# Max – A Multimodal Assistant in Virtual Reality Construction

Stefan Kopp, Bernhard Jung, Nadine Leßmann, and Ipke Wachsmuth

**In the Collaborative Research Center SFB 360 "Situated Artificial Communicators" the anthropomorphic agent Max is under development. Max inhabits a CAVE-like Virtual Reality environment where he is visualized in human size. Engaged in cooperative construction tasks, the human user meets a multimodal communication partner that employs synthetic speech, gaze, facial expression, and gesture. Combining deliberative and reactive conversational behaviors with the capability to initiate assembly actions, the agent guides the user through interactive construction procedures. This paper describes the overall architecture of Max with a focus on dialog and interaction management and the generation of synchronized multimodal utterances.**

## 1 Introduction

Virtual humanoid agents offer an exciting potential for interactive applications in Virtual Reality (VR). Cohabiting the environment with the human user, such agents may serve as communication partners that show the user around, present and explain the virtual world, or provide assistance in other kinds of education or training. A related vision is embodied conversational agents, i.e. agents that are able to engage in face-to-face conversations with the user, demonstrating many of the same communicative behaviors as humans do in such situations.

This article describes Max – the "Multimodal Assembly eXpert" – a VR-based conversational agent who can assist the user in virtual construction tasks. Max is being developed in the Collaborative Research Center SFB 360, which is concerned with the design of "Situated Artificial Communicators" that integrate multimodal conversational abilities for task-oriented dialogs in dynamic environments. In the basic scenario, a human and an artificial communicator engage in natural language interaction to cooperatively solve construction tasks with Baufix parts. Max is one realization of such an artificial communicator in a CAVE-like VR environment, where he is visualized in human size (see Fig. 1). In communicating with the user in a face-to-face manner, the agent employs prosodic speech, deictic and iconic gestures, eye gaze, and "emotional" facial expressions. The VR user - who is equipped with data gloves, optical position trackers, and a microphone – may employ natural language and gesture when interacting with Max. Furthermore, both Max and the user can initiate assembly actions in their environment.

The complete environment of Max builds on previous work in the Laboratory for Artificial Intelligence and Virtual Reality and the SFB 360 in Bielefeld. Our approach to the interpretation of multimodal, speech and gesture based input has been described in (Sowa, Kopp & Latoschik, 2001; Latoschik, 2002). The generation of human-like multimodal utterances for Max is explained in (Kopp & Wachsmuth, 2002). The interactive simulation of assembly operations, given a high-level specification of the task, is enabled by the Virtual Constructor (Jung et al., 2000). In a series of increasingly complex scenarios, this article explores how the virtual agent Max is able to assist the user in on-going assembly tasks (Section 2). Related work is discussed in Section 3. Then, Max's architectural framework is presented (Section 4) that closely couples conversational behavior with environment perception and manipulative actions. An important part of Max's abilities concerns the planning and management of the dialog with the user (Section 5). Finally, our approach to realizing synchronized speech and gesture based utterances is briefly described (Section 6).

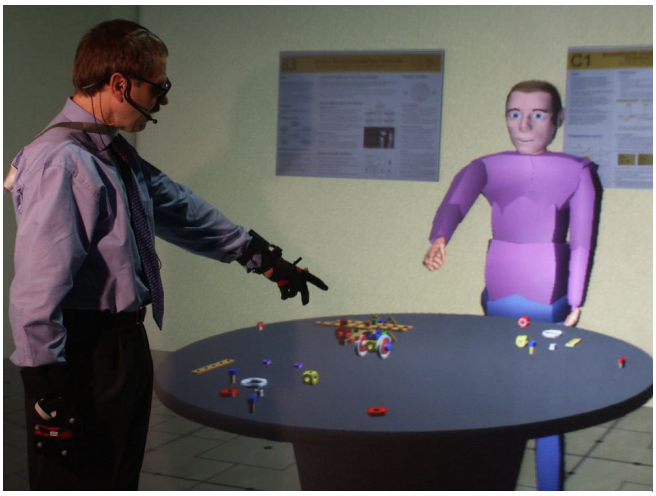


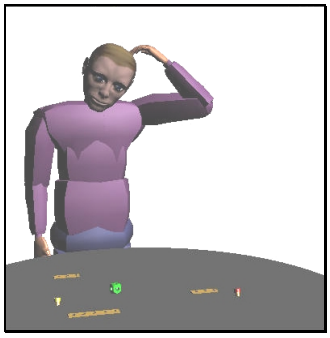
*Figure 1: In his CAVE-like Virtual Reality environment, Max guides the user through interactive construction procedures involving Baufix parts.*

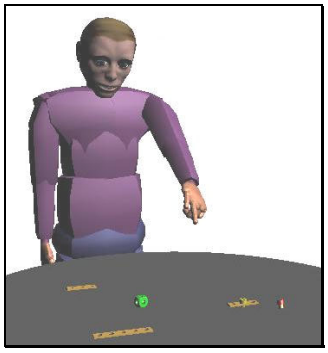
## 2 Example Scenarios

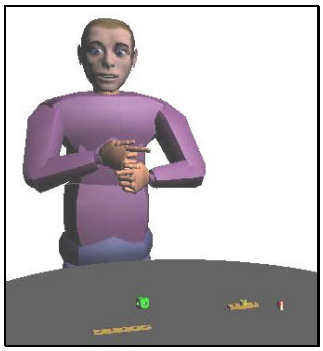
Max assists the user in VR-based construction procedures. The user and Max stand face-to-face across a table which has a number of Baufix parts lying on it. The user may rearrange the parts, add parts to the scene, or build something with the available parts. At any time, the user may ask Max for assistance with the construction of subassemblies of a Baufix airplane. All interactions of the user, both to trigger changes in the virtual environment and to communicate with Max, are specified using speech and gesture. In the following series of three scenarios of Max assisting the user in interactive construction procedures, the possible courses of actions are less and less predictable. Accordingly, they become increasingly demanding w.r.t. Max's capabilities concerning dialog, turn taking, and action planning.

*Scenario 1: Max explains and builds*

In the first scenario, Max both explains the construction and issues the commands for building Baufix assemblies using the currently available parts. For example, Max may be instructed *"Show me how to build a propeller"*. If a propeller







**User**: Explain the assembly of a propeller.

**Max**: Let's build it together. First, a screw needs to be inserted in the middle of a three-hole-bar.

[User takes the turn and proposes an instruction.]
**User**: Put the yellow screw into the bar.

[Environment: The yellow screw is inserted in the first hole of a bar.]

**Max**: No, that was the wrong hole.

[Environment: The wrong action is undone, i.e. the yellow screw moves out of the bar.]

**Max**: I explain it to you again. A screw needs to be inserted in the middle of a three-hole-bar.

**User**: Put the screw into the middle of the three-hole-bar.

[Environment: The assembly is performed.]

**Max**: Alright. Now, insert this screw into a three-hole-bar.
[During his explanation Max points at the yellow screw.]

**User**: You do it.

**Max**: Okay, I will initiate the action.

[Environment: The objects are assembled as instructed.]

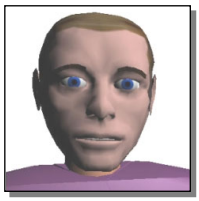
**Max**: And now the bars need to be turned crosswise.
[Max indicates the rotation using an iconic gesture.]

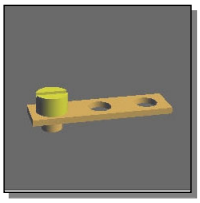
[A long pause occurs; after a while Max takes the turn.]

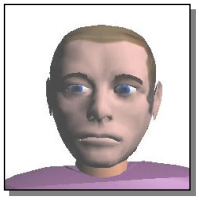
**Max**: Well, then I will do the construction-step.

[Environment: The objects are rotated as instructed.]

**Max**: Great. We have just completed the assembly of a propeller. Can I explain you something else?







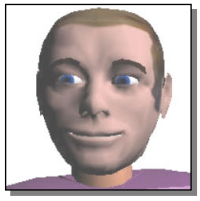


*Figure 2: Max assists the user in the assembly of a Baufix propeller (translated from German).*

can be built from the currently available parts, Max will explain the necessary assembly steps by producing verbal utterances like *"insert this screw into that bar"*. The verbal statements may be accompanied by gesturing, such as pointing or one- or two-handed iconic gestures, e.g., for depicting the desired relative orientation of two parts. Upon completion of a multimodal utterance, Max initiates the corresponding assembly action in the virtual environment. At any point during the demonstration, the user may interrupt Max (*"stop!"*) and later instruct him to continue. If an assembly group cannot be built from the currently available parts, Max will report failure to the user (*"A propeller cannot be built from the current parts"*).

*Scenario 2: Max explains, user builds*

Similar to Scenario 1, Max verbally and nonverbally explains the construction of Baufix assemblies, e.g. using instructions like *"Insert <pointing> this screw into the middle hole of <pointing> that three-hole-bar"*. Now, however, the user has to instruct the Virtual Constructor to assemble the parts. Clearly, putting the human in the loop makes the scenario more interactive but also introduces another source for failure, as the user may not perform the assembly step as instructed by Max. Therefore, Max observes the scene and implements a form of *action monitoring*. If the user's assembly fails to match the explained action Max informs the user about this failure. Max's explanation will also indicate the reason for the failure (e.g. *"No, that was the wrong hole"*). Then, Max undoes the wrong assembly step in the environment and repeats the last assembly instruction. Max also shows different "emotional" facial expressions as feedback depending on the success of the user's action.

*Scenario 3: Max explains, user or Max builds; generic explanations*

The third scenario combines features of Senario 1 and 2 in that either Max or the user may initiate the assembly steps explained by Max. The question who of the interlocutors is to perform the assembly action may now become subject of discourse. For example, the user may decide to have Max demonstrate the assembly (*"You do it!"*). Max, in the absence of user efforts to act, may also proactively decide to conduct the assembly.

As further increment of complexity, Max now explains the construction procedures in a more generic fashion: Instead of selecting individual parts to build an assembly (*"insert this screw into the middle of that bar"*), Max at first only refers to the types of the involved parts (*"insert a*

*screw into the middle of a three-hole-bar"*). Once the construction procedure has started, subsequent explanations need however refer to the individual parts used so far. Max's instructions may now contain a mixture of references to specific parts and generic descriptions (*"now insert the screw into the middle of a three-hole-bar"*). Fig. 2 shows an example of a construction procedure cooperatively performed by Max and the user.

# 3 Related work

Several virtual agents have been presented that are able to conduct multimodal dialog with a human user, e.g., Gandalf (Thorisson, 1997) who can answer questions about the solar system, or REA (Cassell et al., 2000) who provides house descriptions in the real-estate domain. These systems focus on the processing of multimodal input and output, i.e., how information is intelligibly conveyed using synchronized verbal and nonverbal modalities. Concerning the generation of multimodal utterances, this work culminated in the BEAT system (Cassell et al., 2001) that autonomously suggests and schedules verbal and nonverbal behaviors for a textually given utterance by exploring content and discourse structure. Yet, the employed techniques for realizing the devised behaviors suffer from limited flexibility when it comes to adjusting a gesture's timing to accompanying speech. This can be ascribed to the lack of sufficient means of modulating, e.g., shrinking or stretching, single gesture phases (cf. Cassell et al., 2001) and to a behavior execution that runs in an isolated animation system and cannot be monitored.

In the aforementioned systems, communication takes place in rather static scenarios, with the agent fulfilling the role of a presenter and the user only observing presented scenes. In contrast – and comparable to our assembly assistance scenarios – some educational applications allow a human student to perform the actions that are subject of the training process, while being monitored by a tutoring agent. Such agents thus need to combine communicative behavior with the ability to observe and to react to environmental changes. This poses greater demands on more general cognitive and perceptual capabilites. In the STEVE system (Rickel & Johnson, 1999), this has led to the usage of Soar (Laird, Newell & Rosenbloom, 1987), a general framework for modeling cognitive processes of intelligent agents. However, STEVE's architecture was limited as to that predefined templates were employed for processing user input as well as for realizing agent behaviors.

In recent work by Traum & Rickel (2002), the STEVE architecture was extended by a comprehensive dialog system that accounts for multimodal, multi-party, and multi-utterance conversations with open, unpredictable dialogs. Though only partially implemented, the model provides a broad foundation for dialogs in interactive virtual worlds in distinguishing between several layers each concerning a distinct aspect of the dialog's information state. Relating the model to our scenarios of assembly assistance (see Section 2), Max primarily needs to keep track of the dialog state w.r.t. *turn, initiative, topic,* and *obligation*. By initiative we consider the power to seize control of the dialog by presenting or confining a domain goal for the participants to achieve. Thus, sudden switches of initiative may occur, e.g., when the user asks for explanation of a new aggregate at a certain time in the discourse, but also when Max instructs the user to conduct a certain assembly action, possibly bringing up the same goal again. The resulting mixed-initiative dialogs are characterized by openness and unpredictability of the discourse. Besides switches of initiative, both Max and the user may take the turn or assign it to the interlocutor. The topic of the dialog is restricted to the assembly target and may be set or switched by the user at any time. Max is obliged to assist the user and hence must adopt the topic by, first, planning the requested assembly explanations and, second, demonstrating the construction procedure in a step-by-step manner, being sometimes committed to initiate actions himself when the user refuses or hesitates to do so. At any state, the forthcoming discourse is influenced by the situational context (e.g., the mutual consent on individual parts employed so far, the state of the ongoing assembly, or the outcome of a user action). To implement all of these dialog strategies in a unifying framework and to integrate them with perceptual and planning capabilities, we arrived at the architectural framework presented in the following section.

# 4 Architectural Framework

Max is controlled by a cognitively motivated agent architecture that is outlined in Fig. 3 and at first displays the classical perceive-reason-act triad. The direct connection between perceive and act illustrates the *reactive* component being responsible for reflexes and immediate responses. *Deliberation* processes take place in the reason section. Processing in the perception, reason, and act components runs concurrently such that reactive responses and deliberative actions can be simultaneously calculated.

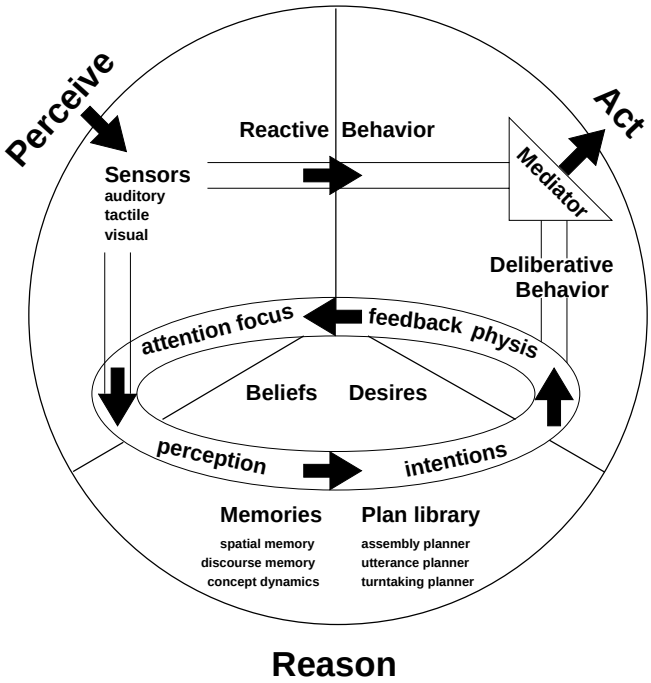



*Figure 3: Overview of the architectural framework of Max.*

**Perception**

The perception component consists of different sensory input channels, e.g., view and touch sensors that calculate the sighted or touched objects of the virtual scene, real

cameras tracking the user, the Virtual Constructor that recognizes construction relations between objects, and a speech recognizer. Perception is supposed not to be an inflexible acquisition of data but to be led by context-sensitive foci and, in addition, to directly trigger reactive responses. Inspired by the ACT-R (Anderson & Lebiere, 1998) approach how to connect perceptual with cognitive processing, acquired and interpreted percepts are encapsulated in dedicated buffers that write the most relevant information, the *beliefs*, in a central working memory. This either happens when new important data is recognized or when the focus of attention is shifted and the memories are told to reveal the currently relevant information. For example, when the Virtual Constructor recognizes a new aggregation of Baufix parts performed by the user, this information immediately becomes part of the beliefs.

**Deliberation**

The deliberative processes of Max are supposed to instantiate and manage the agent's dialog strategies during interaction with the user. Similar to the STEVE architecture, which is based on Soar, we chose to build these processes on top of a general model of rational reasoning in intelligent agents. However, we adopted the BDI architecture (Rao & Georgeff, 1991) for it provides better provisions for modeling intentional actions in form of plans, helping to perform complex tasks under certain context conditions while being interruptable and able to recover from failure. Soar agents, on the contrary, just associate individual actions with goals, which makes them very flexible on the one hand, but on the other hand necessitates extra knowledge to remain committed to a plan. In addition, we found the modularity of the BDI framework helpful to incorporate self-contained planners that can operate on appropriate knowledge representations.

The deliberative module consists of a BDI kernel partly based on JAM (Huber, 1999). The kernel operates on the *beliefs*, on *desires* representing the agent's persistent goals, and a plan-library to formulate adequate *intentions*. Desires can emerge from internal processing but also through interactions with the environment or the user. Persistent top-level goals are differentiated from instantiated subgoals. Intended courses of action are represented as plans with preconditions, context conditions, effect and a utility function. Such plans can either directly trigger specific behaviors to act but may also invoke dynamic, self-contained planners that, in turn, construct context-dependent plans employing specialized memories. That way, more complex plans can be hierarchically expanded on demand by the instantiation of lower-level plans. The BDI-interpreter continually pursues the plan with the highest utility value as an intention.

**Reactive Behaviors and Mediation**

Both the reactive and the deliberative component instantiate *behaviors* at the behavior layer of the architecture to perform their actions. The reactive module of the architecture, first, facilitates immediate responses to situation events or user actions. Currently, reactive behaviors of Max include gaze tracking and focussing the user in response to prompting signals (*"Max!"*). In addition, the module is responsible for incessant secondary behaviors to make the agent appear more lifelike, e.g., eye blink and breathe.

Behaviors instantiated at the behavior layer can activate themselves dynamically when certain preconditions have been satisfied and, then, compete for control over the required effectors (certain joints of Max's skeleton, muscles of his face model, or speech). A *mediator* allows the behaviors access to the effectors and resolves conflicts in favor of the behavior with the highest priority value. Such values may be altered by a behavior itself, possibly developing over time or according to a behavior's success.

# 5 Interaction Management

To guide the user through interactive assembly procedures, Max first has to plan the overall course of the construction task and, secondly, has to monitor its execution while allowing changes of initiative for single steps. These capabilities are realized in the deliberative component by refining hierarchical plans as situated responses to the environment and the user. The discourse between the user and Max thus evolves from the interplay of desires, plans, and beliefs, the latter reflecting perceived user actions as well as situation events. To this end, both environment manipulations as well as communicative acts are modeled as intentional actions, realized in a context-dependent manner by plans that fulfill subgoals at hand. Environmental manipulations are executed by sending massages to the Virtual Constructor simulation system. With respect to communicative acts, the currently available plans allow the user as well as Max to *inform*, *command*, or *ask* the interlocutor as well as to *refuse* or *confirm* a request or assembly action at stake. For example, if the user asks Max to interactively demonstrate the assembly of a certain aggregate, a desire is evoked in Max's deliberative component. This may lead to the instantiation of a general plan (e.g. constructTogether), which triggers a self-contained *Assembly-Planner* that employs expert knowledge about the assembly of complex aggregates to dynamically generate a suitable sequence of construction steps. Depending on the result, this plan will instantiate dedicated plans for realizing single communicative acts (e.g. refuse if the requested aggregate cannot be built).

## Assembly planning

Assembly plans are derived on-the-fly from assembly group representations in the frame-based representation language COAR (Jung, 1998), which also enable the Virtual Constructor to recognize the assemblies constructed by the user. To this end, the required parts in the COAR definition of an assembly group are matched against the currently available parts in the virtual environment. If the COAR definition requires two of these parts to be connected, a construction step specifying the connection of these two parts is inserted in the plan. If the COAR definition requires two parts to be oriented in a certain spatial relation, e.g. parallel, an action step achieving the rotation is inserted. In result, a sound construction plan for the requested aggregate is created. Note however that, depending on the current state of the virtual environment, it is very well possible that an aggregate whose assembly is requested by the user cannot be built from the currently available parts.

The superordinate constructTogether plan is expanded by the assembly plan, providing an abstract skeleton for the construction procedure. During the discourse Max tracks this procedure by subsequently instantiating a plan to explain the required actions and, then, asserting a sub-goal (constructOneStep) for realizing the particular step. The corresponding plan accomplishes the interactive fulfillment of a construction step by, first, managing the adoption of initiative by the user or Max and, second, performing appropriate actions such as initiating the necessary assembly step or evaluating the user's efforts and providing feedback. In accord with the state of the dialog as well as current beliefs, this will lead to the assertion of specific sub-goals (i.e. instantiated plans), notably assemble for performing assembly steps, explainActionStep or giveFeedback for informing the user, evaluateAction for evaluating the user actions, and commandActionStep for directing the user, respectively. As depicted in Fig. 4, the realization of such conversational actions is achieved by (1) forming an appropriate multimodal utterance and (2) synthesizing the required verbal as well as nonverbal behaviors (described in Section 5). The latter is initiated by the utter plan, which additionally employs a specialized plan (havingTurn) to detect whether Max is allowed to talk (turn-taking). Likewise, a data-driven "conclude" plan allows Max to deliberately react to the loss of the turn, e.g. when the user raises the hand. A possible configuration of the goal stack and current beliefs during an interaction with the user is shown in Fig. 5.

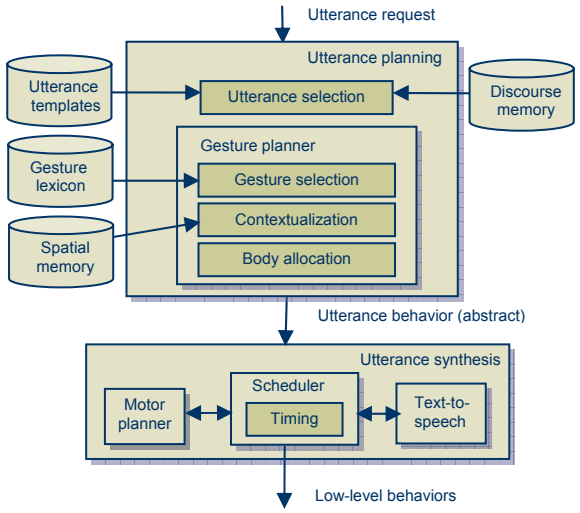



*Figure 4: Overview of the utterance production process.*

## Utterance planning

The utterance planner generates an utterance from the performative, the content (e.g., the assembly step to be explained), the involved objects, the discourse memory and the current situation. It builds on a database of utterance templates formulated in MURML, an XML-based representation language. Such descriptions (see Fig. 6 for an example) contain the verbal part, augmented with accompanying gestures and their cross-modal affiliation. Speech is defined already at the surface structure level, but may be parameterized w.r.t. single constituents. In contrast, gestures may be stated either explicitly in terms of form features of the meaningful phase or by specifying a desired communicative function a gesture is to fulfill.

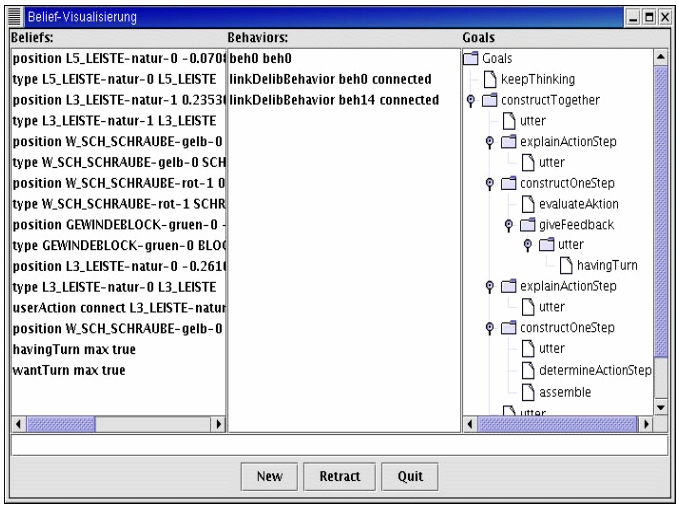



*Figure 5: Snapshot of current beliefs, active deliberative behaviors and goals during an interactive assembly procedure.*

By selecting an appropriate template and filling in verbal description parameters, the planner, on the one hand, is able to derive utterances which demonstrate a certain construction step to the user. This includes utterances in which a deictic gesture is designated to refer to an individual object (as in Fig. 6). In case an arbitrary object of only a certain type is involved, just the object category along with the specific connection-ports is verbally described. In addition, the required action may be illustrated by an accompanying iconic gesture for expressing a spatial relation. On the other hand, the utterance planner can provide feedback whether or not the user action was correct. In case of a failure, e.g., mismatching objects or connection-ports or if the wrong object was used, the utterance planner is able to verbalize the particular cause. Furthermore, if a construction plan has been successfully completed, feedback utterances for commenting on the outcome of the aggregations are generated.

Once the utterance planner has selected a template and prepared its verbal part as in Fig. 6, a gesture planner concretizes the gestures involved by specifying their expressive phases in terms of single movement constraints (Kopp & Wachsmuth, 2000). To this end, it chooses a gesture template from a lexicon that is appropriate for the desired communicative function, allocates body parts, expands movements constraints e.g. for symmetrical two-handed gestures, and resolves deictic references based on information from the spatial memory (see Fig. 4). As result, the MURML representation of the target utterance is fully qualified. Such specifications are the basis for an abstract deliberative *utterance behavior* that is further processed at the behavior layer where several low-level verbal and nonverbal behaviors are synthesized and executed.

In addition to speech and hand-gestures, Max can produce a variety of facial expressions and body postures to convey his "emotional" and attentional state (as outlined in Fig. 2). Simple secondary behaviors, e.g., looking around or head scratching, are automatically produced by the reactive component and pass the mediator when no deliberative behaviors with higher priority values, e.g. due to the user interacting with Max, are instantiated.

```
<definition>

 <!-- context parameters -->
 <parameter name="LocMovedObj"
value="(position_of schraube-1)"/>
 <parameter name="LocFixObj"
value="(position_of leiste-2)"/>

 <!-- utterance definition -->
 <utterance>
  <specification>
   Insert <time id="t1"/> this screw <time
   id="t2" chunkborder="true"/> in <time
   id="t3"/> this bar. <time id="t4"/>
  </specification>

  <!-- refer to screw by gesture -->
  <behaviorspec id="gesture_0">
   <gesture>
    <function name="refer_to_loc">
      <argument name="refloc"
       value="$LocMovedObj"/>
      <argument name="frame_of_reference"
       value="world"/>
    </function>
    <affiliate onset="t1" end="t2"
     focus="diese"/>
   </gesture>
  </behaviorspec>

  <!-- refer to bar by gesture -->
  <behaviorspec id="gesture_1">
    .. .. ..
  </behaviorspec>

 </utterance>
</definition>
```

*Figure 6: Sample specification of a multimodal utterance.*

# 6 Synthesis of Multimodal Utterances

The behavior layer of Max is in charge of generating coordinated verbal, gestural and facial behaviors for realizing an abstract utterance behavior. This includes the synthesis of intelligible verbal and gestural acts per se and, in addition, their combination into a continuous flow of human-like multimodal utterance. In particular, gestural and verbal elements, which concertedly express the rhematic part of the utterance, have to appear in temporal synchrony (cf. Cassell et al., 2000).

Max's speech synthesis module controls prosodic parameters like speech rate and intonation and delivers detailed timing information at the phonem level. In addition, it is able to prosodically focus single words by simulating realistic pitch accents (e.g., emphatic or contrastive stress).

In the gesture generation module, shown in Fig. 7 and described in detail in (Kopp & Wachsmuth, 2002), gesture animations are built and executed that accurately and reliably reproduce the spatio-temporal properties given in the MURML specification. A motor planner seeks a solution to control movements of the agent's upper limbs that satisfy the imposed constraints. Following a biologically motivated, functional-anatomical decomposition of motor control, the movements are driven by multiple kinematic controllers (local motor programs; LMPs) running concurrently and synchronized in a more abstract motor control program (MCP). Specialized motor control modules for the hands, the wrists, the arms, and the neck instantiate the local motor programs and arrange them in controller networks that lay down their potential interdependencies (de-/activation). At execution-time, the motor programs are able to de-/activate themselves as well as other controllers in the network. That way, suitable motion generators are applied, which control realistic submovements within a limited set of degrees of freedom and for a designated period of time.

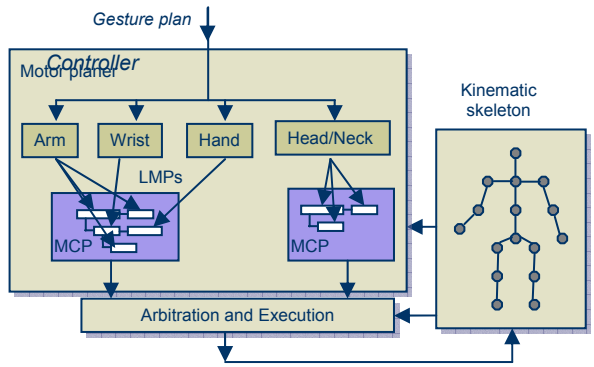



*Figure 7: Gesture generation at the behavior layer.*

To combine speech and gesture synthesis, utterances are produced incrementally under control of a global scheduler (see Fig. 4). Single chunks, each expressing a single idea unit by pairing an intonation phrase with a gesture phrase, are subsequently uttered but produced in an interleaved fashion. For each chunk, the classical two-phase, planning-execution procedure is extended by additional phases (*pending*, *lurking*, and *subsiding*) in which the production processes of subsequent chunks interact. That way, cross-modal coordination takes place on two levels of the utterance's structure: Within a chunk, the scheduler sets up timing constraints for the gesture to ensure cross-modal coherence. The stroke starts slightly before the affiliate and spans it. In case an affiliated word is narrowly focused in speech and thus carries the nucleus of the phrase, the stroke onset is set to coincide with the nucleus. At the inter-chunk level, gesture transitions result from self-activation of the LMPs and depend on the timing of the successive stroke. In consequence, fluent, situation-dependent gesture transitions emerge that may range from the adoption of intermediate rest positions to direct transitional movements. Likewise, the onset of the more ballistically executed intonation phrase is influenced by gesture timing (deferred in accord with the anticipated duration of gesture preparation). Using this incremental production model, Max synthesizes the multimodal utterance on-the-fly and generates them in a continuous fashion. Once all behaviors have been executed, feedback information is delivered to the deliberative module to signal the successful completion of the abstract utterance behavior.

# 7 Conclusions

We have presented the embodied conversational agent Max who serves as an assistant in Virtual Reality assembly tasks. Compared to the scenarios of other conversational agents, the virtual assembly environment provides both Max and the user with rich possibilities for autonomously manipulating the environment – affecting the subject of their ongoing

conversation. The dynamic nature of the environment and the resulting partial predictability of discourse with the user – combined with the overall need for knowledge, planning, communication, and acting – has led to a hybrid agent architecture. Max's deliberative processes are controlled by a BDI interpreter (JAM) which allows to treat communicative acts and world actions in a uniform way. It builds on a behavior layer that addresses an important, so far only insufficiently solved problem in the design of embodied conversational agents, namely the generation of tightly synchronized speech and graphics animation from formally specified utterance steps.

The architectural framework proved to be sufficient to implement the core functionality of an interactive assembly assistant in increasingly complex scenarios. We started out with a simple scenario in which Max acts as an expert and explains individual construction steps of an assembly. This was realizable by integrating an assembly planner relying on domain knowledge and a simple dialogue protocol through which subgoals were successively instantiated to explain one bit of information at a time. A hierarchical decomposition made the explanation process interruptible and controllable. In a next step, we integrated a simple perception component in the control flow which enabled Max to monitor whether the user performed the explained step in the requested way. In this scenario, Max appeared to be too restrictive in expecting the user to take the construction steps with exactly the referenced objects. To achieve a more cooperative scenario in which the user can actively contribute to the construction task, we extended Max's perception and interpretation abilities to enable him to abstract from individual objects and to explain sub-steps in a more general fashion. An utterance planner was established to generate context-dependent multimodal utterances taking into account the objects involved. A new plan was integrated to handle whether the user takes the initiative for single construction steps or Max should engage, either proactively or on demand, in the construction task. In summary, Max is able to pursue long-term plans for guiding a user through interactive construction procedures while the actual dialog emerges from the interaction with the user.

The setting of Max as face-to-face communication partner in an immersive VR installation provides a very attractive environment for the study and simulation of situated conversational behavior. As such it is an ideal testbed for research in SFB 360 concerned with the context-dependent communication in collaborative construction tasks. In future work, we plan to incorporate processing of subsymbolic information, supported by JAM in form of utility values. This will enable to model additional preferences as to which plan should be pursued in a given context, for example affected by an emotion module. Similarily, emotions must have an impact at the behavior generation layer where different emotions should lead to different facial display or prosodic properties of speech.

## Acknowledgments

This research is partially supported by the DFG in the Collaborative Research Center SFB 360.

# References

Anderson, J. R. & Lebiere, C. (1998). *Atomic Components of Thought*. Hillsdale, NJ: Lawrence Erlbaum.

Cassell, J., Bickmore, T., Campbell, L., Vilhjalmsson, H., & Yan, H. (2000). Human conversation as a system framework: Designing embodied conversational agents. In J. Cassell, J. Sullivan, S. Prevost, & E. Churchill (Eds.), *Embodied Conversational Agents*, pp. 29-63. Cambridge (MA): The MIT Press.

Cassell, J., Vilhjalmsson, H. & Bickmore, T. (2001). BEAT: the behavior expression animation toolkit. *Proceedings of SIGGRAPH 2001*, pp. 477-486.

Huber, M.J (1999). JAM: A BDI-theoretic mobile agent architecture. *Proceedings Third Int. Conference on Autonomous Agents*, pp. 236-243. Seattle, WA.

Jung, B. (1998). Reasoning about Objects, Assemblies, and Roles in On-Going Assembly Tasks. *Distributed Autonomous Robotic Systems* 3, pp. 257-266. Springer.

Jung, B., Kopp, S., Latoschik, M.E., Sowa, T. & Wachsmuth, I. (2000) Virtuelles Konstruieren mit Gestik und Sprache. *Künstliche Intelligenz* 2/2000, pp. 5-11.

Kopp, S., & Wachsmuth, I. (2000). A knowledge-based approach for lifelike gesture animation. In W. Horn (Ed.), *Proceedings of the 14th ECAI 2000*, pp. 661-667. Amsterdam: IOS Press.

Kopp, S., & Wachsmuth, I. (2002). Model-based animation of coverbal gesture. In *Proceedings of Computer Animation 2002*, pp. 252-257. Los Alamitos: IEEE Press.

Laird, J.E., Newell, A. & Rosenbloom, P.S. (1987). Soar: An architecture for general intelligence. *Artificial Intelligence* 33(1), pp. 1-64.

Latoschik, M.E. (2002). Designing Transition Networks for Multimodal VR-Interactions Using a Markup Language. In *Proceedings of the IEEE 4th Int. Conference on Multimodal Interfaces,* pp. 411-416.

Noma, T., Zhao, L. & Badler, N. (2000). Design of a virtual human presenter. *IEEE Computer Graphics and Applications* 20(4), pp. 79-85.

Rao, A. & Georgeff, M. (1991). Modeling rational behavior within a BDI-architecture. In *Proceedings Int. Conference on Principles of Knowledge Representation and Planning,* pp. 473-484.

Rickel, J. & Johnson, W.L. (1999). Animated Agents for Procedural Training in Virtual Reality: Perception, Cognition, and Motor Control. *Applied Artificial Intelligence* 13, pp. 343-383.

Sowa, T., Kopp, S. & Latoschik, M.E. (2001): A Communicative Mediator in a Virtual Environment: Processing of Multimodal Input and Output. *Int. Workshop on Information Presentation and Natural Multimodal Dialogue*, Verona, Italy.

Thorisson, K.R. (1997). Gandalf: An Embodied Humanoid Capable of Real-Time Multi-Modal Dialog with People. In *Proceedings First Int. Conference On Autonomous Agents*, pp. 536-537.

Traum, D. & Rickel, J. (2002): Embodied Agents for Multi-party Dialogue in Immersive Virtual Worlds. In: *Proceedings First Int. Joint Conference on Autonomous Agents and Multiagent systems*, pp. 766-773.