Recording Service

The following details the recording service API communictation using XML-RPC over HTTP. The TCP port number for a given recording channel can be calculated as 4242 + N where N is the number of the recording service (zero based).

The recording service has a queue of recording jobs. There may be at most one current job at any given time. All jobs that are "above" the current job are either in "Done" or "Skipped" state. If there is no current job it means that all the jobs are either in "Done" or "Skipped" state.

All duration and times are specified either in frames (when an Integer type variable is used) or julian days (when an Double type variable is used). See the conversion of calendar time to Julian days and vice versa.

Calculation of calendar time to Julian days.

XML-RPC Methods

rec_add_job

Creates a new job

```
parameter 0 - the structure describing the recorded field clip is as
follows:

    int "src" - source type
        0 - VTR
        1 - LIVE
    string "src_name" - source name, tape name for a VTR job
    double "start_time" - start time for this job, -1 if unspecified
    int "start_type" - job start type
        0 - manual start
        1 - automatic start after the previous job finishes
        2 - hard time start
    int "in_point" - in point for VTR jobs
    int "duration" - job duration
    string "type" - "Type" field for the media asset to be created
    int "have_title" - 0 if the "title" field is not set (autogenerated by
the service),
                       non-zero if the "title" field is set manually
    string "title" - media ID for the asset to be created
    int "have_folder" - 0 if the "folder" field is not set (autogenerated by
the service),
                        non-zero if the "folder" field is set manually
    string "folder" - folder for the asset to be created
    string "comment" - comment field for the asset to be created
    string "group" - job block name
```

```
    int "src_pin" - video router source pin number or -1 if no change
    int "on_exists" - action to be performed if an asset with the same
"title" already exists.
                     possible values are: 0 - rename, 1 - skip, 2 -
replace, 3 - attach
    int "dtl" - number of days after which the asset will be deleted by the
storage server
                (sets the "Delete At" property of the media asset)
    int "chunk_size" - chunk size in frames for chunking recording or -1 for
normal mode
    int "before" - ID of the job before which a newly created item will be
inserted or                    -1 if inserting after the last element return
value - int - ID of the newly created job or -1 if an error occured
```

rec_del_job

Deletes record job

```
parameter 0 - int - ID of the job to be deleted
return value - int: 1 - if successful, 0 if failed
```

rec_start_job

Starts the current job

```
parameters - none
return value - none
```

rec_stop_job

Stops the current job if it's active. The job following the current becomes a new current job

```
parameters - none
return value - none
```

rec_pretake_job

Cues the current job

```
parameters - none
return value - none
```

rec_list_jobs

retrieves job list

```
parameter 0 - double - server time when the previous call to "rec_list_jobs"
was executed or
-1 if unspecified. The returned list will be calculated as the difference
since this time stamp.
return value - a struct that describes the job list
    int "full_update" - 1 - the returned list contains all the data.
                              (the previous time stamp is -1 or items have
been added or deleted
                               since previous time stamp)
                        0 - the returned list contains only the different to
the previously
                              returned value
    double "srv_time" - current server time that can be used in next calls
as the parameter
                        for the difference calculation
    int "empty" - 1 - the "lst" array present
                - 0 - the "lst" array is absent
    array - "lst" - array with the difference to the previously returned
one.
                        Each element of the array is a struct with the
following fields:
        mandatory fields:
          int "idx" - job id
          double "frame_time" - 1/fps for the current video mode
          int "order" - element order in the job queue (zero based)
        optional fields (if absent the previous value should be used):
          int "src" - source type
            0 - VTR
            1 - LIVE
          string "src_name" - source name, tape name for a VTR job
          double "start_time" - start time for this job, -1 if unspecified
          int "start_type" - job start type
            0 - manual start
            1 - automatic start after the previous job finishes
            2 - hard time start
          int "in_point" - in point for VTR jobs
          int "duration" - job duration
          string "type" - "Type" field for the media asset to be created
          int "have_title" - 0 if the "title" field is not set
(autogenerated by the service),
                              non-zero if the "title" field is set manually
          string "title" - media ID for the asset to be created
          int "have_folder" - 0 if the "folder" field is not set
(autogenerated by the service),
                              non-zero if the "folder" field is set
manually
          string "folder" - folder for the asset to be created
          string "comment" - comment field for the asset to be created
          string "group" - job block name
```

```
            int "src_pin" - video router source pin number or -1 if no
change
            int "on_exists" - action to be performed if an asset with the
same "title" already exists.
                            possible values are: 0 - rename, 1 - skip, 2 -
replace, 3 - attach
            int "dtl" - number of days after which the asset will be deleted
by the storage server
                        (sets the "Delete At" property of the media asset)
            int "chunk_size" - chunk size in frames for chunking recording
or -1 for normal mode

            int "state" - job execution state
              0 - some of the job parameters are invalid and the job can't
be executed
              1 - job parameters are good and the job can be executed
              2 - job is cuing
              3 - job is cured and is ready for an immediate start
              4 - job is running
              5 - job is finished
              6 - job is paused
            int "status" - job queue status
              0 - job is below current
              1 - job is current
              2 - job is above current and was skipped or interrupted
              3 - job if above current and successfuly finished execution
```

rec_get_info

retrieves information about a job

```
parameter 0 - int - ID of the job
return value - a struct that describes the job
  fields:
    int "exists" - 1 - job with the ID specified exists
                   0 - job with the ID specified doesn't exist
    double "srv_time" - current server time
    double "frame_time" - 1/fps for the current video mode

    int "src" - source type
       0 - VTR
       1 - LIVE
    string "src_name" - source name, tape name for a VTR job
    double "start_time" - start time for this job, -1 if unspecified
    int "start_type" - job start type
       0 - manual start
       1 - automatic start after the previous job finishes
       2 - hard time start
    int "in_point" - in point for VTR jobs
    int "duration" - job duration
```

```
    string "type" - "Type" field for the media asset to be created
    int "have_title" - 0 if the "title" field is not set (autogenerated by
the service),
                       non-zero if the "title" field is set manually
    string "title" - media ID for the asset to be created
    int "have_folder" - 0 if the "folder" field is not set (autogenerated by
the service),
                       non-zero if the "folder" field is set manually
    string "folder" - folder for the asset to be created
    string "comment" - comment field for the asset to be created
    string "group" - job block name
    int "src_pin" - video router source pin number or -1 if no change
    int "on_exists" - action to be performed if an asset with the same
"title" already exists.
                       possible values are: 0 - rename, 1 - skip, 2 -
replace, 3 - attach
    int "dtl" - number of days after which the asset will be deleted by the
storage server
                  (sets the "Delete At" property of the media asset)
    int "chunk_size" - chunk size in frames for chunking recording or -1 for
normal mode

    int "state" - job execution state
     0 - some of the job parameters are invalid and the job can't be
executed
     1 - job parameters are good and the job can be executed
     2 - job is cuing
     3 - job is cured and is ready for an immediate start
     4 - job is running
     5 - job is finished
     6 - job is paused
    int "status" - job queue status
     0 - job is below current
     1 - job is current
     2 - job is above current and was skipped or interrupted
     3 - job if above current and successfuly finished execution
```

rec_set_info

changes job description

```
 parameter 0 - a struct that describes a record job
   fields:
    int "idx" - ID of the job to be changed
    int "src" - source type
       0 - VTR
       1 - LIVE
    string "src_name" - source name, tape name for a VTR job
    double "start_time" - start time for this job, -1 if unspecified
    int "start_type" - job start type
```

```
      0 - manual start
      1 - automatic start after the previous job finishes
      2 - hard time start
    int "in_point" - in point for VTR jobs
    int "duration" - job duration
    string "type" - "Type" field for the media asset to be created
    int "have_title" - 0 if the "title" field is not set (autogenerated by
the service),
                       non-zero if the "title" field is set manually
    string "title" - media ID for the asset to be created
    int "have_folder" - 0 if the "folder" field is not set (autogenerated by
the service),
                        non-zero if the "folder" field is set manually
    string "folder" - folder for the asset to be created
    string "comment" - comment field for the asset to be created
    string "group" - job block name
    int "src_pin" - video router source pin number or -1 if no change
    int "on_exists" - action to be performed if an asset with the same
"title" already exists.
                      possible values are: 0 - rename, 1 - skip, 2 -
replace, 3 - attach
    int "dtl" - number of days after which the asset will be deleted by the
storage server
                (sets the "Delete At" property of the media asset)
    int "chunk_size" - chunk size in frames for chunking recording or -1 for
normal mode
return value - int - 1 is successfuly changed, 0 - an error occured
```

rec_set_processing

Sets the flag for processing the task list.

```
parameter 0 - int - 1 when the list of tasks are processed by default, or 0
in case of line-by-line execution of the list of tasks, in this case the
tasks are not started automatically, only when rec_start_job is explicitly
called
the return value is no
```

rec_get_processing

Gets the status of the task list processing flag.

```
parameters - no return value - int - 1 when the list of tasks are processed
by default, or 0 if the list of tasks is executed line by line, in this case
the tasks are not started automatically, only when rec_start_job is
explicitly called
```

rec_restart

stops any active jobs and sets the first item in the record list as current

```
parameters - none
return value - none
```

rec_jump

stops any active jobs and sets a new current job

```
parameter 0 - int - ID of the job to be set as current
return value - none
```

From:
http://wiki.skylark.tv/ - **wiki.skylark.tv**

Permanent link:
http://wiki.skylark.tv/api/recsrv

Last update: **2024/12/13 10:53**