Towards Multilingual Protocol Generation For Spontaneous Speech Dialogues*

Jan Alexandersson, Peter Poller
DFKI GmbH, Stuhlsatzenhausweg 3, D-66123 Saarbrücken,
{alexandersson,poller}@dfki.de

Abstract

This paper presents a novel multi-lingual progress protocol generation module. The module is used within the speech-to-speech translation system VERBMOBIL. The task of the protocol is to give the dialogue partners a brief description of the content of their dialogue. We utilize an abstract representation describing, for instance, thematic information and dialogue acts of the dialogue utterances. From this representation we generate simplified paraphrases of the individual turns of the dialogue which together make up the protocol. Instead of writing completely new software, the protocol generation component is almost exclusively composed of already existing modules in the system which are extended by planning and formatting routines for protocol formulations. We describe how the abstract information is extracted from user utterances in different languages and how the abstract thematic representation is used to generate a protocol in one specific language. Future directions are given.

1 Introduction

VERBMOBIL is a research project aiming at a Speech-To-Speech translation system for Face-to-Face dialogues [Bub and Schwinn 1996, Bub, Wahlster, and Waibel 1997, Wahlster 1993]. In its first phase (1993 - 1996), a bilingual (English-German) system for time scheduling negotiation dialogues was developed. For its current second phase (1997 - 2000), a third language, Japanese, has been incorporated. Additionally, more than two people should be able to participate in the dialogue, a setting we call *multi-party*. For the second phase, a novel system feature is currently under development - *protocol generation*. The idea is to provide the user(s) with three kinds of protocols:

Progress Protocol The most *salient* parts of the dialogue are summarized.

Result Protocol The *result* of the negotiation is summarized.

Status Protocol The current *status* of the negotiation is summarized.

The first two are generated after the dialogue is finished. For the result protocol, a summarization of the goal reached in the negotiation is generated. The progress protocol serves as a simplified recapitulation of the progress of the dialogue, whereas the status protocol will, on demand, be generated during the ongoing dialogue. Its task is to deliver a brief description of the current status of the dialogue. All protocols must be generated in all three languages, putting extra requests on the protocol generation component: particularly, being as language independent as possible. In this paper we focus on the first of the protocol types listed above - the progress protocol.

*The research within VERBMOBIL presented here is funded by the German Ministry of Research and Technology under grant 01IV101K/1. The authors would like to thank Amy Demeisi, Michael Kipp, Norbert Reithinger and the anonymous reviewers for comments on earlier drafts on this paper.

2 An overview of VERBMOBIL

2.1 Some terminology

A turn is a contribution of one dialogue participant. It may be divided into segments, which sometimes resemble linguistic clauses like sentences. Basic processing entity for some components is the so-called *dialogue act* [Bunt 1981, Jekat et al. 1995, Alexandersson et al. 1997]. For this work we use a set of 18 dialogue acts, some purely illocutionary, e.g., *requesting a proposal* for a date. Some comprise propositional content, e.g., *proposing a date.* An important property of the dialogue act is its language independence – it should be possible to be used for the annotation of dialogues in any language. Linguistic information is encoded in an abstract data type, the so called *VIT* (VERBMOBIL *Interface Term*, [Bos et al. 1996, Dorna 1996]). A VIT is a semantic representation formalism following the *Discourse Representation Theory* (*DRT*) of [Kamp and Reyle 1993]. A VIT consists of a set of semantic conditions (i.e. predicates, roles, operators and quantifiers) and allows for under-specifications with respect to scope and subordination or inherent under-specifications. Each discourse individual is formally represented by a discourse referent (also called instance). Information about the individual is encoded by one (or more) VIT–condition(s), combining a predicate with the discourse referent (see section 4).

Propositional information (currently time expressions) is encoded in a knowledge representation language [Küssner and Stede 1995]. It is a quite surface–oriented language, but contains some interlingua-like expressions for, e.g., month-of-year (`moy`), weekdays (`dow`), time-of-day (`tod`) and part-of-day(`pod`). Figure 1 gives some examples. of so-called "tempex[1]-expressions".

The first of February	`[moy:2,dom:1]`
Two o'clock in thursday	`[tod:14:00,dow:thu]`
Tomorrow between 8 in the morning and 14 hour	`[dow:tomorrow, boundaries([tod:08:00], [tod:03:00,pod:pm], from_to)]`

Figure 1: Some tempex expressions

2.2 The VERBMOBIL system

The VERBMOBIL system is a flexible, speaker independent speech–to–speech translation system for spontaneous speech. To support the robustness of the overall system, the translation process is subdivided into several processing tracks. The most accurate translation track is a deep linguistic analysis in combination with semantic transfer and syntactic generation (see figure 2). When this track fails, the translation is performed by other shallow, translation components. In this paper we will just consider the dialogue act based analysis and transfer [Block 1997]. Effects of spontaneous speech like hesitations, corrected and revised utterance parts do not provide any translation- and thereby protocol- relevant information. They have to be recognized and filtered out during the analysis and/or translation process. In figure 2 a sketch of some of the linguistic components is given. The translation process consists of two tracks, the deep and the shallow translation. The deep linguistic translation track, whose modules all exchange linguistic information encoded in VITs, consists of three components: An HPSG–parser combined with a robust semantic component,

[1]TEMPoral EXpression

the semantic based transfer component [Dorna and Emele 1996] and the generation component [Becker et al. 1998], an efficient multi-lingual generator (some more details below).

The shallow track bases its translation on dialogue acts and the propositional content (currently time expressions). Based on the input string, the dialogue act is determined, which in combination with the propositional content is transfered using fixed templates. The DIAKON component exchanges contextual information with 15 different components. It consists of two sub–components, the Context Disambiguation component and the Dialogue component. The former is responsible for, e.g., the disambiguation of semantic predicates and the computation of time expressions. The latter, following a hybrid approach (more details in the next section), supports, for instance, the analysis component with top down predictions what dialogue act is next to come.

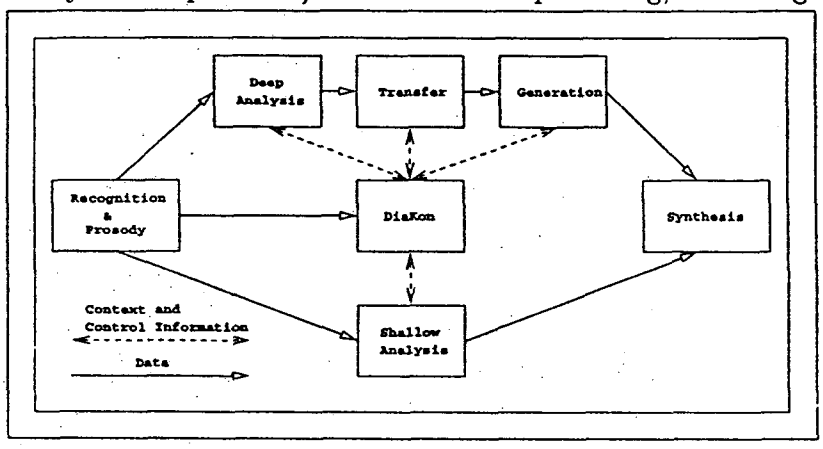

Figure 2: Overview of the VERBMOBIL system

3 Requirements for the protocol generation

The protocol may contain original system translations as well as paraphrases of the user utterances depending on the translation system's internal information about them. Additionally, thematically irrelevant parts of the dialogue, i.e. dialogue contributions not relating to the communicative goal, should not be reflected in the protocol. Moreover, some information is condensed in the protocol structure (in comparison to the original dialogue contributions) or removed from the protocol structure following different criteria of "protocol relevance". Example of the latter are utterances annotated with the dialogue acts e.g. FEEDBACK_* or DELIBERATE_* can under most circumstances be removed. We are putting extra attention to this, since the removal of utterances and turns must not threaten the correctness of the protocol (e.g. the dialogue must still reflect the actual negotiation). Parts of the dialogue, e.g., clarification sub–dialogues, can under some circumstances be left out from the protocol. Stereotypical dialogue phases like the greeting phase, are reflected in the protocol by meta comments. This procedure has the following advantages:

1. The structure of the protocol is very close to the original dialogue and therefore it is easy to follow for dialogue participants as well as for someone else.
2. The use of the original system translations into the language of the protocol instead of paraphrases simplifies the recapitulation of the dialogue for the dialogue participants later on.
3. The use of paraphrases of user utterances in the output language of the protocol emphasizes that the system has "understood" the user contributions.

Furthermore, by utilizing different sources of information, e.g., deep and shallow processing, we can always generate a protocol. The planning procedure of a protocol formulation depends on whether the original turn segment was given in the output language of the protocol or not. We have to keep in mind that the dialogue partners speak different languages which means that about half of the protocol has to be re–constructed (i.e. condensed and paraphrased) from utterances in the output language of the protocol. The other half of the protocol formulations may consist of utterances which were translated into the output language of the protocol by the system. Consequently, the planning procedure operates as follows:

System Translations If the system translated the user utterance into the output language of the protocol, this translation is chosen as the protocol formulation. In a system translation all irrelevant effects of spontaneous speech (e.g., hesitations, corrected or revised utterance parts) are already removed so that the selected protocol formulation is reduced to thematically relevant information.

Deep VIT–representation In the opposite case (i.e. the user utterance was spoken in the output language of the protocol) the original VIT–representation of the user input produced by the deep analysis component (if such a VIT exists) is used. The phrase is then produces by re–generating from the VIT. Again, effects of spontaneity are removed in such a VIT.

Tempex–VIT In all other cases (i.e. there is no deep analysis of the user utterance) we use the language independent tempex mechanism whose handling with respect to the planning of protocol formulations is described below.

Up to this point only the third type has been implemented.

4 Extraction of Protocol-Relevant Data

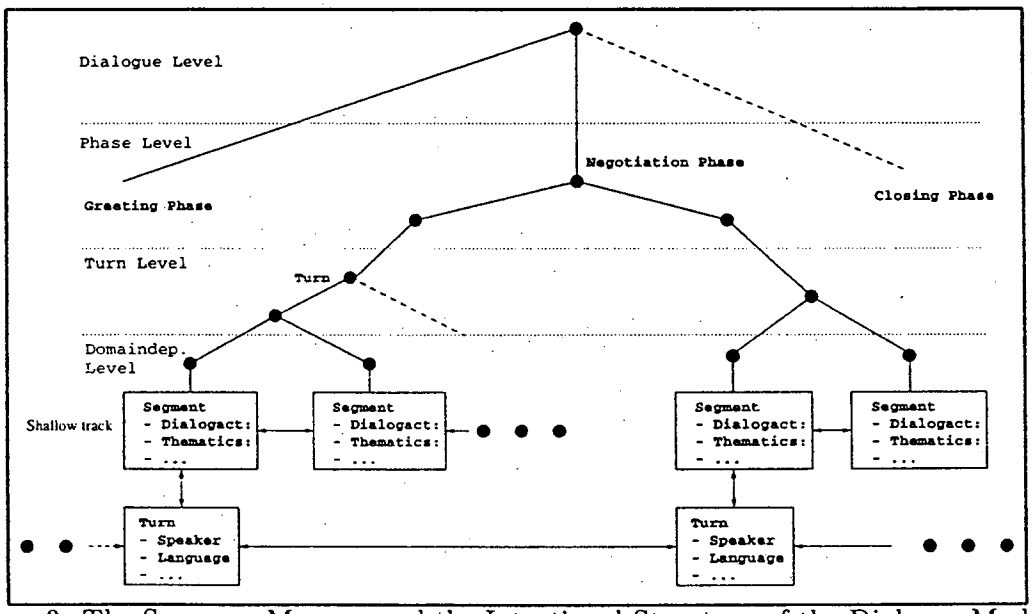

Figure 3: The Sequence Memory and the Intentional Structure of the Dialogue Module

For the extraction of protocol relevant data, we utilize a part of the DIAKON module, namely the *dialogue module* [Alexandersson, Reithinger, and Maier 1997], a hybrid component consisting of a dialogue memory, a statistical component, and a plan processor. Its processing is centered around dialogue acts [Jekat et al. 1995] – it is assumed that every utterance can be attributed one (or more) dialogue acts. Important parts of the dialogue module are the *sequence memory*, which contains data structures for turns and segments (see figure 3). Each turn keeps information like speaker and source language. In each segment information like the dialogue act, thematic information and VITs for different languages are stored.

4.1 The Plan Processor

We use the *plan processor* [Alexandersson and Reithinger 1997] for the selection of protocol relevant data. It incrementally traverses the sequence memory, building a structure which we call the *intentional structure*. It is a tree like structure mirroring different abstractions of the dialogue, like segment (dialogue act), turn (turn class), greeting phase. In figure 3 a sketch of the intentional structure is shown. It divides into 4 distinct levels, where the top-most spans over the whole dialogue, the next distinguishes the dialogue phases *greeting, negotiation* and *closing*. The third level connects segments within a turn and distinguishes its turn-class. Finally, the fourth implements (with some minor extensions) the dialogue act hierarchy. The leaves correspond to the utterance(s) of a turn. For each translation track, one instance of the processor is used, but the figure shows just one. The plan processor uses a set of plan operators (currently about 175) (see figure 4 and 5).

```
(defleaf feedback_acknowledge
  :goal (in-domain-dependent FEEDBACK_ACKNOWLEDGMENT ?in ?in)
  :constraints (first-in-turn (current-utterance))
  :actions (progn  ;; protocol relevance?
                   (cond ((not (single-utterance (current-utterance)))
                          (mark (current-utterance) :protocol-relevance nil)))
                   ;; dialogue phase
                   (mark (current-utterance) :dialogue-phase
                         (get-phase-from-context (current-utterance)))
  :leaf  FEEDBACK_ACKNOWLEDGMENT)
```

Figure 4: A Plan Operator for Processing Utterances of Type FEEDBACK_ACKNOWLEDGMENT

Both hand-coded as well as automatically derived operators from the VERBMOBIL corpus are used to build its structure. The operators are incrementally expanded left to right using a mixed top down and bottom up strategy. A plan operator can be attributed with `constraints` and `actions`. The constraints are used to check the relevance of a certain operator in a certain context, whereas the actions are mostly affecting the context. Examples of the latter are: marking an utterance as being protocol relevant or not, or setting the dialogue phase.

4.2 Central Contents of a Turn

By determining the central contents we remove unimportant segments and merge segments in such a way that the intended meaning of the turn is preserved. The plan processor performs this operation on two levels: based on segments and based on turns. By looking at, for instance, the dialogue act, it can be determined whether the segment can be removed or not. More abstract plan operators are responsible for removing whole turns. An example of the latter is clarification sub-dialogues. This, however, turns out to be very difficult when scaling up, due to irregularities in the dialogues and more problematic: recognition errors. In the current implementation, the plan processor performs two operations on the context: (*i*) Marking a segment as (not) relevant for the protocol, and (*ii*) Merging two or more segments into one.

Figure 4 shows a plan operator designed for processing an utterance which has been annotated with the dialogue FEEDBACK_ACKNOWLEDGMENT. The operator is designed for being applied when the utterance is the first one in the turn, and it is stated that, unless this utterance is the only one in the turn, it is marked as not relevant for the protocol. Another example of actions is shown in figure 5. This operator is used for merging two successive utterances of type ACCEPT_DATE into one. This is done when not more than one of the utterances contains propositional contents. To clarify the concept of "central contents" further, consider the following turn taken from our corpus (figure 6). If we use similar operators as in figure 4 for processing the second and third segment – as pointed out before, deliberations and feedbacks can under most circumstances be removed

```
(defplan accept-iterate
  :goal (in-domain-dependent accept-iterate ?in ?out)
  :actions (progn ;; reduction
                  (cond ((or (not (has-proposition (left-most (me))))
                             (not (has-proposition (right-most (me)))))
                         (merge (left-most (me)) (right-most (me)) :da 'accept_date)
                         (replan-turn))))
  :subgoals (:and (in-domain-dependent accept ?in ?tmp)
                  (in-domain-dependent accept ?tmp ?out)))
```

Figure 5: A Plan Operator for merging two ACCEPTs

– the protocol relevance of the segments is affected: The first three segments are considered as not relevant for the protocol at all, whereas the last two following segments (ACCEPT) are merged into one segment. For the generation of a progress protocol, just one segment would thus be

Transcription:
MAW004: <P> ja , (FEEDBACK_ACKNOWLEDGMENT)(*OK*)
da mu"s ich mal eben kucken . (DELIBERATE_EXPLICIT)(*I have to look*)
+/der/+ das ist ein<Z> <A> Samstag (DELIBERATE_IMPLICIT)(*That's a Saturday*)
das ist bei mir kein Problem <A> . (ACCEPT_DATE)(*That's no problem for me*)
<Ger"ausch> <P> +/neun/+ <P> neun Uhr ist kein Problem (ACCEPT_DATE)(*Nine o'clock is OK for me*)
Paraphrase:
MAW004: Neun Uhr passt bei mir (ACCEPT_DATE)(*Nine o'clock suits me*)

Figure 6: An example turn

considered, namely one expressing an acceptance of nine o'clock. The result of the generation does not necessarily contain the spoken words, but mirrors the illocution behind the utterance.

4.3 Turning a segment into a VIT

Preparing the input for the surface generator we want to prepare it in the same format that it is already able to cope with in the translation mode: VITs. Furthermore we prefer a format that it is usable for not just one language, but for all three. If we construct, e.g., German VITs for protocol formulations we can utilize the transfer component to transfer the VITs into any other VERBMOBIL language, and then make use of the surface generator as it is. This task is split into two steps, where the first consists of generating partial language independent VITs (henceforth tempex-VIT) on basis of the information in the selected segments (see below). The second step involves enriching the partial VITs with language dependent, e.g., verb-information (see section 5).

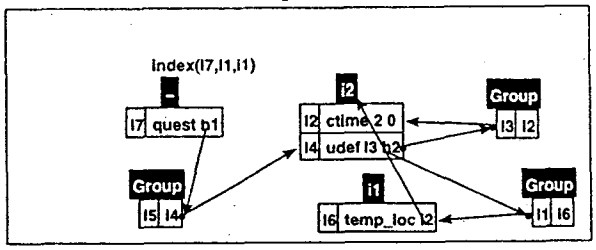

Figure 7: VIT-semantics of a time expression

Additionally, the plan processor draws more global inferences like, which dialogue phase a segment is part of. It also determines the turn class; information which we utilize while preparing the VITs, for instance when determining the sentence mood for a segment. For the utterance "How about at 2 o'clock" the corresponding information could be the dialogue act SUGGEST_SUPPORT_DATE and the tempex expression `[tod:02:00]`. A graphical representation of the partial VIT for this segment is shown in figure 7.

The index/3 is the entry point from where the VIT can be traversed. From the entry point we can reach the first grouping (15). It is pointing at a representation the instance i2, which is containing the representation for 2 o'clock (ctime). The udef predicate stands for indefiniteness, with scopus over ctime (via grouping 13). Finally, temp_loc is an under-specification for temporal or locational – the resolution of this under-specification is solved in the second step or by the surface generator.

4.4 Input to the syntactical generator

We can now produce the input structure for the generator. It contains on one hand general information about the dialogue (which is initially stored in the dialogue memory) and on the other hand an ordered list of protocol–relevant information of the individual segments of the turns. The following information is included in the abstract protocol representation:

- the date, time and location where the dialogue took place,
- the beginning time and the end time of the dialogue,
- a detailed list describing the thematic contents of all turns of the dialogue. Each of the turn descriptions includes the information about (*i*) speaker, and (*ii*) a detailed description of the individual segments of the turn each of which contains the following information: dialogue phase, dialogue act, sentence mood, turn class, a deep VIT of the original user utterance (if it exists), the original system translation, and the tempex–VIT as described above.

5 Surface Generation of the Protocol

The general parts of the representation produced by the DIAKON module can be handed over to the protocol more or less directly while the information about the individual turns and their segments is abstract and has, therefore, to be the source of a planning process of an appropriate protocol formulation. In order to keep the original ordering of turns and their segments intact, our module generates one protocol formulation for each segment. The planning of an appropriate protocol formulation for one segment has to consider different cases which require different planning procedures to come up with a formulation.

5.1 From Tempex–VIT to Protocol Representations

The processing proceeds on the semantic level based on the VIT–formalism but now switching to language specific operations. We defined a structured set of VIT-patterns whose task is to insert a verb into the tempex–VIT (which only contains time expressions) and its obligatory verb arguments in order to come up with a complete sentence. This final VIT-representation will then be handed over to the generation module of the VERBMOBIL system for verbalization. The core planning step consists of the selection and application of an appropriate VIT–pattern of a segment which is determined by the following three main criteria:

- The first criterion to find an applicable pattern is the dialogue act of the segment which means that for all 18 possible dialogue acts there is a structured list of applicable patterns.
- The second criterion is the sentence mood of the user utterance.
- The third one is the question whether the time expression contained in the tempex–VIT can be assigned directly to a verb argument position or not. This distinction is important because it requires different semantic handling of the tempex–VIT (e.g., the subject argument of the main verb).

Figure 8 shows an example of such a structured pattern list which contains only one alternative[2] for each possible parameter combination for the dialogue act SUGGEST_SUPPORT_DATE:

```
(etle 'SUGGEST_SUPPORT_DATE
 '(((:sentence-mood)
    (quest ((:tempex) (yes ((:abstr-tempex)
                            (yes VALUE (verb+ppron-subj+tempex-pp freihaben))
                            (no VALUE (verb+tempex-subj+bei-pp gehen_passen))))
                      (no VALUE #IMPOSSIBLE#)))
    (decl  ((:tempex) (yes ((:abstr-tempex)
                            (yes VALUE (verb+ppron-subj+tempex-pp-as-acc vorschlagen))
                            (no VALUE (verb+ppron-subj+tempex-acc vorschlagen))))
                      (no VALUE #IMPOSSIBLE#))))))
```

Figure 8: Example of a Pattern choice net

The entry is organized as a discrimination tree consisting of declaratively annotated alternating tests and values. The test-key :sentence-mood stands for the test of the sentence mood of a segment representation, :tempex checks for the existence of a tempex–VIT (which is not necessarily the case – as in figure 6, an ACCEPT can contain a temporal expression or not.) and :abstr-tempex checks the mentioned distinction whether the time expression of a tempex–VIT can be assigned directly to a verb argument position or not. Sentence moods above are *quest* (question) and *decl* (declarative). The keyword *VALUE* indicates , e.g., *(verb+ppron-subj+tempex-pp freihaben)*. This is a pattern function which will be applied on the tempex–VIT. In this case the *verb freihaben* and a personal pronoun in subject position (*ppron-subj*) will be added to the tempex–VIT which will presumably be realized as a prepositional phrase (*tempex-pp*). The application of *verb+ppron-subj+tempex-pp* to the example VIT in figure 7 (which is a (language independent) representation for "at 2 o'clock ?" as an elliptical question) results in the language specific (German) VIT given in figure 9.

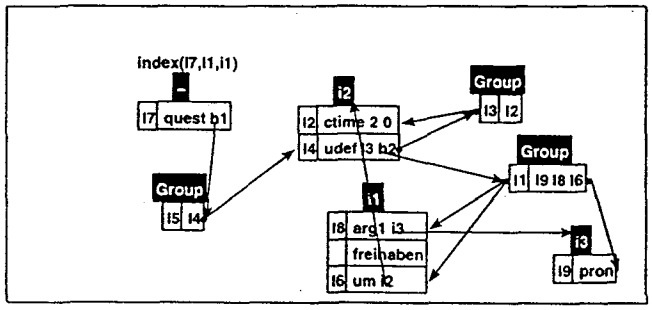

Figure 9: VIT-semantics of a protocol formulation

The semantic structure of the time expression remains unchanged. The under-specified condition *temp_loc* for discourse referent *i1* of the tempex–VIT has been exchanged by a condition for the German (temporally intended) preposition *um* (*at*). *I1* is extended furthermore by the main verb *freihaben* and a condition for its verb argument in subject position (*arg1*) which is realized by the personal pronoun (*pron*) in *i3*. The verbalization of this VIT will be "Haben Sie um 2 Uhr frei ?" (literally: "Are you at 2 o'clock free ?").

Dialog Nr. 3

- A: Guten Tag .
  (VM: Hi !)
- B: Hello. I would like to make a date with you.
  (VM: Wir müssen einen Termin ausmachen.)
- A: Was halten Sie vom siebzehnten ?
  (VM: How about beginning the seventeenth ?)
- B: Seventeenth is ok. How about in the morning, say, at ten o'clock ?
  (VM: Wie bitte ? Wir könnten uns dann treffen .)
- A: Wie wäre es morgens um zehn ?
  (VM: How about in the morning ten o'clock ?)
- B: That's fine with me.
  (VM: Das passt gut bei mir.)
- A: Gut. Ich trage unser Treffen für zehn Uhr ein.
  (VM: Okay ! That's fine with me.)
- B: Ok, see you then.
  (VM: Auf Wiedersehen !)
- A: Gut, bis dann.
  (VM: Okay !)

Figure 10: Example Dialogue with System Translations

[2]Currently, we are working on heuristics for the choice between multiple applicable patterns.

5.2 Syntactic Generation and Protocol Formatting

The VIT–semantics of the protocol formulation is then handed over to the syntactic generator VM–GECO [Becker et al. 1998] for verbalization. VM–GECO is a highly efficient multi-lingual generation component which consists of a language independent kernel syntactic generator and language specific declarative knowledge sources for syntactic and lexical choices. The last step of our protocol generation module is the formatting of the protocol into an easily understandable and readable format. As the protocol consists of global information about the dialogue itself as well as paraphrased turn segments we chose a protocol format which allows for clear distinction between these parts of the protocol. Furthermore it is important to assign the speaker's name (if it is known) to the protocol formulations of each turn. There are three different formatting devices. The most prominent one is the production of a HTML-format of the protocol. Additionally, LaTeX and ASCII versions are available.

Figure 10 shows an example dialogue and the respective system translations. Figure 11 shows the HTML–format of the protocol[3] of this dialogue.

VERBMOBIL VERLAUFSPROTOKOLL Nr. 3

Datum: 13.1.1998 , Uhrzeit: 16:34 Uhr

GESPRÄCHSVERLAUF:

A und B begrüßen sich .

- B: (INIT_DATE) Ich möchte mit Ihnen einen Termin ausmachen .
- A: (SUGGEST_SUPPORT_DATE) Ich schlage am siebzehnten vor .
- B: (CLARIFY_STATE, SUGGEST_SUPPORT_DATE) Schlagen sie am siebzehnten vor ? Am siebzehnten schlage ich am Morgen vor .
- A: (SUGGEST_SUPPORT_DATE) Am Morgen schlage ich um 10 Uhr vor .
- B: (ACCEPT_DATE) Das geht bei mir .
- A: (CONFIRM, ACCEPT_DATE) Einverstanden . Das geht bei mir .

A und B verabschieden sich .

Protokollgenerierung automatisch am 13.1.1998 16:34:52 h

Figure 11: HTML-format of the Protocol

Obviously, some of the user utterances are not correctly understood and translated by the system, which is reflected in the protocol. However, with respect to the available data the protocol is correct. All protocol formulations have been generated based on the tempex mechanism. The protocol consists of three major parts. First there is a title (VERBMOBIL VERLAUFSPROTOKOLL NR. 3 – VERBMOBIL PROGRESS PROTOCOL NO. 3), followed by general information about the dialogue: date (*Datum*) and time (*Uhrzeit*). The main content (GESPRÄCHSVERLAUF – PROGRESS OF THE DIALOGUE) are the individual turns which consist of the paraphrased segments. The individual dialogue acts of the segments are noted for debugging purposes.

6 Conclusions and Future work

We presented a novel module for progress protocol generation. Up to now at least 20 different protocols have been successfully generated. There are still open questions to be answered and further work is necessary to get a system able to generate more natural and flexible protocols. Future attention are and will be payed to: (*i*) Extending the current protocol generator to the generation of result and status protocols. Currently we are developing a language independent representation of propositional contents of turns and dialogues as a whole and also extending the generation component to produce coherent paragraphs. (*ii*) Indirect speech: By utilizing abstract information like dialogue act and propositional contents of the segments, we are free to switch to indirect speech. Rules and heuristics for when and how have to be developed. (*iii*) By coupling the thematic structure more tightly to the plan processor we hope to be able to utilize information like given–new and contrast, thereby generating more natural paraphrases, and determining when and how two segments carrying the same dialogue act and temporal information can be merged. (*iv*) The

[3]The text reads: A and B greet each other. A: I would like to make a date with you. B: How about the seventeenth ? A: Do you mean the seventeenth ? I suggest the seventeenth in the morning. B: I suggest in the morning at 10 o'clock. A: That's fine with me. A: OK That's fine with me. A and B say goodbye.

use of the transfer component to transfer the German protocol formulations into other languages. (*v*) Reduction: Better and more accurate rules for the reduction of clarification sub-dialogues have to be developed. (*vi*) Heuristics for the choice between multiple applicable VIT–patterns. Finally, (*vii*) Alignment: Rules and heuristics for the alignment of successive segments and turns.

References

[Alexandersson et al. 1997] Alexandersson, Jan, Bianka Buschbeck-Wolf, Tsutomu Fujinami, Elisabeth Maier, Norbert Reithinger, Birte Schmitz, and Melanie Siegel. 1997. Dialogue Acts in VERBMOBIL-2. Technical Report 204, DFKI Saarbrücken, Universität Stuttgart, Technische Universität Berlin, Universität des Saarlandes.

[Alexandersson and Reithinger 1997] Alexandersson, Jan and Norbert Reithinger. 1997. Learning dialogue structures from a corpus. In *Proceedings of EuroSpeech-97*, pages 2231–2235, Rhodes.

[Alexandersson, Reithinger, and Maier 1997] Alexandersson, Jan, Norbert Reithinger, and Elisabeth Maier. 1997. Insights into the Dialogue Processing of VERBMOBIL. In *Proceedings of the Fifth Conference on Applied Natural Language Processing, ANLP '97*, pages 33–40, Washington, DC.

[Becker et al. 1998] Becker, B. W. Finkler, A. Kilger, and P. Poller. 1998. An efficient kernel for multilingual generation in speech–to–speech dialogue translation. In *Submitted to COLING/ACL-98*, Montreal, Quebec, Canada.

[Block 1997] Block, Hans Ulrich. 1997. The language components in verbmobil. In *Proceedings of ICASSP-97*, pages 79–82, München.

[Bos et al. 1996] Bos, J., B. Gambäck, C. Lieske, Y. Mori, M. Pinkal, and K. Worm. 1996. Compositional semantics in verbmobil. Technical report, University of the Saarland, Computational Linguistics, Saarbrücken, July. Verbmobil Report 135.

[Bub and Schwinn 1996] Bub, Thomas and Johannes Schwinn. 1996. Verbmobil: The evolution of a complex large speech-to-speech translation system. In *Proceedings of ICSLP-96*, pages 2371–2374, Philadelphia, PA.

[Bub, Wahlster, and Waibel 1997] Bub, Thomas, Wolfgang Wahlster, and Alex Waibel. 1997. Verbmobil: The combination of deep and shallow processing for spontaneous speech translation. In *Proceedings of ICASSP-97*, pages 71–74, Munich.

[Bunt 1981] Bunt, Harry C. 1981. Rules for the Interpretation, Evaluation and Generation of Dialogue Acts. In *IPO Annual Progress Report 16*, pages 99–107, Tilburg University.

[Dorna 1996] Dorna, M. 1996. The ADT-Package for the VERBMOBIL Interface Term. Verb*mobil* Report 104, IMS, Universität Stuttgart, Germany.

[Dorna and Emele 1996] Dorna, M. and M. Emele. 1996. Efficient Implementation of a Semantic-Based Transfer Approach. In *Proceedings of ECAI-96*, pages 567–571, Budapest, Hungary, August.

[Jekat et al. 1995] Jekat, Susanne, Alexandra Klein, Elisabeth Maier, Ilona Maleck, Marion Mast, and J. Joachim Quantz. 1995. Dialogue Acts in VERBMOBIL. Verbmobil Report 65, Universität Hamburg, DFKI Saarbrücken, Universität Erlangen, TU Berlin.

[Kamp and Reyle 1993] Kamp, H. and U. Reyle. 1993. *From Discourse to Logic*, volume 42 of *Studies in Linguistics and Philosophy*. Kluwer Academic Publishers, Dordrecht.

[Küssner and Stede 1995] Küssner, Uwe and Manfred Stede. 1995. Zeitliche Ausdrücke: Repräsentation und Inferenz. Technical Report Verbmobil Memo 100, Technische Universität Berlin, December. In German.

[Wahlster 1993] Wahlster, Wolfgang. 1993. Verbmobil–Translation of Face-to-Face Dialogs. Technical report, German Research Center for Artificial Intelligence (DFKI). In Proceedings of MT Summit IV, Kobe, Japan, 1993.