MPML 3.0 Specification

Descamps Sylvain, Master Student
Internal Report
Ishizuka Laboratory
June 15, 2001

Motivation

This document contains the specification of MPML3.0 on the 15th of June 2001. Almost all what is written here is implemented already and if not, will be soon. Also any MPML script following this document should be able to be used with further version of MPML3.0. However some mistake may remain in it so don't hesitate to inform me if you find some. From now some new tags or functionality may be added but what is in this document will remain. A working version of the converter program will **SOON** be free of download from my computer (dynabook4) if you wish to test the system or even create your own presentation. I hope this will appear to be useful for whose working on MPML or interested in the project.

Tags definition explanation

EXAMPLE

Description of the tag.

Example:
Short example.

Attribute: Attibute_name1, [Attribute_name2], …, Attribute_nameN.
Attribute_name1: description.
Attribute_name2: description.
…
Attribute_nameN: description.
Limits between attribute if so.
[ ] means can be omitted.

Child tags: <Tag_name1>?, <Tag_name2>*, …, <Tag_nameM>?.
? means only 1 child of this type possible.
* means any number of children of this type possible.

Tags list

Tags description

MPML

This tag contains the MPML description in the file. It must be unique. Globally replaces the HTML tag.

Example:
<MPML>
…
</MPML>

Attribute: none.

Child tags: <HEAD>?, <BODY>?.

HEAD

This tag contains the HEAD zone, where the resources will be defined.

Example:
<HEAD>
…
</HEAD>

Attribute: none.

Child tags: <META>*, <TITLE>?, <AGENT>*, <SPOT>*.

BODY

This tag contains the BODY zone, where the presentation is described.

Example:
<BODY>
…
</BODY>

Attribute: none.

Child tags: <SEQ>?, <SEQ> children (see remark).

Remark: if the first tag is not a SEQ tag, then a default SEQ tag is introduced and then all the current children of the BODY tag becomes child of this new SEQ tag.

META

This tag is used as in XML specification.

Example:
<META id="meta1" name="author" description="author of the presentation" content="Sylvain Descamps" />

Attribute: id, [name], [description], content, [charset], [http-equiv], [scheme], [lang], [dir].

id: ID to use to refer to the meta.
name: name of the meta.
description: short explanation about the meta.
content: content of the meta (see XML specification for details about meta).
charset, http-equiv, scheme, lang, dir: see XML specification for details about meta.

Child tags: empty tag.

Remark: not useful at all in MPML3.0. Use as much as you want.

TITLE

This tag gives a title to the MPML presentation.

Example:
<TITLE>
    MPML definition …
<TITLE/>

Attribute: none.

Child tags: content.

SPOT

This tag defines a point on the screen. This point can be used later as spatial reference.

Example:
<SPOT id="spot1" name="middle" description="middle of the screen" x="400" y="300" />
<SPOT id="spot2" name="image" description="image of the machine" location="20,150" />

Attribute: id, [name], [description], x, y, location.
id: ID to use to refer to the spot.
name: name of the spot.
description: short explanation about the spot.
x: x coordinate of the spot.
y: y coordinate of the spot.
location: coordinate using "x,y" format.
Must choose between {x, y} or {location}.

Child tags: empty tag.

AGENT

This tag defines an agent. This agent can be used later with its ID. All agents have to be defined like this before use.

Example:
<AGENT id="agent1" name="merlin the sorcerer" description="A friendly character showing the pictures." system="MSAgent" character="merlin" spot="spot1" agreeableness="90" activity="50"/>
<AGENT id="agent2" name="very polite Japanese guy" description="The brother of merlin." system="DigiAgent" character="kojimasan" x="100" y="300" voice="{B674FFB0-A161-11D1-9019-006097B010E2}" activity="20"/>

<AGENT id="agent3" name="genie" description="Character appearing only to make bad comments." character="genie" location="20,434" agreeableness="10" />

Attribute: id, [name], [description], system, character, spot, location, x, y, [voice], [agreeableness], [activity].
id: ID to use to refer to the agent.
name: name of the agent.
description: short explanation about the agent.
system: agent system used. Can be "MSAgent" until now, but will be used for "DigiAgent" and "SmArt" new agent system supported.
spot: initial place of the agent. Defined by a spot.
location: initial coordinate using "x,y" format.
x: initial x coordinate of the agent.
y: initial y coordinate of the agent.
voice: voice to use for the TTS speech engine.
agreeableness: defines the friendliness of the agent's personality (Five Factors Model). Value between 0 and 100.
activity: defines the activity of the agent's personality (Five factors Model). Value between 0 and 100.
Must choose between {spot}, {location}, or {x, y}.

Child tags: empty tag.

PAGE

This tag defines a background of the presentation. Once the page is defined, the background of the presentation is known. The presentation needs a background before any action can be done.

Example:
<PAGE id="page1" name="results" description="Results of the study" ref="page1.html" >
…
</PAGE>

Attribute: [id], [name], [description], ref.
id: ID to use to refer to the page.
name: name of the page.
description: short explanation about the page.
ref: URL of the page to show as background.

Child tags: <SCENE>*, <SEQ>*, <PAR>*, <PAGE>*, <MOOD>*, actions*.

Remark: if a PAGE tag contains another PAGE tag, the background will change and then come back to the first page when the second PAGE tag is closed. Action needs to be contained directly or indirectly in at least one PAGE tag.

SCENE

This tag defines a scene. A scene is used to structure the presentation like in a theater, and decides which agents are visible.

Example:
<SCENE id="intro" name="presentation introduction" description="Merlin introduces what will be presented." agents="merlin, genie" >
…
</SCENE>

Attribute: [id], [name], [description], [agents].
id: ID to use to refer to the scene.
name: name of the scene.
description: short explanation about the scene.
agents: the agents which will be visible during the scene. If no agents are given, no agents are visible (can be used for configuration pages or others).

Child tags: <SCENE>*, <SEQ>*, <PAR>*, <PAGE>*, <MOOD>*, actions*.

SEQ

This tag defines a sequential set of actions. These actions will be executed one by one. Inspired from SMIL specification, however simplified.

Example:
<SEQ id="S1" name="merlin's arrival" description="merlin starts the presentation" agents="merlin">
…
</SEQ>

Attribute: [id], [name], [description], [agents].
id: ID to use to refer to the seq.
name: name of the seq.
description: short explanation about the seq.
agents: the agents will can be use during the sequence. Used mainly for checking the script. Be careful that you have to define all the agents that will be used in the children as well, even if they are not visible. If the *agents* attribute is absent, it means all the agents are used.

Child tags: <SCENE>*, <SEQ>*, <PAR>*, <PAGE>*, <MOOD>*, actions*.

PAR

This tag defines a parallel set of actions. These actions will be executed in the same time. See SMIL specification for further details.

Example:
<PAR id="S2" name="agent fight" description="merlin and genie argue" >
…
</PAR>

Attribute: [id], [name], [description].
id: ID to use to refer to the par.
name: name of the par.
description: short explanation about the par.

Child tags: <SCENE>*, <SEQ>*, <PAR>*, <MOOD>*, actions*.

Remark: a PAR tag cannot contain a PAGE tag as a child, even indirect.

MOOD

This tag defines the mood of the agents. With no mood tag, the agents have neutral mood. The MOOD tag use the emotion-like assign attribute. See EMOTION tag for details.

Example:
<MOOD id="Mood1" name="Nice beginning" description="Everyone is happy." assign="happy" >
…
<MOOD id="Mood2" name="Messy" description="Rocky arrived and made bad comment so Merlin's mood gets bad." assign="merlin:unhappy" >
…
</MOOD>
…
</MOOD>

Attribute: [id], [name], [description], assign.
id: ID to use to refer to the mood definition.
name: name of the mood definition.
description: short explanation about the mood definition.
assign: the definition string. See EMOTION tag for details.

Child tags: <SCENE>*, <SEQ>*, <PAR>*, <PAGE>*, <MOOD>*, actions*.

EMOTION

This tag defines the emotion of the agents. It makes the agent react to the occurring emotion and then getting back to normal again.

Example:
<EMOTION id="Emo1" name="Jealousy" description="Genie gets angry because all the others are very happy when the new friend arrives." assign="happy+,genie:angry" />

Attribute: [id], [name], [description], assign.
id: ID to use to refer to the emotion definition.
name: name of the emotion definition.
description: short explanation about the emotion definition.
assign: the definition string. The format is like: "[agent1:]emotion[,agent2:emotion2]…[,agentN:emotionN]". Agent1 receives the emotion after ':' which is a personal emotion. If an emotion is described without an agent for it, every agent without personal emotion will receive it.

Child tags: empty tag.

Remark: it is an *action* tag.

EXECUTE

Execute a JavaScript command on the background page. Can be used to synchronize the background events with the flow of the presentation.

Example:
<EXECUTE id="picture2" name="Picture change" description="Change the current picture to the next one." target="changepicture()" />

Attribute: [id], [name], [description], target.
id: ID to use to refer to the execute.
name: name of the execute.
description: short explanation about the execute.
target: the address of the function to execute.

Child tags: empty tag.

Remark: cannot use the character " in the function, like for parameters, yet. It is an *action* tag.

WAIT

Consult a JavaScript variable existing on the background page and wait until it is different from 0.

Example:
<WAIT id="wait2" name="Movie Synchronization" description="Wait until the JavaScript indicates the movie has terminated using the moviefinished variable" target="lowerframe.moviefinished" />

Attribute: [id], [name], [description], target.
id: ID to use to refer to the wait.
name: name of the wait.
description: short explanation about the wait.
target: the address of the variable to test for the synchronization.

Child tags: empty tag.

Remark: it is an *action* tag.

CONSULT

Test a variable on the background page and choose what to do depending of its value.

Example:
<CONSULT id="consult1" name="Response to user" description="Check the answer of the student and give the appropriate answer to it." target="choice" mode="pass">
    <TEST value="1" >
        …
    </TEST >
    <TEST value="2" >
        …
    </TEST >
</CONSULT>

Attribute: [id], [name], [description], target.
id: ID to use to refer to the consult.
name: name of the consult.
description: short explanation about the consult.
target: the address of the variable to test.
mode: indicates what to do in case nothing is recognized. Can be "pass" to indicate to go to the next tag if nothing matches the value of the variable or "redo" to ask to check the variable until a match is found.

Child tags: <TEST>*.

Remark: it needs top have TEST tag children to indicate the possible values and what to do in case a match is found. It is an *action* tag.

TEST

Test the variable of the parent CONSULT and contain the script to execute in case the corresponding match is found.

Example: See CONSULT tag.

Attribute: value.
value: the value the variable should match to execute what is in this TEST tag.

Child tags: <SEQ>?, <SEQ> children (see remark).

Remark: if the first tag is not a SEQ tag, then a default SEQ tag is introduced and then all the current children of the TEST tag becomes child of this new SEQ tag.

PAUSE

Make the indicated pause before going on.

Example:
<PAUSE id="pause2" name="Pause before asking" description="Wait 1s before asking to the student what he though of the presentation." pause="1000" />

Attribute: [id], [name], [description], pause.
id: ID to use to refer to the pause.
name: name of the pause.
description: short explanation about the pause.
pause: duration of the pause in milliseconds.

Child tags: empty tag.

Remark: it is an *action* tag.

MOVE

Make an agent move on the screen

Example:
<MOVE id="goleft1" name="Merlin jogging" description="Merlin moves to the left of the screen." agent="agent1" spot="left_of_screen" />

Attribute: [id], [name], [description], agent, spot, location, x, y.
id: ID to use to refer to the move.
name: name of the move.
description: short explanation about the move.
agent: ID of the agent moving.
spot: ID of the point where to go.
location: destination coordinate as "x, y" format.
x: destination x coordinate of the agent.
y: destination y coordinate of the agent.
Must choose between {spot}, {location}, or {x, y}.

Child tags: empty tag.

Remark: it is an *action* tag.

PLAY

Make an agent show gesture or play animation.

Example:
<PLAY id="hello1" name="Arrival" description="Genie smile to the user." agent="agent3" act="Pleased" />

Attribute: [id], [name], [description], agent, act.
id: ID to use to refer to the action.
name: name of the action.
description: short explanation about the action.
agent: ID of the agent acting.
act: name of the act (must conform to the agent/system acts).

Child tags: empty tag.

Remark: it is an *action* tag.

SPEAK

Make an agent speaks.

Example:
<SPEAK id="intro" name="introduction" description="introduction from robot." agent="agent4" >
    Welcome to my presentation.
    <NB/>
    My name is Robby.
</SPEAK>

Attribute: [id], [name], [description], agent.
id: ID to use to refer to the speech.
name: name of the speech.
description: short explanation about the speech.
agent: ID of the agent speaking.

Child tags: <EMOTION>*, <NB>*, <TXT>*.

Remark: it is an *action* tag.

THINK

Make an agent thinks.

Example:
<THINK id="thought1" name="perlin thinking" description="perlin makes fun of merlin." agent="agent2" >
    He is stupid.
    <NB/>
    He looks ugly.
</THINK>

Attribute: [id], [name], [description], agent.

id: ID to use to refer to the thought.
name: name of the thought.
description: short explanation about the thought.
agent: ID of the agent thinking.

Child tags: <EMOTION>*, <NB>*, <TXT>*.

Remark: it is an *action* tag.

NB

Divide the speech or thought into several bubbles (if supported by current agent) and can add pause in-between.

Example:
<SPEAK id="intro" name="introduction" description="introduction from robot." agent="agent4" >
    Welcome to my presentation.
    <NB/>
    My name is Robby.
</SPEAK>
<THINK id="thought1" name="perlin thinking" description="perlin makes fun of merlin." agent="agent2" >
    He is stupid.
    <NB pause="500" />
    He looks ugly.
    <NB pause="1000" />
</THINK>

Attribute: [pause].
pause: pause before the next sentence in millisecond.

Child tags: empty tag.

TXT

Used to allow a variable speech. It reads a JavaScript variable from the background and includes it in the speech.

Example:
<SPEAK id="intro" name="introduction" description="introduction from robot." agent="agent4" >
    Welcome to my presentation <TXT target="name" />!
    <NB/>
    My name is Robby.
</SPEAK>

Attribute: target.
target: the address of the variable to use in the background.

Child tags: empty tag.