

ViSiCAST Milestone M5-10: Initial SiGML Definition

Project Number:	IST-1999-10500
Project Title:	ViSiCAST
	Virtual Signing: Capture, Animation, Storage and Transmission
Document Type:	Milestone Report

Milestone Number:	M5-10
Planned Date of Delivery:	June 2000
Actual Date of Delivery:	June 2000
Title of Milestone:	Initial SiGML Definition
Work-Package:	WP5 (Language and Notation)
Author(s):	JR Kennaway, JRW Glauert, R Elliott, and K Parsons (UEA)

Abstract:

This internal milestone marks the successful completion of the first stage of definition of the SiGML notation.

[By agreement within the project, "SiGML" is the new name for the notation originally designated "GML" in the project proposal.]

The definition consists of three (text) files, of which the second and third are auxiliary definitions, supplementing the first:

- (a) sigml_200000728.dtd;
- (b) gml_20000803.dtd;
- (c) hamgram_annot_20000803.edt.

The first of these (a) is the DTD (document type definition, or XML grammar) defining SiGML itself: this defines a language approximately equivalent in expressive power, and broadly similar in structure to, HamNoSys. It is a streamlined version of the DTD (b), which is essentially a reworking of the HamNoSys grammar in DTD notation. The HamNoSys grammar itself, annotated with informal semantic definitions, is given in (c). As yet there is no separate semantic definition for SiGML, but the semantics should be clear to anyone familiar with HamNoSys, and hence to anyone who follows the auxiliary definitions (b) and (c).

A prototype tool is also available for converting HamNoSys into SiGML.

ViSiCAST “HamNoSys-to-SiGML Translation”

R Elliott, JR Kennaway, K Parsons
SYS, UEA Norwich
{re,jrk,kjp}@sys.uea.ac.uk

September, 2000

1 Description of Work in Progress

We give here an outline of work currently in progress on translation from *HamNoSys* to *SiGML*. This is currently performed in two stages:

- translation from *HamNoSys* to “old” *SiGML* (sometimes referred to as plain *GML*);
- translation from “old” *SiGML* to “new” *SiGML* (the latter of which is sometimes referred to as “revised” *SiGML*).

1.1 From *HamNoSys* to “Old” *SiGML*

The first of the two stages is performed by a Java program, itself generated using Terence Parr’s *Antr* translator generation system (available from www.antlr.org). The relevant source files and build scripts are held in the main `hamToSiGML` folder. This folder also contains a text file, `hamgram_annot_20000803.{txt,edt}`, defining the input *HamNoSys* grammar annotated with informal definitions of the intended semantics. The Java bytecode for the translator is held in the `hpclasses.jar` file. The translator can be run using a command of the form:

```
genSigml hamnosys-input-file sigml-output-file
```

The result of applying the translator to the *HamNoSys* sign for “schmaltz” are shown in the *SiGML* output file `test.gml`. The file `test.out` shows the corresponding AST (abstract syntax tree), whose structure, it may be observed, is very close to that of the *SiGML* output — reflecting the (intentionally) very close similarity of structure between *HamNoSys* and “old” *SiGML*.

1.2 From “Old” *SiGML* to “New” *SiGML*

For the second translation stage, a different approach has been adopted: as both the input to, and the output from, this stage are XML, it has been implemented as an application of the XSLT transformation notation.

The relevant files are held in the `sigmlTransform` sub-folder of the main folder:

- The DTD’s for “old” and “new” *SiGML* are in the respective files `gml_20000728.dtd` and `sigml_20000728.dtd`.
- The definition of the XSL transformation between these two is held in `sigml.xsl`.
- `gml100July27.xml` is a simple “old” *SiGML* test file.
- `sigml100July27.xml` is the corresponding “new” *SiGML* file, generated using the given XSL definition (see below).
- Scripts to check the validity of these input and output files are held in the `check{Old,New}SigmlValid.vbs` (VBScript) files.

The XSL translation generating the “new” *SiGML* file corresponding to the given test file can be performed using James Clark’s *xt* translator (www.jclark.com), or using Microsoft’s *MSXML* library (msdn.microsoft.com/xml).

Again, testing to date has been very limited, due to the limited number of examples as yet available via the first stage.