CLUE protocol

draft-presta-clue-protocol-02

CLUE design team meeting

15/10/2013
Outline

• Clue message types
• Extensions management
  – Rationale
  – OPTIONS mechanism proposal
Message types

• Each CLUE message inherits the characteristics of the corresponding class
• Three families
  1. Requests
     – Issued from a MC to a MP
     – Each request is answered by a response message
       • They are coupled by looking at the sequence number
     – Example: CONF, RE-ADV
  2. Responses
     – Answers to request messages, from the MP to the MC
  3. Notifications (*new*)
     – Sent asynchronously from the MP to the MC to notify offers and changes on the provider’s side
     – Example: the ADV sent to notify the current available telepresence capabilities
Why notifications

• The advertisement is not semantically a request
• The configure is not semantically a response
  – There can be more than one configuration request referred to the same advertisement
• ACKs/NACKs are a discussion apart
  – Used to guarantee that the message has been received
  – Can be piggy-backed on request-response pairs
  – Not considered as of yet

15/10/2013
Demux rule

• A CLUE endpoint can act as a MP and as a MC simultaneously, on the same channel
• Messages can be demultiplexed on the basis of their type
  – The MC part receives only
    • Responses
    • Notifications
  – The MP part receives only
    • Requests

15/10/2013
Extensions

• Something that is not envisioned in the current specification of the protocol...
• ...*And* something that is not envisioned in the current specification of the datamodel
  – since data model elements are included in CLUE messages
• Extensions are defined elsewhere
  – In other documents, in other XML schemas
  – An extension can be identified by the defining XML schema
Extension examples

• Extensions can be

  1. New data model elements
     • For example, a new audio capture attribute that can be used to provide an enhanced description
  2. New protocol message fields
     • For example, a new field in the request message identifying the sender of the message
  3. New protocol messages
     • For example, a new notification message

• New information (1 and 2) can be passed in place of the “any” and “any attribute” fields of the existing schema
• New messages (3) can be obtained deriving the CLUE message types
The OPTIONS proposal

• OPTIONS request
  – Issued by a MC to the MP as soon as the channel is ready
  – Contains
    • the data model extensions supported by the MC
    • The protocol extensions supported by the MC
    • The version number of the CLUE protocol supported by the MC

• OPTIONS response
  – Provides
    • The extensions supported by the MP among those proposed by the MC
    • The version number of the CLUE protocol supported by the MP
      – Lower than or equal to the one proposed by the MC
OPTIONS example

OPTIONS 3.0
  dm-ext: s1, s2, s3
  pr-ext: s4, s5

RESPONSE 1.0
  dm-ext: s1
  pr-ext: -

ADV 1.0

CONFIGURE 1.0

15/10/2013
Open issues

• Introduction of a notification acknowledge mechanism
  – E.g. to acknowledge an ADV

• How to manage consumer-only and provider-only couples
  – i.e., when the two parties are simultaneously consumer-only or provider-only

• Delta mechanism in the ADV

15/10/2013
How to manage consumer-only and provider-only couples (1/2)

• Current solution
  – Managed with timeouts
  – Provider-only case
    • Both MPs wait for an OPTIONS request that does not arrive
  – Consumer-only case
    • Both MCs issue an OPTIONS request but the responses do not arrive

15/10/2013
How to manage consumer-only and provider-only couples (2/2)

• Solution 2
  – Forcing each CLUE endpoint to implement both the MC and MP state machines only for the OPTIONS mechanism
  – Conceive option fields to say if the endpoint can be a provider and/or a consumer

• The MC side of the endpoint issues an OPTIONS request

• From the response, it is able to understand what are the roles performed by the other endpoint