all modules within the backbone communicate with it.

3 Discourse Phenomena in Multimodal Dialogs

In general, a module for processing discourse has to fulfill two major tasks:

- (i) the enrichment and validation of intention hypotheses
- (ii) the resolution of referring expressions

The first task is involved in the process of determining the intended intention hypothesis out of a set of possible user intention hypotheses. The discourse modeler receives a sequence of intention hypotheses which have to be validated and enriched if possible with information of the preceding discourse. The second task occurs if the user has uttered a referring expression that is not accompanied by a deictic gesture. The discourse modeler then receives a request for resolving the referring expression.

Both tasks need access to a multimodal contextual representation of the preceding discourse (including both the user and the system contributions). Figure 2 shows a dialog excerpt¹ exemplifying the two tasks of the discourse modeler.

U1: *I'd like to see a movie tonight.*

S1: [Displays a list of movies] *Here [\nearrow] you see a list of the films running in HEIDELBERG.*

U2: *Hmm, none of these films seems to be interesting... Please show me the TV program.*

S2: [Displays a list of films] *Here [\nearrow] you see a list of broadcasts running tonight.*

U3: *Then tape the first one for me!*

Figure 2: Dialog excerpt 1

During analysis of U2, the discourse modeler receives a set of hypotheses. These hypotheses are compared and enriched with previous discourse information, in this example stemming from U1 and S1. Although U2 has a different topic than U1 (U1 requests information about the *cinema* program, whereas U2 concerns the *TV program*), the temporal restriction (*tonight*) of the first request is transferred into the interpretation of the second request.

In general, this propagation of information from one discourse state to another is obtained by comparing a current intention hypothesis with previous discourse states and by enriching it with consistent information if possible. For each comparison, a score has to be computed reflecting how well this hypothesis fits into the current discourse state. For this purpose an operation called OVERLAY is introduced in [Alexandersson and Becker, 2003].

¹The arrow \nearrow indicates a deictic gesture that points to some entity on the display

The second task is related to the determination of the intended referents for referring expressions in multimodal dialogs. Consider for example U3, where the intended referent for the referring expression *the first* is not found in its preceding linguistic discourse. However, it can be found in the visual context of the utterance U3, since the system has presented the list of films on the screen. This example shows the need for a context representation that allows access to the objects of the textual discourse as well as the visual context.

The context representation must provide access not only to the objects it comprises, it must also provide access to compositional information of these objects to permit the resolution of *partial anaphoric* expressions like the one in U3.

4 A Generic Multimodal Discourse Model

4.1 Context Representation

Our approach to discourse representation is based on that of [LuperFoy, 1991] and [Salmon-Alt, 2000]. We extended the three-tiered context representation of [LuperFoy, 1991] by generalizing her linguistic layer to a *modality* layer (see Figure 3). Additionally, we have adopted some ideas from [Salmon-Alt, 2000] by explicitly representing compositional information of discourse objects (see Figure 4). The advantage of our approach to discourse representation lies in the *unified* representation of discourse objects introduced by the different modalities. As we show below, this allows for, e.g., cross-modal reference resolution. The context representation of the discourse modeler consists of three levels which are described in the following sections.

Modality Layer

An object at the modality layer (henceforth MO) encapsulates information about the concrete realization of a referential object depending on the modality of presentation (see for example *LO2* of figure 4). Corresponding to the three different types of presentation, the modality layer is an accommodation of the different types of objects introduced by the different modalities:

- *Linguistic Objects* (LOs) For each occurrence of a referring expression in a generated or interpreted utterance one LO is added [LuperFoy, 1991]
- *Visual Objects* (VOs) For each visual presentation of an object that can be referred to one VO is added
- *Gesture Objects* (GOs) For each gesture performed either by the user or the system a GO is added

Each modality object is linked to a corresponding discourse object.

Discourse Object Layer

The central layer of the discourse model is the discourse object layer. There a discourse object (DO) represents a concept which potentially serves as a referent for referring expressions, including objects, events, states and collections of objects. Every time a concept is newly introduced into discourse by speech, a DO is created. A DO is also created for directly perceived concepts, e.g., graphical presentations [Salmon-Alt, 2000].

Each DO relies on two classes of information, (i) modality specific information, and (ii) domain information. MOs at the modality layer *mention* a DO, but are only able to modify the DO with respect to the domain model by adding new information to the corresponding domain representation. For example, a LO with syntactic gender-marking *female* (see *LO2* in figure 4) constrains the linguistic relations it enters, but does not affect its corresponding DO. Note that DOs are unique at the discourse layer. For each concept introduced during discourse there exists only one DO regardless of how many MOs mention this concept. The domain information of a DO represents the current information state of a DO in terms of the domain model. For each mention of a DO there exists at the domain object layer a corresponding application object or subobject. To provide access to these instances of the domain model, the domain information consists of a list of pointers to these instances. Additionally, the domain information includes a unified representation of these instances, that is used for the identification of identical DOs and for the resolution of referring expressions.

The compositional information of DOs representing collections of objects is provided by partitions [Salmon-Alt, 2000]. Such partitions are based either on perceptive information (e.g., the set of movies visible on the screen) or discourse information (e.g., “*Do you have more information about the first and the second movie*” in the context of a set of movies presented on the screen). Each element of a partition is a pointer to another DO, representing a member of the collection. The elements of a partition are distinguishable from one another by at least one *differentiation criterion* ($v(DC)$), like relative position on the screen, position within a set, size, color, etc. Within a partition, one element at most may be in focus, according to gestural or linguistic salience. Figure 4 depicts a sample configuration of a discourse object (DO2) with a partition.

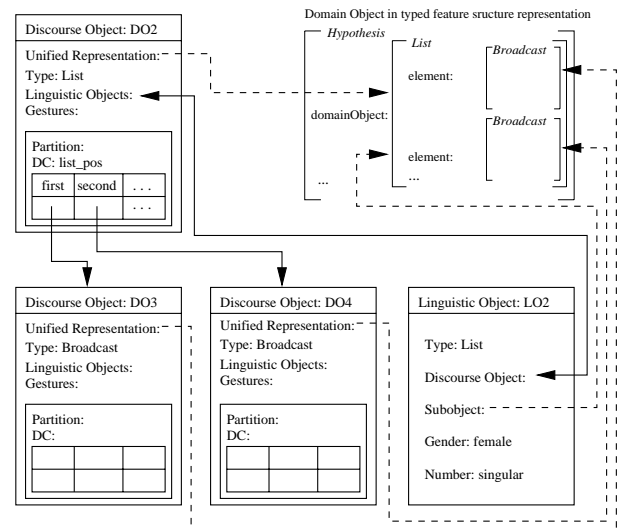
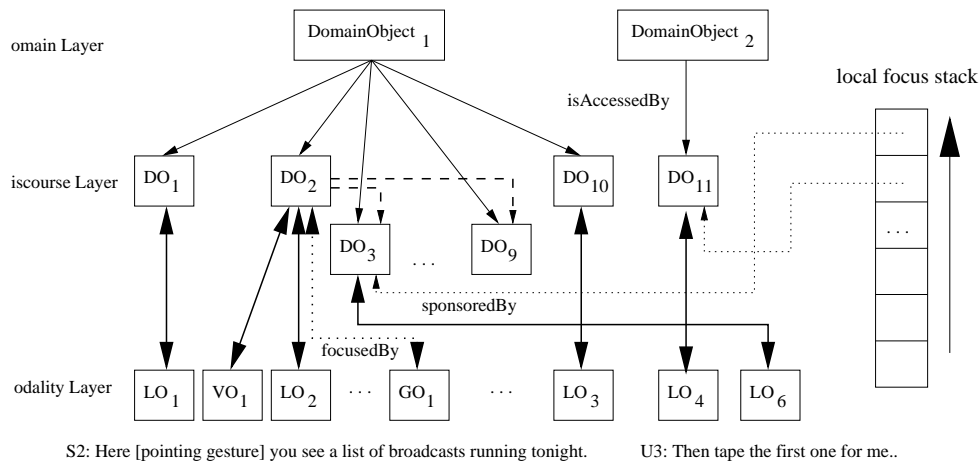


Figure 4: Discourse Objects



S2: Here [pointing gesture] you see a list of broadcasts running tonight. U3: Then tape the first one for me..

Figure 3: The Multimodal Context Representation

Domain Object Layer

The domain object layer provides the mapping between a DO and instances of the domain model. Instances of the domain model provide a semantic representation of actions, processes, and objects. In SMARTKOM, our domain model is described in OilEd [Bechhofer *et al.*, 2001] and provides a type hierarchy.

5 Focus Tracking

Generally, there are different ways of modeling the structure of dialog. In SMARTKOM, the focus structures are imposed by the action planner. However, our approach is not limited to this flat structure of discourse representation, but allows for hierarchical structuring too.

We differentiate between a *global focus* that represents the structure of discourse and a *local focus* that provides access to discourse objects. The global focus is represented by a *stack*² of (global) focus spaces, where each focus space covers the turns of the dialog participants belonging to the same topic. A focus space additionally contains a pointer to its corresponding local focus. The local focus contains pointers to discourse objects that are antecedent candidates for later reference. The DOs are ordered by salience.

5.1 Structuring Discourse

We use simplified, non-hierarchical *initiative-response units* (IR-units, see [Ahrenberg *et al.*, 1991]) to model a dialog structure that groups related user and system turns topically together (see section 5.2) and that restricts the access to possible referents in an appropriate way. When for example the user takes the initiative by requesting some information, the system can either respond to this request by providing the requested information or by taking the initiative and starting a

²Although we use the common terminology *stack* for these structures, we would like to emphasize, that they are not stacks in the strict sense. Objects are pushed onto the top of the structures. However, our experience is that nothing is really popped from the top, but rather (1) moved from somewhere further down to the top, or (2) removed from the structure because it is “decayed”.

sub-dialog that does not necessarily belong to the same topic. However, after the sub-dialog is finished, the system will provide (if possible) the requested information and by doing so it closes the IR-unit.

5.2 Global Focus

The global focus stack represents the topical structure of the discourse. For each topic introduced during discourse a global focus space is set up. Such a global focus space comprises information about its topic, its corresponding local focus stack, and the turns it covers. We distinguish two types of turns: (i) *UserTurns* containing the analysis of a selected hypothesis and (ii) *SystemTurns* containing the visually and linguistically presented information (including gestures performed by the animated agent).

Whether a new turn belongs to an already existing focus space or opens a new focus space depends on both its topic compared to the topic of the available focus spaces and the recent initiative-response (IR) structures.

The focus space at the top of the global focus stack is the currently active focus space. It serves as the preferred focus space for integrating new turns. A global focus space that is currently not the focused one is still open and therefore ready to be extended with new turns, if it comprises an open IR-unit. Note that an IR-unit is open unless an appropriate response has been given. In contrast, a focus space – not being the focused one – is closed and cannot be extended anymore, if it comprises only closed IR-units. A focus space can be *reactivated* by an explicit mention of its topic or its focused discourse object (e. g. by an explicit mention of its topic). A reactivated focus space is moved from its current position to the top. Where it initially provides only access to the focused discourse object of its local focus stack.

5.3 Local Focus

The local focus stack consists of pointers to discourse objects (see section 4.1) representing the set of possible referents ordered by salience.

For each user or system turn concerning a specific topic, the local focus stack is extended with all mentioned concepts which serve as potential referents. Their identification involves different processing steps depending on whether the processed turn stems from the user or the system.

Discourse objects mentioned by the user by speech are identified by analyzing the syntactic information of the selected (best scored) intention hypothesis. This syntactic information consists of lexical elements containing a part-of-speech tag, linguistic features (like number, gender) and a reference to a subobject of the semantic representation for each linguistic constituent. The referenced subobject describes the corresponding instance of the domain model for a linguistic constituent.

For the identification of discourse objects in a system turn, we distinguish between different presentation modalities. For each system turn two representations are processed: (i) the sequence of potential referents being constituents of the system utterance and (ii) a representation for the objects that are visible on the screen.

6 Discourse Processing

In this section we present and discuss our two main tasks: enrichment and validation of intention hypothesis on the one hand, and resolution of referring expressions and ellipses on the other. Our algorithms make heavy use of the three-tiered representation of discourse context. To enhance performance in time and precision, we utilize the focus structures for guiding the search of referents.

Many systems processing, e. g., spoken language and gestures are faced with competitive hypotheses representing different interpretations of the user contribution. The analysis components (linguistic/gestural analysis, modality fusion and discourse modeling), have the tasks to score each hypothesis, what we call *validation*. Our module additionally has to interpret the hypotheses in context - *enrichment*.

6.1 Enrichment and Validation

We use one single well-defined non-monotonic operation - OVERLAY - for enrichment and validation. The starting point for the development of this operation is the perception that instances of our domain model can be viewed as typed feature structures. Hence it is tempting to use unification for accommodation of the discourse state. However, since unification fails in case of conflicts, e. g., the user changes her mind and utters a new time for going to the movies, a non-monotonic operation has been developed (see [Alexandersson and Becker, 2003] for a detailed description).

Validation is performed by counting conflicting values and type clashes [Pfleger *et al.*, 2002] and finally by computing a single value representing how well the new structure fits the structure of the context. This information is collected during OVERLAY. Based on this score and the scores of the other analysis modules, the best hypothesis can eventually be selected.

We distinguish between full or partial utterances. An example of the former is a complete description of a user action, e. g., U4 in Figure 5, whereas partial utterances often but not

necessarily are elliptical responses to a system request. We handle these cases differently as described below.

U4: *What is on tv tonight.*

S4: [Displays a list of broadcasts] *Here [↗] you see a list of the broadcasts running tonight.*

U5: *What is running on CBS?*

S5: [Displays a list of broadcasts for CBS tonight] *Here [↗] you see a list of the broadcasts running tonight on CBS.*

U6: *and CNN?*

S6: ...

Figure 5: Dialog excerpt 2

Interpretation of Full Utterances

For, e. g., task-oriented dialog, there are many situations where information can and should be inherited from the discourse history as shown in the dialog excerpt in figure 5.

Due to spatial restrictions on the screen it may be impossible for the system to display every broadcast for all channels, e. g., in S4. The system therefore chooses some broadcasts of some channels. Clearly, the intention in U5 is to ask for the program on CBS *tonight* thus requiring the system to inherit the time expression from U4.

The OVERLAY-operation provides an elegant mechanism for inheriting information from the background for these cases. In fact, it is used as the basic operation for this kind of inheriting. Full utterances are processed by traversing the global focus structure and pick the focused application object in each focus space (if any). The further back the referent, the more the score gets punished.

Interpretation of Partial Utterances

The general procedure for interpreting partial utterance (e. g., U6) is to integrate the subobjects into a new application object of the same type as the object of the focused application. Enrichment is achieved by applying OVERLAY to the new application object and the application object in focus. We consider three cases that cause a partial utterance in terms of initiative-response pairs (a detailed description for this is provided in [Löckelt *et al.*, 2002]):

(i) *The user has the initiative and relates her utterance elliptically to a previous request.* Here, the user might intend to change the value of an already set slot of an application object by uttering only this subobject and skipping the remaining constituents of the preceding request. The correct interpretation of such a subobject is a replication of the first request including the changed subobject. The identification of such a discourse ellipsis is based on either the absence of an *expected slot*, or a type clash of the *expected slot* and the subobject plus the simultaneous presence of a subobject of the same type in the previous user request.

(ii) *The system has the initiative and the user relates her answer elliptically to the system utterance.* The answer uttered by the user contains only the requested subobject (i. e. the user skips the constituents that are already mentioned in the question). Here, the user can include more information

than requested in her answer. The identification of an elliptical answer to the system request is based on the presence of an *expected slot* of the same type as one of the incoming subobjects.

(iii) *Misinterpretation of the analysis module or non-cooperative user behavior*. This case is encountered when it is impossible to integrate the subobject into an application object; the subobject has to be returned without changes indicating that this particular interpretation has less or nothing to do with the current state of the discourse. This is the default case if none of the previous identification patterns matched.

6.2 Resolving Referring Expressions

Typically, the identification of the correct antecedent for a referring expression takes place by a search over a list consisting of the potential candidates for the antecedent (e. g., [Grosz *et al.*, 1995]). This search is based on a number of resolution factors which are used to track down the correct antecedent. Factors employed frequently in the resolution process include number and gender agreement, semantic consistency, semantic and syntactic parallelism, proximity etc. These factors decompose into two classes concerning the properties of the candidates (like number and gender agreement and semantic consistency) and the structure of discourse (like syntactic and semantic parallelism, or proximity).

When the modality fusion finds a referring expression that cannot be resolved using the gestures performed by the user, it sends a subobject that is as specific (close to a full domain object) as can be inferred from the intra-sentential context, together with the linguistic features and partition information (if there is any) to the discourse modeler. The partition information consists of a differentiation criterion defining the type of the partition and a value for accessing the partition (e. g., differentiation criterion: `list_position` with value: `first`).

Determining the Referent

The search for the intended referent depends on the type of the referring expression. We are differentiating three types of referring expressions: (i) total anaphora, (ii) partial anaphora, and (iii) discourse deictic expressions. The distinction between total and partial anaphora shows how an anaphora is related to its referent (see [LuperFoy, 1991]). If an anaphora shares the meaning with its antecedent entirely it is called total anaphora. However, if an anaphora only incorporates part of the meaning of its referent it is called partial anaphora. Additionally, we differentiate between *linguistic* and *discourse sponsorship* relations that can be established between the anaphora and their antecedents. Linguistic sponsorship means that potential antecedents for an anaphora must be comprised by the linguistic context. Therefore it cannot depend on entities that are not explicitly mentioned in the previous discourse which constrains the list of potential sponsors significantly. Discourse sponsorship relationship can be established without consideration of the linguistic features of the entities being involved. This permits the resolution of anaphora – antecedent pairs showing a gender mismatch as it occurs in the German language for example.

For the resolution of total anaphoric expressions, the active

local focus stack is traversed. The first discourse object that satisfies the type restriction and that has been mentioned by a linguistic object with the same linguistic features is taken to be the intended referent. In this case there is a linguistic sponsorship relation [LuperFoy, 1991] established between the anaphora and its referent.

If no linguistic sponsorship relation can be established, the first discourse object that satisfies the type restrictions is taken to be the intended referent – as long as no other discourse object within the same turn both satisfies the type restriction and shares the linguistic features with the referring expression. In this case there is a discourse sponsorship relation established between the anaphora and its referent.

However, if no matching discourse object is found, the focused discourse objects of the other global focus spaces are tested. For each focus space, the focused discourse object is tested as to whether it fulfills the type restriction and shares the linguistic features with the referring expression. The first matching discourse object is taken to be the intended referent.

In case of a partial anaphoric expression, the search for the appropriate referent decomposes into two steps. Initially, the first discourse object showing compositional information for the differentiation criterion specified in the request of the modality fusion is selected. In the second step it is tested as to whether the discourse object which is referenced by the partition satisfies the type restriction of the request. If it does so it is taken to be the intended referent. Otherwise, the resolution process goes back to the first step and continues with the next discourse object showing compositional information for the differentiation criterion.

In case that the type of the referring expression could not be identified, the focused discourse object of the currently active local focus is tested as to whether it shares the linguistic features with the request. If it does, that discourse object is taken to be the intended referent, otherwise the referring expression is interpreted as being a discourse deictic one.

Example: Cross-modal Reference

Please consider again dialog excerpt in figure 2. Figure 4 depicts the focused discourse object D02 which represents the list of films displayed on the screen in S2. This discourse object was mentioned by speech ... *you see a list...* and was supplementary put into focus by the pointing gesture of the agent.

During analysis of U3 the modality fusion is not able to combine the referring expression *the first* with a gesture. However, it can infer from the intra-sentential context that the referent for this expression has to be a domain object of type `Broadcast` and additionally it can infer from the definite description *the first* that the referent has to be computed by an extraction of a partition. These information are sent to the discourse modeler.

The discourse modeler classifies the referring expression as being a partial anaphoric reference since feature `partition` is present in the request. Additionally, the differentiation criterion (henceforth DC) specifies that the intended discourse object providing the compositional information must have at least DC information of the type `list_pos`. The discourse object D02 (see figure 4) is the

first one that fulfills this requirement. The element of the partition for the DC value *first* points to the discourse object D03. Since this discourse object satisfies the type restriction of the request, it is taken to be the intended referent and returned to the modality fusion. Figure 3 shows the configuration of the discourse context after U3 has been analyzed.

7 Conclusions and Future Work

We presented a generic robust discourse module which has been developed for and used in several mono- as well as multi-modal dialog systems. Our “largest” system - SMARTKOM- is a multimodal system for which 43 different functionalities has been implemented. During the development we have tested the system on over 100 test dialogs. Despite the comparatively mature status of the system, future work will focus on the following items:

- Extending the focus handling. Especially the functionality of the local focus will be extended. The idea is to use spreading activation to move not just the mentioned object to the top of the focus structure, but related items as well.
- A related phenomenon is encircling gestures which will be added in the near future. There, all objects encircled should be put into focus.
- The complete system is currently given a evaluation which will reveal the performance of the *complete* system. Of course, for a formal evaluation of the discourse model covering different anaphoric expressions, ellipses and validation has to be done as well.

References

- [Ahrenberg *et al.*, 1991] Lars Ahrenberg, Arne Jönsson, and Nils Dahlbäck. Discourse Representation and Discourse Management for a Natural Language Dialogue System. Research Report LiTH-IDA-R-91-21, Institutionen för Datavetenskap, Universitetet och Tekniska Högskolan Linköping, August 1991.
- [Alexandersson and Becker, 2003] Jan Alexandersson and Tilman Becker. The Formal Foundations Underlying Overlay. In *Proceedings of the Fifth International Workshop on Computational Semantics (IWCS-5)*, Tilburg, The Netherlands, February 2003.
- [Baker *et al.*, 1998] Collin F. Baker, Charles J. Fillmore, and John Lowe. The Berkeley Framenet project. In *Proceedings of COLING-ACL*, Montreal, Canada, 1998.
- [Bechhofer *et al.*, 2001] Sean Bechhofer, Ian Horrocks, Carole Goble, and Robert Stevens. OilEd: a reasonable ontology editor for the semantic web. In *Proceedings of KI2001, Joint German/Austrian conference on Artificial Intelligence*, number 2174 in Lecture Notes in Computer Science, pages 396–408, Vienna, September 2001. Springer-Verlag.
- [Engel, 2002] Ralf Engel. SPIN: Language understanding for spoken dialogue systems using a production system approach. In *Proceedings of 7th International Conference on Spoken Language Processing (ICSLP-2002)*, pages 2717–2720, Denver, Colorado, USA, 2002.
- [Grosz *et al.*, 1995] B.J. Grosz, A. K. Joshi, and S. Weinstein. Centering: A Framework for Modelling the Local Coherence of Discourse. Technical Report IRCS Report 95-01, The Institute For Research In Cognitive Science, Pennsylvania, 1995.
- [Gurevych *et al.*, 2003] Iryna Gurevych, Robert Porzel, Elena Slinko, Norbert Pfeleger, Jan Alexandersson, and Stefan Merten. Less is more: Using a single knowledge representation in dialogue systems. In *Proceedings of the HLT-NAACL’03 Workshop on Text Meaning*, Edmonton, Canada, 2003.
- [Kipp *et al.*, 1999] M. Kipp, J. Alexandersson, and N. Reithinger. Understanding Spontaneous Negotiation Dialogue. In *Workshop Proceedings ‘Knowledge And Reasoning in Practical Dialogue Systems’ of IJCAI ’99*, pages 57–64, 1999.
- [Löckelt *et al.*, 2002] Markus Löckelt, Tilman Becker, Norbert Pfeleger, and Jan Alexandersson. Making sense of partial. In *Proceedings of the sixth workshop on the semantics and pragmatics of dialogue (EDILOG 2002)*, pages 101–107, Edinburgh, UK, September 2002.
- [LuperFoy, 1991] Susann LuperFoy. *Discourse Pegs: A Computational Analysis of Context-Dependent Referring Expressions*. PhD thesis, University of Texas at Austin, December 1991.
- [Pfeleger *et al.*, 2002] Norbert Pfeleger, Jan Alexandersson, and Tilman Becker. Scoring functions for overlay and their application in discourse processing. In *KONVENS-02*, Saarbrücken, September – October 2002.
- [Pfeleger, 2002] Norbert Pfeleger. Discourse processing for multimodal dialogues and its application in smartkom. Master’s thesis, Universität des Saarlandes, 2002. Forthcoming.
- [Reithinger *et al.*, 2000] Norbert Reithinger, Michael Kipp, Ralf Engel, and Jan Alexandersson. Summarizing multilingual spoken negotiation dialogues. In *Proceedings of the 38th Conference of the Association for Computational Linguistics (ACL’2000)*, pages 310–317, Hong Kong, China, 2000.
- [Russel and Norvig, 1995] Stuart Russel and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, Englewood Cliffs, NJ, 1995.
- [Salmon-Alt, 2000] Susanne Salmon-Alt. Interpreting referring expressions by restructuring context. In *Proceedings of ESSLLI 2000*, Birmingham, UK, 2000. Student Session.
- [Wahlster, 2000] Wolfgang Wahlster, editor. *VERBMOBIL: Foundations of Speech-to-Speech Translation*. Springer, 2000.