

# Notes from MOS Protocol Group Meeting

16 Jun 2020

## 1. Element Status Codes

Andre Edmonson (Ross Video) provided a recap of his proposal.

Motivation: Critical to newsroom user to see status.

The current limited options of "NEW" "UPDATED" "MOVED" "BUSY " "DELETED", "NCS CTRL", "MANUAL CTRL", "READY", "NOT READY", "PLAY," "STOP" do not provide the kind of information, nor the flexibility required to convey meaningful status for the broadening range of media servers in use today and in the future. Categories of devices that were not even in use at the time the current standard was set included MAM and Automation systems.

Ex: Not ready – Too vague of a description.

Could be an unfulfilled placeholder, media transcoding in progress, or other.

Andre had previously posted earlier versions of his proposal in the MOS Slack channel.

There would be three elements to a Status level:

Standard Meaning

End User Meaning

Troubleshooting details

Each status level would have a numeric value associated with it. These would be similar to HTTP status codes (200-range: Success, 400-range: Error, etc.)

At previous meetings, there had been discussion of whether there should be Status levels with Editorial meaning. The version Andre presented at this meeting included an "Approved" status – representing an Editorial approval.

The group decided at this meeting to remove all Editorial status levels from the codified list of Reserved status levels – leaving intact only Technical status levels. Though a media system vendor may freely add other values as suits its needs – with the provision that they sit within the appropriate numbering range.

The proposal was approved unanimously by those casting a vote.

The full adopted version of the proposal is at: <http://mosprotocol.com/wp-content/MOS-Protocol-Documents/MOS-Item-Air-Status-Level--Adopted-20200616.pdf>

The new status classifications will be included in the next numbered versions of the MOS Protocol.

## 2. Profile 7

A proposal was made last year by Kai-Uwe Kaup (SciSys/CGI) regarding MOS Profile 7.

Broadly, the proposal was to make the nature of Profile 7 communication equally applicable whether initiated by the MOS or the NRCS.

As Kai not was not able to attend this meeting, discussion was limited, so Kai may better flesh-out his proposal and ideas. See Kai's notes at <http://mosprotocol.com/wp-content/MOS-Protocol-Documents/MOS-Profile-7-Parity-Proposal-20191210.pdf>

Commenting on Profile 7, in general, Shawn Snider (Ross Video) noted that an issue for his company has been Locking behavior.

This proposal will be brought up for further discussion at our next meeting.

## 3. Defer and Commit (MOS message threading)

Johan Nyman (SuperFly) presented a proposal in late 2019 for threading and delimiting strings of related MOS messages. See the proposal at <http://mosprotocol.com/wp-content/MOS-Protocol-Documents/MOS-Defer-and-Commit-proposal-20191101.pdf>

The goal of the proposal is to help a MOS build and organize a collection of messages. Prime example: an roList (or roCreate) message with all of the roStorySend messages that will come after it.

There were several issues raised.

What to do if Commit is not received? How are the earlier ones handled?

What happens if a story is updated (and the roStorySend message sent) before the whole collection of threaded roStorySend messages is sent?

Comments:

- Eira Monstad (Vizrt): This is not an urgent problem for Vizrt
- Mike Paquin (Ross Video): This is an edge case. But when it happens "it's not fun."

Johan will think on it some more, and discuss the proposal again in the future

#### **4. Obj paths**

The proposal was previously made to broaden the allowed formulation of an Object Path (or an ObjectProxy Path) in a MOS Object or MIR.

The current requirement is that the path statement point directly to the media (or other) file, and that the path URL end with the name of the object – including an appropriate media-type file extension.

While pointing to a specific file name at a specific location worked consistently well for traditional data storage repositories (e.g.: a specific server directory accessed via an HTTP daemon), the proposal would facilitate access/retrieval of files from non-traditional hosts (e.g.: MAMs or Databases).

One question raised was if there would be (somewhere in the new format) the name of the expected file and the media-type/file-extension that could be passed on to rendering or publishing systems that need that information.

Also, to support security around more complex content repository devices, if there should be authentication around the call. For media servers MOS4, this could be available as part of that standard.

A subgroup will be formed to follow up on the proposal.

#### **5. Obj Dur – And other Object vs. Item issues**

Shawn Snider says the Protocol provides no guidance regarding Item duration, as it does for Objects (objTB coupled with objDur).

Milan recalled that it had previously been part of the Protocol, but disappeared a few years ago.

Hence, objDur and objTB have been missing from Item messages.

Related – Concepts of Objects vs. Items. Items were originally designed to be references to Objects. But in practice, that is not how they have consistently been used.

Also, a question about objAir in the Protocol documents. Some use it in lieu of a media Status field.

A subgroup will be formed to follow up on the subject.

## 6. Float

Improvements to Float messaging and workflow was approved previously by the MOS Protocol Group.

Brief discussion about who is using it and how it is going.

Ross: Hasn't implemented yet because of versioning identification

Would plan to implement quickly after it is formally published as part of a MOS Protocol version.

SuperFly: Wants to use soon

Octopus: Octopus and Aveco implemented a new Float messaging and workflow with Aveco prior to Milan Varga presenting the proposal to make it a formal part of the MOS protocol. Their implementation is not exactly according to the version that was ultimately approved by the Group.

## 7. IABM

Phil Avner (AP/ENPS) presented a summary of newsroom technology resources being made available by IABM – of which most MOS Protocol Group members are also members.

Members may access the IABM website at <https://theiabm.org>

## 8. MOS Message Requires a Reply

ENPS reported recent problems where MOS systems are not always replying to messages from NRCs if the MOS believes the message to be a repeat of an earlier message or otherwise redundant.

This has led to backups in MOS message queues.

The reminder was made that, as a “guaranteed delivery” protocol, every message requires an appropriate reply – even if it is only a NACK.

## 9. Supporting redundant MOSes

ENPS relayed a customer request to have a standard mode for an NRCS to work with redundant or backup nodes of a MOS.

Shawn Snider says that part of the reason for MOS4 is its ability to be routed through a load balancer – allowing two or more redundant nodes to be equally available without the opposite end of the conversation caring which node is handling a request.

Discussion may be picked up in a breakout group later this year.

## 10. MOS 4

MOS4 was approved previously by the MOS Protocol Group

A brief discussion of what members are using it, and how it is working for them so far.

Ross Video has been implementing it in some products – StreamLine and Inception. Especially for use when there is a cloud-terrestrial hybrid environment. More products will be upgraded to incorporate a MOS4 option in the not-too-distant future.

CueScript has implemented it to work with Ross. Implementation was not difficult. Response time was reported to be faster than traditional socket connections.

### Next Meeting Dates

The next full MOS Protocol Group will be in October (exact date TBD)

Dates for meetings of Subgroups mentioned above, as well as a sub-group to discuss versioning (what gets included in the next published versions of the MOS Protocol and how it is designated) will be set in the coming weeks.

Written discussion will be in the MOS Slack space.