Yves-Marie Omnès and Laurent Gabiot

# Improving communication amongst production and post-production teams by enhancing digital sound reports: an iXML extension proposition.

# Report from the IBC Meeting (september the 16th, 2014)

**The meeting**:

The 16th of September, 2014, on the Aaton Digital stand, during the IBC show at Amsterdam, was organized by the AFSI a meeting with manufacturers about our new propositions for iXML.

Were present Pascal Grillère from Aaton, and Paul Isaacs from Sound Devices. Aeta, Sonosax and UK Gallery were unable to attend, as well as Nagra and Zaxcom.
The following propositions were accepted:

# The Agreed Propositions

About this document:
- Tags are in bracket, as in <TAKE_TYPE>. New tags are in <RED>, while already existing ones are in <BLUE>.
- For optional dictionary of possible values, new values are in RED, old ones in BLUE. When no particular values are specified, we imply that a user defined string is to be used.
- Examples are in *<BLACK> ITALICS*.
The propositions are artificially grouped in three categories: project, take, and track specific informations.

**Project Specific Informations**

A set of fields about the current production:

<FULL_TITLE>
The name of the project. This might typically be the name of the motion picture or program without character limitation. It is not to be mistaken with the <PROJECT> field, which indicates usually a shorter, character limited version of the project name, usually chosen as the name of the folder containing the project related audio files.
*For instance: "On Dangerous Ground" from Nicholas Ray, would have the following fields:*
<PROJECT>DGROUND</PROJECT>
<FULL_TITLE>On Dangerous Ground</FULL_TITLE>


<DIRECTOR_NAME>
Obvious.
*For instance:*
<DIRECTOR_NAME>Nicholas Ray</DIRECTOR_NAME>


<PRODUCTION_NAME>
Idem.
*For instance:*
<PRODUCTION_NAME>RKO Radio Pictures</PRODUCTION_NAME>


<PRODUCTION_ADDRESS>
Obvious. Example obvious as well...


<PRODUCTION_MAIL>
Idem.


<PRODUCTION_PHONE>
Idem.


<PRODUCTION_NOTE>
A field to write some notes about the company that manages/produces the project to which this file belongs. This is a very generic field, that can be handy.
*For instance, it could be used to store contact of the lab person in charge of the rushes, or contact of the post production coordinator, etc... :*
<PRODUCTION_NOTE>Post Production Coordinator: John Smith (0)456 213 6746</PRODUCTION_NOTE>


A set of fields with informations regarding the sound mixer.

<SOUND_MIXER_NAME>
Obvious


<SOUND_MIXER_ADDRESS>
Idem


<SOUND_MIXER_MAIL>
Idem


<SOUND_MIXER_PHONE>
Idem


<SOUND_MIXER_NOTE>
A very generic field that can be used for many things.
*For instance:*
<SOUND_MIXER_NOTE>for the duration of the shooting in Italy, from 07/15 to 09/04, reach me at 0039 789 6545 instead of the usual number</SOUND_MIXER_NOTE>

Informations about the recorder itself.

<AUDIO_RECORDER_MODEL>
same information that the one stored in the <BWF_ORIGINATOR> tag contained in the <BEXT> object.

<AUDIO_RECORDER_SERIAL_NUMBER>
Obvious

<AUDIO_RECORDER_FIRMWARE>
Idem

**Take Specific Informations:**

<TAKE_TYPE>
TAKE(=default), PICKUP, WILD_SOUND, REHEARSAL, ANNOUNCEMENT, SOUND_GUIDE
Simply the kind of sound we are recording. By default it is a TAKE, but the user can choose a different definition from the included dictionary, or enter a free definition.
It should be noted that this new tag is redundant with an already existing tag <WILD_TRACK> which was new in V 1.52.


A new object with related tags regarding location:

<LOCATION>
This is a new object (i.e. a tag that will contain other tags) that will contain all location related informations. The included tags follow:


<LOCATION_NAME>
Shooting or script location description, as freely chosen by the user.
*For example:*
*<LOCATION>Norman Bates's house staircase</LOCATION>*


<LOCATION_GPS>
For recorders that are GPS equipped, the GPS coordinates of the place where the recording occurred. Encoding for coordinates needs to be chosen.
For example:
*<LOCATION_GPS>47.756787, -123.729977<LOCATION_GPS>*


<LOCATION_TYPE>
NONE(=default), INT, EXT, INT/EXT
The type of location: interior, exterior, a mix of the two, or the default value, NONE, if the user did not enter this data.


<LOCATION_TIME>
NONE(=default), SUNRISE, MORNING, MIDDAY, DAY, AFTERNOON, EVENING, SUNSET, NIGHT
Moment of the day of the current scene, as described by the script. NONE is default if not set by the user.


*Here is an example of a full* *<LOCATION>* *object:*

*<LOCATION>*

   *<LOCATION_NAME>Machine Room</LOCATION_NAME>*
   *<LOCATION_GPS>52.387485, 13.121908</LOCATION_GPS>*
   *<LOCATION_TYPE>INT</LOCATION_TYPE>*
   *<LOCATION_TIME>NIGHT</LOCATION_TIME>*

*</LOCATION>*


Miscellaneous tags:

<RATING>
NONE(=default), 0, 1, 2, 3, 4, 5
A rating given by the user to a take, like those found in MP3 players. A take could be a 5 stars take if it is a very good take. NONE is default if not set by the user. Note that NONE is not the same as 0 (a very bad take!).


<INPUT_SETTINGS>
A new object describing input settings of the recorder. Only phase and delay are supported as of now.
A set of generic tags is used to describe each kind of inputs. In the definition below, X stands for the number of the input.

<MIC_X> for microphone inputs, <LINE_X> for line inputs, <AES_X> for AES inputs, <AES42_X> for AES42 inputs, <MADI_X> for madi inputs, <WIRELESS_X> for integrated wireless inputs, <ETHERNET_X> for ethernet inputs (or <DANTE_X>, <RAVENNA_X>, etc...). Of course the list could grow in order to accommodate any hardware solution.

Then two further tags gives the settings of the corresponding input:

`<PHASE>`
`NORMAL, REVERSED`
if an input can be phase inverted, the last setting at the end of the recording will be stored.

`<DELAY>`
Delay should be expressed in samples.

*For example:*
```
<INPUT_SETTINGS>
                <MIC_1>
                    <PHASE>NORMAL</PHASE>
                    <DELAY>56</DELAY>
                </MIC_1>
                <MIC_2>
                    <PHASE>REVERSED</PHASE>
                    <DELAY>0</DELAY>
                </MIC_2>
                [...]
</INPUT_SETTINGS>
```

Slate identification tags improvements:

Currently a slate is recorded as a `<SYNC_POINT>` whose `<SYNC_POINT_FUNCTION>` is set as `SLATE_GENERIC`. In order to be able to specify the kind of slate being recorded, one way would be to add new values to the `<SYNC_POINT_FUNCTION>` dictionary. New values would be: `HEAD_SLATE,` `TAIL_SLATE, NO_SLATE` on top of `SLATE_GENERIC` to store slate informations.

The architecture is as follows:
```
<SYNC_POINT_LIST>
      <SYNC_POINT_COUNT>XX</SYNC_POINT_COUNT>
      <SYNC_POINT>
                <SYNC_POINT_TYPE>RELATIVE OR ABSOLUTE</SYNC_POINT_TYPE>
                <SYNC_POINT_FUNCTION>SEE SYNC POINT FUNCTION DICTIONARY
                </SYNC_POINT_FUNCTION>
                [...]
```
So in addition to the `SLATE_GENERIC` value, we would add the following values to the `<SYNC_POINT_FUNCTION>` dictionary: `HEAD_SLATE`, `TAIL_SLATE`, and `NO_SLATE`.
If the user wants to specify the kind of slate being recorded, he can do so here.

`<SYNC_POINT_ANNOUNCE>`
`WITH(=default), WITHOUT, ERROR`
This field would only be created if a sync point has a `<SYNC_POINT_FUNCTION>` tag whose value is either `SLATE_GENERIC`, `HEAD_SLATE`, or `TAIL_SLATE`.
Default value is `WITH`.
It would allow to inform that the slate announce is missing, or that it is somehow incorrect, as it sometimes happens.
*For example, a tail slate with a missing announce:*
```
<SYNC_POINT_LIST>
      <SYNC_POINT_COUNT>4</SYNC_POINT_COUNT>
      <SYNC_POINT>
                <SYNC_POINT_TYPE>ABSOLUTE</SYNC_POINT_TYPE>
                <SYNC_POINT_FUNCTION>TAIL_SLATE</SYNC_POINT_FUNCTION>
                <SYNC_POINT_ANNOUNCE>WITHOUT</SYNC_POINT_ANNOUNCE>
                [...]

      </SYNC_POINT>

      [...]

</SYNC_POINT_LIST>
```

**Track Specific Informations**

For track specific informations, we proposed that possibly all tracks informations (such as `<NAME>`, `<CHANNEL_INDEX>`, etc...) could be stored in each file that makes a take. This proposition has first to go to through a thorough evaluation.

It would imply the creation of a new object `<OTHER_TRACKS_LIST>` which will contain informations for the tracks not found in the current audio file. The `<OTHER_TRACKS_LIST>` object will have the same architecture than the current `<TRACK_LIST>` object, namely a `<TRACK_COUNT>` tag, and `<TRACK>` objects.

*An example of a 3 tracks take made out of monophonic files, recorded on an 8 tracks recorder (which implies 5 un-recorded tracks), could give for the first file:*

```
<TRACK_LIST>
        <TRACK_COUNT>1</TRACK_COUNT>
        <TRACK>
                <NAME>BOOM</NAME>
                <CHANNEL_INDEX>1</CHANNEL_INDEX>
                <INTERLEAVE_INDEX>1</INTERLEAVE_INDEX>
                [...]
        </TRACK>
</TRACK_LIST>
<OTHER_TRACKS_LIST>
        <TRACK_COUNT>2</TRACK_COUNT>
        <TRACK>
                <NAME>PAUL</NAME>
                <CHANNEL_INDEX>5</CHANNEL_INDEX>
                [...]
        </TRACK>

        <TRACK>
                <NAME>PASCAL</NAME>
                <CHANNEL_INDEX>6</CHANNEL_INDEX>
                [...]
        </TRACK>

</OTHER_TRACKS_LIST>
```

*The separation in different objects (`<TRACK_LIST>` and `<OTHER_TRACKS_LIST>`) enables a full compatibility with previous iXML versions: the `<TRACK_LIST>` logic stays untouched.*

Beside that change, new tags or new values are also added to the `<TRACK>` object (as found in the `<TRACK_LIST>` or `<OTHER_TRACKS_LIST>` objects):

`<TRACK_SOURCE_LIST>`
A new object that contains two new tags: `<TRACK_SOURCE_COUNT>` and `<TRACK_SOURCE>`.
Describes which inputs are assigned to the track.
`<TRACK_SOURCE_COUNT>` is the number of inputs feeding the track, and each subsequent `<TRACK_SOURCE>` indicates which input was used.
The input names should be following the names of the `<INPUT_SETTING>` object sub tags (see above).
*For example, a track comprising the boom and a hidden microphone that record that small sentence when the actor is shortly out of boom reach:*

```
<TRACK_LIST>
        <TRACK_COUNT>1</TRACK_COUNT>
        <TRACK>
                <NAME>BOOM</NAME>
                <CHANNEL_INDEX>1</CHANNEL_INDEX>
                <INTERLEAVE_INDEX>1</INTERLEAVE_INDEX>
                <TRACK_SOURCE_LIST>
                        <TRACK_SOURCE_COUNT>2</TRACK_SOURCE_COUNT>
                        <TRACK_SOURCE>MIC_1</TRACK_SOURCE>
                        <TRACK_SOURCE>MIC_2</TRACK_SOURCE>
                </TRACK_SOURCE_LIST>
                [...]
        </TRACK>
</TRACK_LIST>
```

`<TRACK_FUNCTION>`
new values added : `1_GENERIC, 2_GENERIC, etc..., N_GENERIC`
so that we can describe complex ensembles of microphones recordings.
*For example:*
*A classic opera recording would use a ramp of 5 microphones on the stage, labelled `1_GENERIC` to `5_GENERIC` from stage left to stage right.*


`<TRACK_PROPERTY>`
`NONE(=default), WILD_SOUND, PLAYBACK, SOUND_GUIDE, NO_SOUND`
Helps to further describe the significance of the track. The default value is `NONE`.
But a track could be a `WILD_SOUND` (a common occurrence, where a stereo ambiance recording is done on a pair of track while dialog is recorded on others.
It could be also a `PLAYBACK`, a `SOUND_GUIDE`, or even be tagged as `NO_SOUND` if the recorder has identified that no sound was indeed printed on that track.