

# OSCQS - Schema for OpenSound Control Query System version 0.0.1

Martin Habets (mhabets@users.sourceforge.net)

August 10, 2005

## Abstract

OSCQS is a simple Query System and Service Discovery protocol using OpenSound Control <sup>1</sup> messages and some URL scheme aspects from RFC 2782 <sup>2</sup>. This document holds the OSC schema for OSCQS version 0.0.1. For more information refer to the homepage at <http://liboscqs.sourceforge.net/>.

## 1 Supported Commands and responses

All commands normally respond with a `/reply` message. If there is a failure processing the command a `/error` message is returned. The `/error` message is detailed in the next section. This section will describe the successful replies send for each command.

### 1.1 Dynamic Commands

The following commands have a variable *path* component. When enabled (see section 2), these commands are automatically enabled for any methods added.

→ /

→ **/path/**

Does namespace exploration. The first form explores the root path. Replies contain all addresses that are present at the path specified:

```
← /reply (sss) '/' 'path' 'two'
```

For a leaf node no addresses are returned:

```
← /reply (s) '/path/'
```

---

<sup>1</sup><http://www.cnmat.berkeley.edu/OpenSoundControl>

<sup>2</sup><http://www.ietf.org/rfc/rfc2782.txt>

→ **/path/documentation**

Obtains user documentation. Reply is:

```
← /reply (ss) '/path/documentation' 'descriptive text'
```

→ **/path/type-signature**

Returns the expected input message layout for the *path*. Reply is:

```
← /reply (ss) '/path/type-signature' 'i'
```

→ **/path/return-type-signature**

Gives the message layout for the message returned from *path*. Reply is:

```
← /reply (ss) '/path/return-type-signature' 'i'
```

→ **/path/current-value**

If *path* is a readable parameter, this returns the current value of that parameter. Reply is:

```
← /reply (si) '/path/current-value' 50
```

→ **/path 80**

If *path* is a writable parameter, this sets that parameter to the value given. Reply is:

```
← /reply (si) '/path' 80
```

## 1.2 Static Commands

The following commands use the *'osc'* container. These commands have no variable part in their address. They should not be defined directly by applications, though applications can define other methods in the *'osc'* container.

→ **/osc/schema**

Get the URL to the schema for the application. Reply is:

```
← /reply (ss) '/osc/schema' 'http://example.com/path.html'
```

→ `/osc/ping`

Check if the other party is running. Reply is:

```
← /reply (s) '/osc/ping'
```

→ `/osc/destination/add category [[host]:[port]]`

Add an output patch for messages matching the *category*. *category* is an integer number holding a bitmap. If the optional host and port are not specified, matching messages are sent to the originator that added the patch. Reply is:

```
← /reply (sis) '/osc/destination/add' category 'host:port'
```

→ `/osc/destination/del category [[host]:[port]]`

Remove an output patch for messages matching the *category* and the *host:port*. If the optional host and port are not specified, the originating host and port are used. Reply is:

```
← /reply (sis) '/osc/destination/del' category 'host:port'
```

→ `/osc/destination/list`

Returns a list of all output patches. The list consist of a number of

```
← /reply (sis) '/osc/destination/list' category 'host:port'
```

replies, and is terminated by a reply containing:

```
← /reply (sis) '/osc/destination/list' 0 ''
```

## 2 Error messages

The format of an error message is:

```
← /error (sss) address code description
```

The *address* is the path of the originating command. *code* is a short string defining the error code. *description* contains a longer description of the error.

Table 1: Flags for Command selection

Command	flags
/	OSCQS_EXPLORE
/path/	OSCQS_EXPLORE
/path/documentation	OSCQS_DOCUMENTATION
/path/type-signature	OSCQS_TYPES
/path/return-type-signature	OSCQS_TYPES
/path/current-value	OSCQS_PARAMETER
/path	OSCQS_PARAMETER
/osc/ping	OSCQS_PING
/osc/destination/add	OSCQS_PATCH
/osc/destination/del	OSCQS_PATCH
/osc/destination/list	OSCQS_PATCH
catch any undefined path	OSCQS_GENERIC

### 3 Command Selection

All commands except `/osc/schema` are optional, and can be turned on using the *flags* attribute of the *init\_data* parameter to *oscqs\_init()*. The flags are summarized in table 1.

Shortcuts exists for the most common combinations:

**OSCQS\_DEFAULTS** All commands except the generic handler.

**OSCQS\_ALL\_SERVICES** All commands including the generic handler.

For more details refer to the `oscqs.h` manpage or the online API on the OSCQS homepage.