

SOUND COMMUNICATION:
A STANDARD SYNTAX FOR
INTER-APPLICATION,
INTER-DEVICE AND
INTER-PLAYER
COMMUNICATION OVER OSC

Scott Hewitt

and

Pierre Alexandre Tremblay

The Standard Features

- Based on reserved (OSC style) name spaces
- Reserved name spaces in any order
- Optional Auto Setup
- Optional self documenting / propagating

Proposed First Layer Address Spaces

- /test
- /setup
- /chat
- /app
- /user
- /time
- /documentation
- /hardware

Name Space Conventions

Name spaces should provide as much context as possible to information carried

- /hardware – namespace descriptive of source
- other namespaces – names descriptive of target

Reserved Name Spaces

- /test

message 'check + ip + time'

response 'auto + ip + time'

Features

- Test structure built into protocol
- Basic realtime network/application profile

Reserved Name Spaces

- /setup

reserved message set x /y

'x' corresponds to the position within the hierarchy
'y' the name for detection

eg. set 1 /setup set 2 /pan

Features

- Dynamic configuration

- **Auto upgrade**

Reserved Name Spaces

- /chat

/chat/<name> message content

eg. /chat/james hello

Features

- Communication standard
- Simple implementation
- Publicly accessible stream

Reserved Name Spaces

- /app

/app/<program>/<unique id>

eg. /app/sc3/hjghju45/

Features

- Stable network aware/transparent API
- Separates application streams
- Allows interaction of applications without awareness

Reserved Name Spaces

- /user

/user/<identifier>

Features

- Flexible area layout
- Accountability for data
- Other name-spaces can be encapsulated within

Reserved Name Spaces

- /time

/time/sec <value>

Features

- Referential time
- /sec /ms /min /hour

Reserved Name Spaces

- /documentation

Messages

- all-messages – list commands in namespace
- help – delivers help message
- manual – delivers url of external manual
- url – delivers home page of app

Reserved Name Spaces

- /hardware

/hardware/<device>/<id>

eg. /hardware/SY99/hjg4g

Features

- Permits connection of device
- Shared or Local
- Utility network future

Why use it?

- Encourages Server / Client topography
- Fully network aware
- Multi-platform
- Easy to implement
- Expandable
- Human Readable
- Obvious Data Structure