2011-04-19 Meeting Minutes

Tuesday, April 19, 2011

10:00 PM

**Attendees:** Sakae OKUBO, Brody Kenrick, Parthasarathi R, Simon Horne, Paul Jones

***AMS-0027 - An update of the AMS information flow [Waseda]***

This document was a follow-up to a similar document presented before (AMS-0027) and subsequent document presented during the most recent SG16 meeting.

We stepped through the call flow in Figure 1. During the SG16 meeting, we discussed a desire to have knowledge as to whether an application is available before initiating the session. This call flow introduces a Status Info message to meet that need versus having a pre-invoke type of message. At step 13, the flow shows an attempt to ensure that media is established properly before alerting the user (akin to H.460.11).

A question was asked as to whether Assemblage A would like to know the capabilities of Assemblage B before attempting communication on a call by call basis. This might be accomplished peer-to-peer via something like SIP OPTIONS or perhaps querying the service node or registrar. This need not be a definitive representation of what services could be invoked with certainty. This information would be useful in informing users of new applications as well as being useful in determining the likelihood of a successful call.

It was noted that there may be some security issues if applications are invoked that might be shared by multiple Containers. Imagine, for example, a video display that is registered with 10 Containers. If a call is placed to any one of those 10 Containers, then perhaps the video application might get invoked automatically.

Related to security concerns, one may not want certain applications to be automatically invoked (e.g., video). Perhaps a way to handle that is to allow the Container to accept the invocation requests, but reply that application 1 was automatically invoked, but applications 2 and 3 require user intervention. Once the call is established, Container B would prompt the user to accept applications 2 and 3. That would then result in messages from Assemblage B to Assemblage A to invoke those applications. These kinds of access control mechanisms need further study.

We did not get to the application handover call flow, but will cover that during the next meeting.