

# Remote Timeline Toolbox v. 1.2

*remote control of a timeline using OSC*

© 2004, 2007 Eric Marty.

ICE (Ideas for Creative Exploration, University of Georgia)

For Max 4.5 or later

download from <http://www.erimarty.com/Software.html>

## Abstractions

OSC-timeline

OSC-clock

chronometer

## GUI Abstractions (for bpatchers)

edit-OSC-timeline

control-OSC-clock

control-OSCTimeTag

ms-to-hms

Requires CNMAT's OSC external objects (OpenSoundControl 1.9.9 or later, and OSC-route)

If using Max 4.5, requires CNMAT's otudp external object.

<http://www.cnmat.berkeley.edu/MAX/downloads/>

## OSC-timeline

---

OSC-timeline is the main abstraction. It maintains a timeline of OSC messages and handles output of messages at the proper time. Every OSC-timeline contains a unique timeline name. Multiple OSC-timeline objects can pass timelines to each other, but do not share the same actual file. The timeline is stored in a uniquely named coll object, and can be saved as a separate coll file. OSC-timeline contains the abstraction OSC-clock.

### input:

open [file (optional)]

The word open with no arguments puts up a standard Open Document dialog box for choosing a coll file to load into OSC-timeline. If open is followed by a symbol filename argument, the named file is located and loaded.

saveas [file (optional)]

Calls up the standard Save As dialog box, enabling the user to save the timeline as a separate coll file. If the word saveas is followed by a symbol, the contents of the coll are saved immediately in a file, using the symbol as the filename.

save

Saves the contents of the coll into the most recently written file. If no prior save or saveas message has been received, save is treated as a saveas message, and a Save As dialog box is opened.

dump

Sends the entire contents of the timeline out the first outlet as a series of OSC packets. The contents of the timeline are added to all timelines downstream.

dumpclean

Performs a dump, but first sends an OSC packet containing the message /timeline/clear out the first outlet. The contents of the timeline therefore replaces the contents of all downstream timelines.

timeline-editor-name [name]

The word “timeline-editor-name” followed by the name of a receive object tells OSC-timeline to send it’s self-generated timeline name to the named receive. This is used to create a link between edit-OSC-timeline and OSC-timeline. The GUI abstraction edit-OSC-timeline has a receive with a unique self-generated name; edit-OSC-timeline sends the timeline-editor-name message out it’s outlet once every second. This allows the link to be created automatically when edit-OSC-timeline is patched to OSC-timeline. The link persists until a different edit-OSC-timeline is patched to the same OSC-timeline.

FullPacket [id]

FullPacket messages are pointers to OSC packets to be decoded. FullPacket messages can be received from the left outlet of another OSC-timeline object, from the left outlet of an OpenSoundControl object (third-party external from CNMAT), from the udpreceive object with a second symbol argument (see reference manual for udpreceive), and from the older otudp object (third-party external from CNMAT).

OSCTimeTag [int] [int]

The word OSCTimeTag, followed by 2 integers is interpreted by OSC-timeline as a time tag. The first integer is ignored, and the second integer is interpreted as milliseconds.

The message “OSCTimeTag 0 1” is a special case and means, “do it immediately.”

OSC address patterns sent to OSC-timeline will be executed immediately if the last OSCTimeTag message received was “OSCTimeTag 0 1” ; otherwise, they are added to the timeline as specified by the last received OSCTimeTag message.

“OSCTimeTag 0 2” corresponds to the earliest possible time in the timeline (2ms). The maximum time tag is “OSCTimeTag 0 2147483647” (596 hours, 31 minutes, 23.646999 seconds.)

NOTE: This is an unorthodox interpretation of OSC time tags. In the OSC Protocol, an OSC Time Tag is a 64 bit fixed point, with the top 32 bits giving number of seconds since midnight, January 1, 1900 and the bottom 32 bits giving fractional parts of a second. This follows the NTP protocol. Future versions of OSC-timeline should allow the additional possibility of interpreting time tags as NTP time stamps, as specified in the OSC protocol.

[OSC Address Pattern]

OSC messages (or address patterns) are added to a bundle. Upon receipt of a bang, OSC-timeline outputs time-tagged bundle. The bundle is implemented immediately (if the time tag is OSCTimeTag 0 1) or added to the timeline, and passed out the left outlet. Certain OSC messages control the timeline itself.

OSC address patterns are OSC messages that may contain special characters for pattern matching, such as \*, ?, brackets, and braces. See documentation for OSC-route (third-party external from CNMAT) for how to construct OSC address patterns (OSC messages).

|                               |   |
|-------------------------------|---|
| bang                          | Closes the OSC bundle and sends the bundle. An OSC message or bundle must be followed by a bang to be output.   |
| /timeline/clear               | Preceded by OSCTimeTag 0 1 and followed by a bang, clears the timeline and all timelines downstream.  |
| /timeline/remove [int]        | The int is a time index in ms. Preceded by OSCTimeTag 0 1 and followed by a bang, removes the given time index and all associated messages from the timeline and from all downstream timelines.   |
| /clock/synch [int]            | The int is a time in ms. Preceded by OSCTimeTag 0 1 and followed by a bang, resets the clock and all downstream clocks to time [int]. The synch time can be a negative number (useful for countdown to the start of the timeline).  |
| /clock/start                  | Preceded by OSCTimeTag 0 1 and followed by a bang, starts (or resumes) the clock and all downstream clocks.   |
| /clock/pause                  | Preceded by OSCTimeTag 0 1 and followed by a bang, pauses the clock and all downstream clocks.  |
| /clock/display_interval [int] | The int is a time interval in ms. Preceded by OSCTimeTag 0 1 and followed by a bang, changes the display interval for the clock and all downstream clocks to time interval [int]. The display interval affects the accuracy of the timed OSC messages: Each interval, OSC-clock (inside OSC-timeline) outputs the time. OSC-timeline then outputs all OSC messages scheduled for between that time and the previously output time. Therefore, OSC messages from OSC-timeline may be late by as much as 1 display interval. Default = 20ms |

**Arguments:** none

**output:**

|                       |   |
|-----------------------|---|
| FullPacket [id]       | Out left outlet. Pointer to OSC packets formatted for input to udp send, to OpenSoundControl, or to another OSC-timeline. |
| [OSC Address Pattern] | Out middle outlet. OSC messages or address patterns output from timeline on schedule.                                     |
| int                   | Out right outlet. Current time in ms.   |

## OSC-clock

---

OSC-clock is controlled by OSC messages and relies on the local cpu's sense of time for accuracy. The clock has an internal micro-second precision, sub-ms accuracy. "OSC-clock" is burried in the "OSC-timeline" patch. OSC-clock contains the abstraction chronometer. Overdrive will slightly improve the accuracy of the output.

### Input:

|                               |  |
|-------------------------------|--|
| /clock/synch [int]            | The int is a time in ms. Preceded by OSCTimeTag 0 1 and followed by a bang, resets the clock and all downstream clocks to time [int]. The synch time can be a negative number (usefull for countdown to the start of the timeline).  |
| /clock/start                  | Preceded by OSCTimeTag 0 1 and followed by a bang, starts (or resumes) the clock and all downstream clocks.  |
| /clock/pause                  | Preceded by OSCTimeTag 0 1 and followed by a bang, pauses the clock and all downstream clocks.   |
| /clock/display_interval [int] | The int is a time interval in ms. Preceded by OSCTimeTag 0 1 and followed by a bang, changes the display interval for the clock and all downstream clocks to time interval [int]. The display interval affects the accuracy of the timed OSC messages: Each interval, OSC-clock (inside OSC-timeline) outputs the time. OSC-timeline then outputs all OSC messages scheduled for between that time and the previously output time. Therefore, OSC messages from OSC-timeline may be late by as much as 1 display interval. If no display interval has been set either with this message or as an argument, then the display interval is as short as possible, which seems to be about .05ms ( $\mu$ s) |

### Argument: (optional)

|              |   |
|--------------|---|
| Int or float | sets an initial value for the time interval at which OSC-clock sends its output (according to Max's internal clock.) If there is no argument, the initial time interval is as short as possible, which seems to be about .05ms ( $\mu$ s) |
|--------------|---|

### output:

|       |  |
|-------|--|
| Float | Out left outlet. Current time in ms.   |
| Bang  | Out right outlet. bang is sent after a /clock/synch message is receive in the inlet. |

## chronometer

---

chronometer is a time keeper that uses the system clock (cpuclock) to accurately report the time since MaxMSP started (the default behavior), or the time counting from the last time number received in the right inlet.

Chronometer is buried in OSC-clock.

### input:

Bang In left inlet. Causes current time to be reported out left outlet.

Float in right inlet. "Synch time." Sets time in ms to float.

### Argument (optional):

0 or -1 Sets direction of time. 0 = forwards (default); -1 = backwards

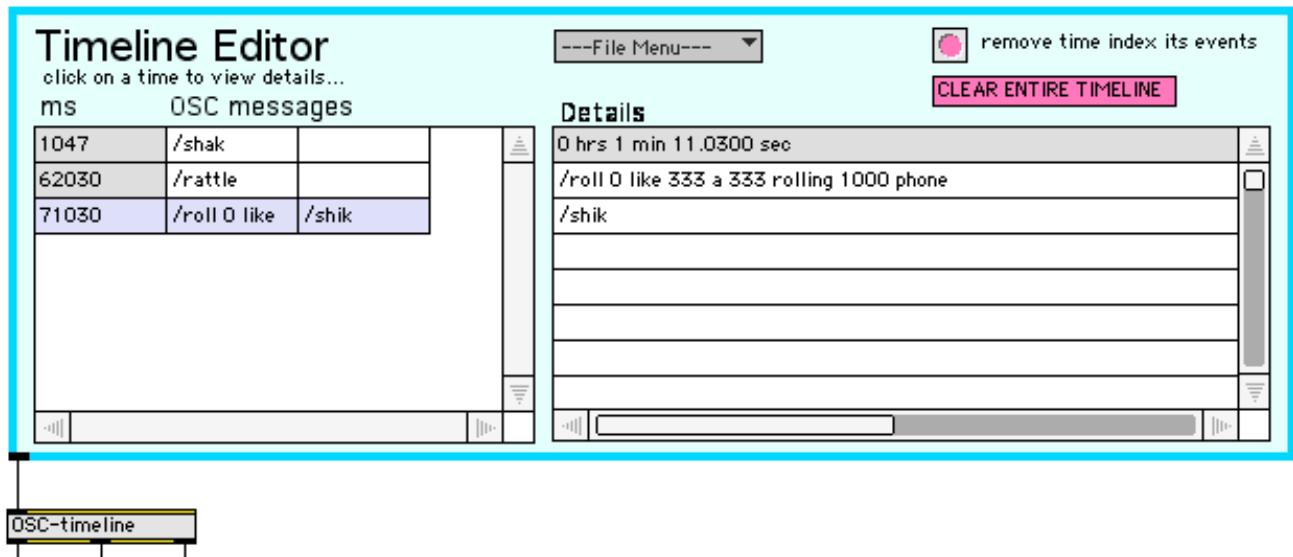
### output:

Float Out left outlet. Current time in ms.

Bang Out right outlet. bang is sent after a new synch time has been received in the right inlet.

## edit-OSC-timeline

edit-OSC-timeline connects to OSC-timeline and simplifies the display and editing of the timeline in OSC-Timeline. Once the objects are patched together, a link is automatically created and the timeline data is shared.



Clicking on a time index at left displays the details for that time index at right.

Clicking the button marked “remove time index and its events” sends OSCTimeTag 0 1, /timeline/remove [time\_index], bang. This clears the time index and its associated events from the timeline and from all downstream timelines.

Clicking on “CLEAR THE ENTIRE TIMELINE” sends OSCTimeTag 0 1, /timeline/clear, bang. This clears the timeline and all downstream timelines.

### File Menu:

open, saveas, save, dump, dumpclean see OSC-timeline for explanation of these commands.

timeline-editor-name [name]

Once every second, the word “timeline-editor-name” followed by the unique self-generated name of a receive object inside edit-OSC-timeline is passed out the outlet. If connected to OSC-timeline, it tells OSC-timeline to send it’s self-generated timeline name to the named receive. This is used to create a link between edit-OSC-timeline and OSC-timeline. This allows the link to be created automatically when edit-OSC-timeline is patched to OSC-timeline. The link persists until a different edit-OSC-timeline is patched to the same OSC-timeline.

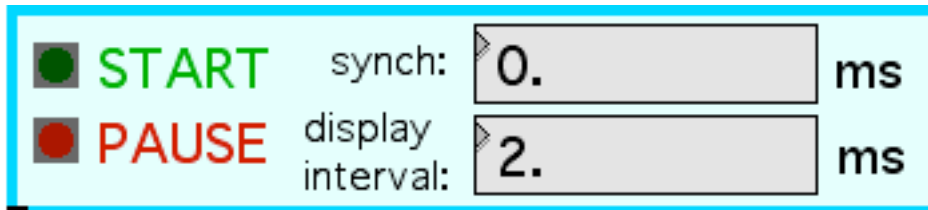
### output:

timeline-editor-name [name]

Once every second, the word “timeline-editor-name” followed by the unique self-generated name of a receive object inside edit-OSC-timeline is passed out the outlet. If connected to OSC-timeline, it tells OSC-timeline to send it’s self-generated timeline name to the named receive. This is used to create a link between edit-OSC-timeline and OSC-timeline. This allows the link to be created automatically when edit-OSC-timeline is patched to OSC-timeline. The link persists until a different edit-OSC-timeline is patched to the same OSC-timeline.

## control-OSC-clock

Connects to OSC-timeline and provides a simplified means of starting, pausing and synching the clock.

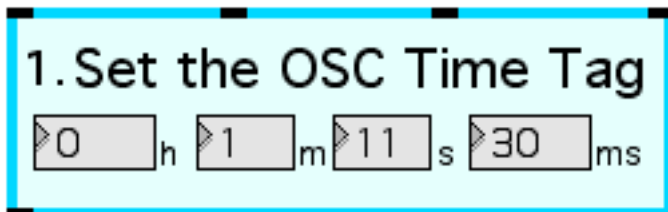


### output:

|                               |   |
|-------------------------------|---|
| /clock/start                  | On pressing 'START,' control-OSC-clock sends out OSCTimeTag 0 1, /clock/start, bang. Starts (or resumes) the clock and all downstream clocks.   |
| /clock/pause                  | On pressing 'PAUSE,' control-OSC-clock sends out OSCTimeTag 0 1, /clock/pause, bang. Starts (or resumes) the clock and all downstream clocks.   |
| /clock/synch [int]            | <p>On changing the "synch" number, control-OSC-clock sends out OSCTimeTag 0 1, /clock/synch [int], bang. Starts (or resumes) the clock and all downstream clocks. [int] is the time in ms.</p> <p>Can be a negative number (useful for countdown to the start of the timeline).</p> |
| /clock/display_interval [int] | On changing the "dsplay interval" number, control-OSC-clock sends out OSCTimeTag 0 1, /clock/dusplay_interval [int], bang. Sets the display interval for the clock and all downstream clocks. [int] is the interval in ms.  |

## control-OSCTimeTag

Sends out an OSCTimeTag message based on the hours, minutes seconds and ms input.



### input:

Int in inlets 1 – 4 changes the hours, minutes, seconds and ms respectively. Any input causes output of a new OSCTimeTag

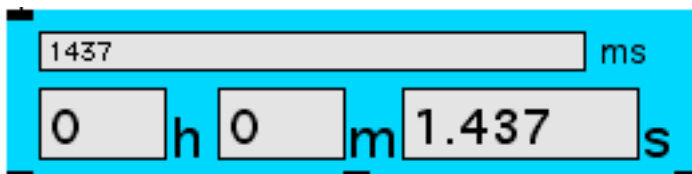
### Output:

Int OSCTimeTag 0 [int] (where [int] is a time in ms)

NOTE: This is an unorthodox interpretation of OSC time tags. In the OSC Protocol, an OSC Time Tag is a 64 bit fixed point, with the top 32 bits giving number of seconds since midnight, January 1, 1900 and the bottom 32 bits giving fractional parts of a second. This follows the NTP protocol. Future versions of OSC-timeline should allow the additional possibility of interpreting time tags as NTP time stamps, as specified in the OSC protocol.

## ms-to-hms

Converts ms to hours, minutes and seconds.



### input:

Int ms

### Output:

Int Out left outlet = hours

Out middle outlet = minutes

float Out right outlet = seconds