SAIBA

Some Previous Discussions

Reviewed by Hannes Högni Vilhjálmssson
hannes@ru.is
### MINDMAKERS Forum

#### Projects
- SAIBA Multimodal Behavior Generation Framework

#### Pages: [1]

<table>
<thead>
<tr>
<th>Subject</th>
<th>Started by</th>
<th>Replies</th>
<th>Views</th>
<th>Last post</th>
</tr>
</thead>
<tbody>
<tr>
<td>On the split and use of SAIBA, BML and FML</td>
<td>thorisson</td>
<td>0</td>
<td>132</td>
<td>August 23, 2006, 06:52:51 PM by thorisson</td>
</tr>
<tr>
<td>Digest of previous email discussions (part 7 of 7)</td>
<td>hannes</td>
<td>0</td>
<td>57</td>
<td>August 21, 2006, 05:53:02 PM by hannes</td>
</tr>
<tr>
<td>Digest of previous email discussions (part 6 of 7)</td>
<td>hannes</td>
<td>0</td>
<td>49</td>
<td>August 21, 2006, 05:51:41 PM by hannes</td>
</tr>
<tr>
<td>Digest of previous email discussions (part 5 of 7)</td>
<td>hannes</td>
<td>0</td>
<td>44</td>
<td>August 21, 2006, 05:49:04 PM by hannes</td>
</tr>
<tr>
<td>Digest of previous email discussions (part 4 of 7)</td>
<td>hannes</td>
<td>0</td>
<td>44</td>
<td>August 21, 2006, 05:48:05 PM by hannes</td>
</tr>
<tr>
<td>Digest of previous email discussions (part 3 of 7)</td>
<td>hannes</td>
<td>0</td>
<td>43</td>
<td>August 21, 2006, 05:37:12 PM by hannes</td>
</tr>
<tr>
<td>Digest of previous email discussions (part 2 of 7)</td>
<td>hannes</td>
<td>0</td>
<td>44</td>
<td>August 21, 2006, 05:32:21 PM by hannes</td>
</tr>
<tr>
<td>Digest of previous email discussions (part 1 of 7)</td>
<td>hannes</td>
<td>0</td>
<td>61</td>
<td>August 21, 2006, 05:09:14 PM by hannes</td>
</tr>
</tbody>
</table>
Topics

- Terminology
- Levels of Detail
- Real-time / Co-ordination
- Sync-Points
- Gaze
- FML
Terminology
The Terms BML and FML

- Behavior Markup Language (BML)
- Function Markup Language (FML)
- Previously existed internally at ISI to refer to the corresponding interfaces
- Terms had never been used externally until Reykjavik and therefore there should be little confusion with anything that existed before
- Top-level elements: `<bml>` and `<fml>`
What is “Communicative Behavior”? 

- Any perceived action is communication 
- Perhaps BML can ultimately encompass all such behavior – even non-communicative 
- But for now we are focusing more on face-to-face dialogue communication 
- We also want to stick to declarative definitions and not go too much into a procedural description of a script to execute in the world
Attribute Naming and Semantics

• As we start to extend BML elements with new attributes, we should try to use existing attribute naming conventions

• For example:
  – :amount (not :amt or :how_much)
  – :angle (not :ang or :degrees)

• The same for interpretations:
  – :angle is always in degrees
  – :duration is always in seconds
Levels of Detail
Varying levels of detail in BML

- BML is the language to formulate behaviors during high-level behavior planning or in a behavior repository.

- Lowest level of detail
  - a broad reference to an animation source.

- Highest level of detail
  - full-blown description of the behavior itself.
Lowest-Level of Detail

• Token behaviors
  – Nod, Shake, Wave, Smile
  – Problem: Lacks variety
  – Possibility: Up to the realizer / add attributes

• Animation names
  – Nod1, CH1A49
  – Problem: Animation source specific
  – Possibility: Also include token behavior
Highest-Level of Detail

• Is it BML “all the way”?
• But other languages already exist for some kinds of behavior
• Embedding those other languages vs. adding them to the BML specification
  – For example prosody (ToBI, SABLE)
• We seem to favour the embedded route for now to focus on core BML
The Gesticon and GRL!

- A whole chapter by itself...
Real-Time / Co-ordination
Real-time vs. Batch Considerations

• The connection between the behavior planner and the behavior realizer has to be two-way

• Ample feedback about behavior progress allows planner to react – supply new behaviors or re-plan
Reacting to World Events

• The generality of the `<wait> / <event>` mechanism
  – Co-ordinating between characters
  – Receiving events from the world
Special BML Scheduling Commands

• When we send a new `<bml>` block to the realizer, should the realizer assume it is to be cued after the previous block?

• Possibility to add attributes to the `<bml>` block that indicate:
  – Interrupt
  – Interrupt and continue
  – Integrate
  – Add to queue
Sync-Points
Sync-Points as a General Mechanism

• Should be able to express whatever time constraints/synchronization you require

• Par / Seq and other hierarchical approaches like BEAT have certain limitations

• How many sync-points are there in a behavior and what do we call them?
  – We want a core sync-point semantics
  – Also want to be able to cover all behaviors
Enforce Acyclic Timing Dependency

- Behaviors can only refer to previously declared sync-points
Unconstrained Time Values

• Should be “NULL” rather than a number such as “-1”

• Or whatever you get when you ask for the value of an attribute that doesn’t exist!
Gaze
What’s in a <gaze>?

- Can the <gaze> element control the entire body, including the legs?
- A really “smart” realizer will do whatever is needed to fulfil the request, even if it involves “stepping gaze”
- But we should be able to achieve the same effect in a “simpler” realizer by sending a <gaze>,<torso>,<legs>
FML
It is quite possible that another pass of FML to BML mapping will occur after the first pass

– For example when generating the text of an utterance
FML Is Pretty Open

• “An utterance can have a range of socio-emotive goals, there are a range of theories that characterize these goals and FML is currently being very open in what it includes” - Stacy