Google

Code

Archive

program-ab - ExtendingAIML.wiki

Export to GitHub

AIMLProcessorExtension

Program AB defines a Java Interface called AIMLProcessorExtension that you can use to define new AIML tags.

A class implementing AIMLProcessorExtension must provide:

- a Set of tag names.
- a function to recursively evaluate the XML parse tree for each node associated with a new tag.

The Program AB source includes a sample implementation of this interface called PCAIMLProcessorExtension, which defines a collection of tags simulating a contacts database.

Details

Provide the AIMLProcessor with a list of extension tag names by defining the fuction *extensionTagSet()*: `public Set<String> extensionTagSet();` which returns a set of extension tag names.

Define a function `public String recursEval(Node node, ParseState ps);` to recursively evaluate the XML parse tree. Because your custom tags may themselves contain ordinary AIML 2.0 tags, the evaluation function should use `AIMLProcessor.evalTagContent(node, parseState ignoreAttributes);` to evaluate any tag or subtag contents.

`public String recursEval(Node node, ParseState ps);` The function should return the result of evaluating the tag contents.

If you define an AIMLProcessorExtension class, such as PCAIMLProcessorExtension, then you need to set the static variable *extension* in AIMLProcessor:

```
AIMLProcessorExtension.extension = new PCAIMLProcessorExtension();
```