Atacama
Large
Millimeter
Array

Scheduling Subsystem
User Guide for the Create Array Panel, and Scheduling Panel

Version: R8.1.1

Status: Draft

2011-08-17

Prepared by:	**Organization:**	**Date:**
David Clarke	NRAO	08/17/11
IPT Leader Approvals:	**Organization:**	**Date:**
Released By:	**Organization:**	**Date:**

Change Record

Version	Date	Affected section(s)	Change Request #	Reason/Initiation/Remarks
A	2006-08-22	All	Created	
A	2006-08-28	All	Modified	Clarified portions after input from readers
A	2006-10-05	3.2, 4.1, 4.3, 5, 6	Updated	Updated to reflect software state at R4
A	2006-10-26		Updated & Added Snapshots	Updated for Holography user test.
A	2006-12-14	All	Modified	Update for new Scheduling Panel
A	2007-03-14	3, 4, 5	Redid all snapshots, modified almost all text.	Updated for GUI FBT enhancements/changes
A	2007-03-27	4.1, 4.8, 5		Updated a few things for the AntennaVerification branch
A	2007-05-01	4.4		Fixed mismatch to software
A	2007-05-07	4.3, 4.7		Added new section about status reporting of a SB's execution
A	2007-08-03	1.1, 1.2, 3.1, 3.2, 4.1-4.4, 4.7, 6		Update to reflect changes made during the GUI2 FBT
A	2007-11-16	4.4, 5, 6		Updated queued scheduling section
A	2009-11-25	All	Modified	Redid snapshots and text for R7.0. Recreated document after formatting loss.
A	12/15/10	All	Modified for R8.0	Reworked for refactored (R8.0) Scheduler

A	08/17/11	4, 5	R8.1.1 update	Initial release of Dynamic Scheduler

Table of Contents

1. Description....6
1.1. Purpose....6
1.2. Scope....6
2. Related Documents and Drawings....6
2.1. References....6
2.2. Abbreviations and Acronyms....6
3. Startup....6
4. Functionality....7
4.1. General GUI Information and Functionality....7
4.1.1. Different types of schedulers....7
4.1.2. Existing arrays tab....8
4.2. Array Creation....8
4.3. Interactive Scheduling....9
4.3.1. Array Panel Overview....9
4.3.2. Project Table....10
4.3.3. SchedBlock Table....10
4.3.4. Project and SchedBlock Details....11
4.3.5. Pending Executions....11
4.3.6. Current Executions....12
4.3.7. Recent Executions....12
4.4. Basic Operation – Interactive Scheduler....13
4.4.1. Selecting And Queuing....13
4.4.2. SchedBlock Execution....14
4.5. Dynamic Scheduling....15
4.5.1. Overview....15

4.5.2. Scheduling Policies.....15
4.6. Getting New or Updated Projects and SchedBlocks.....18
4.7. Creating a Manual Array.....18
4.8. Closing a Scheduler plug-in.....19
4.9. Destroying an array.....19
5. List of Known Issues.....19
6. Next official update.....19

1. Description

1.1. Purpose

Provide information on how to create arrays and use the Scheduler Panels.

1.2. Scope

How to create arrays and how to use the Scheduler Panel features only.

2. Related Documents and Drawings

2.1. References

2.2. Abbreviations and Acronyms

OMC	Operator Master Client from EXEC.
Plug-in	A graphical component within the OMC that can be docked or floating from the OMC.
GUI	Graphical User Interface.
ALMA-OT	ALMA Observing Tool.
CDB	Configuration Data Base.
SB	Scheduling Block.
CCL	Control Command Language.

3. Startup

Instructions on how to configure the R8.1.1 Scheduling subsystem can be found at http://almasw.hq.eso.org/almasw/bin/view/ITS/CdbPendingChanges#Setup_for_the_new_Scheduler.

The Create Array panel is a regular OMC plug-in which must be started from the plug-ins menu (if not already started from there and saved as open in the omc.layout file). See the OMC's documentation about starting plug-ins.

4. Functionality

4.1. General GUI Information and Functionality

4.1.1.Different types of schedulers

There are now two types of scheduling: Interactive scheduling and Dynamic scheduling. The two types are very similar. The principal difference is that the Dynamic scheduler will automatically evaluate the SchedBlocks according to the current Scheduling Policy and rank them accordingly (see 4.5 Dynamic Scheduling for more details). However, it does not do anything with the evaluation other than display it to the operator (in due course it will automatically queue the top ranked SchedBlock). The CreateArray GUI (Figure 1) now shows options for Interactive, Dynamic and Manual – the Queued option is greyed out.

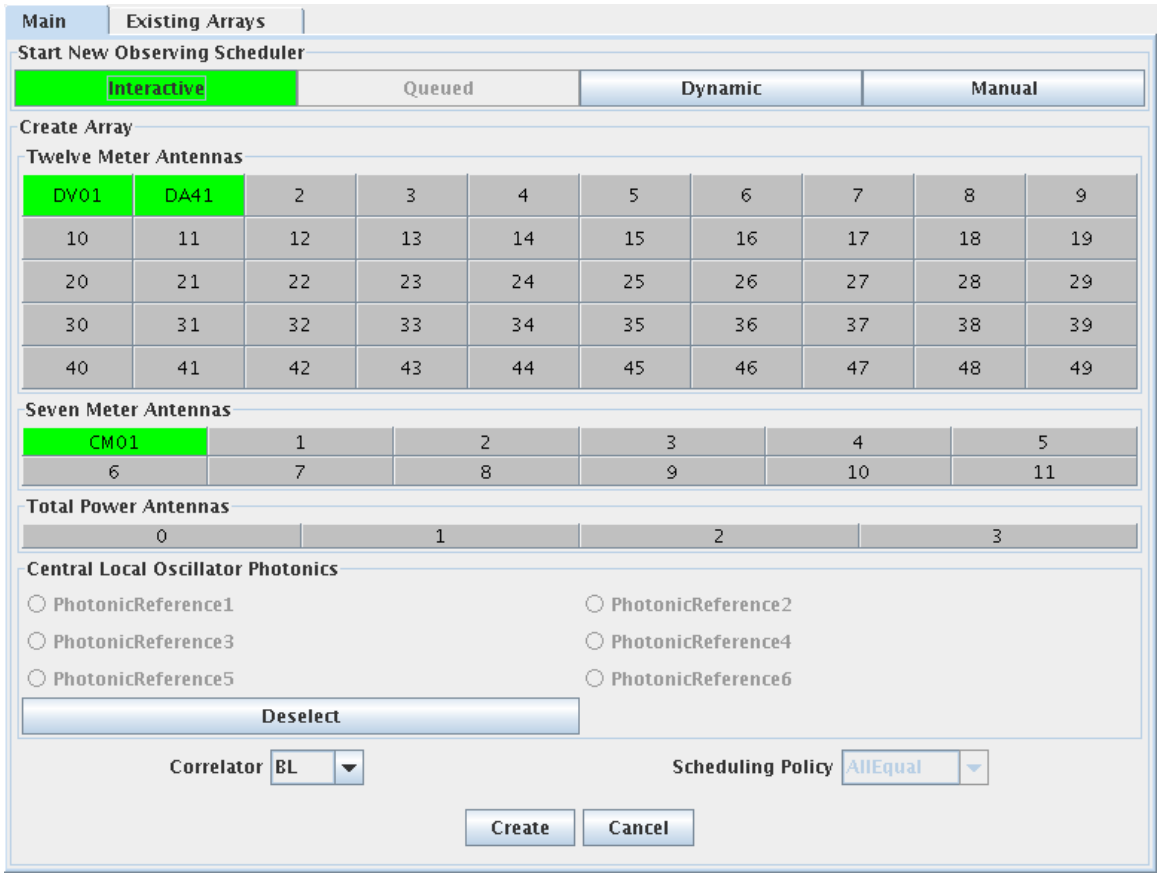

Figure 1: Create Array Panel

When you click one of these buttons the array creation area becomes enabled and the created array (see 4.2 Array Creation) will be assigned to the scheduler type chosen. At which point a new plug-in for your scheduler will be created and displayed.

4.1.2.Existing arrays tab

Another tab in the create array panel is for arrays that already exist. This tab shows all the current arrays and by selecting one then using the right mouse button you can destroy the array or open another plug-in to view and control the array, see Figure 2.

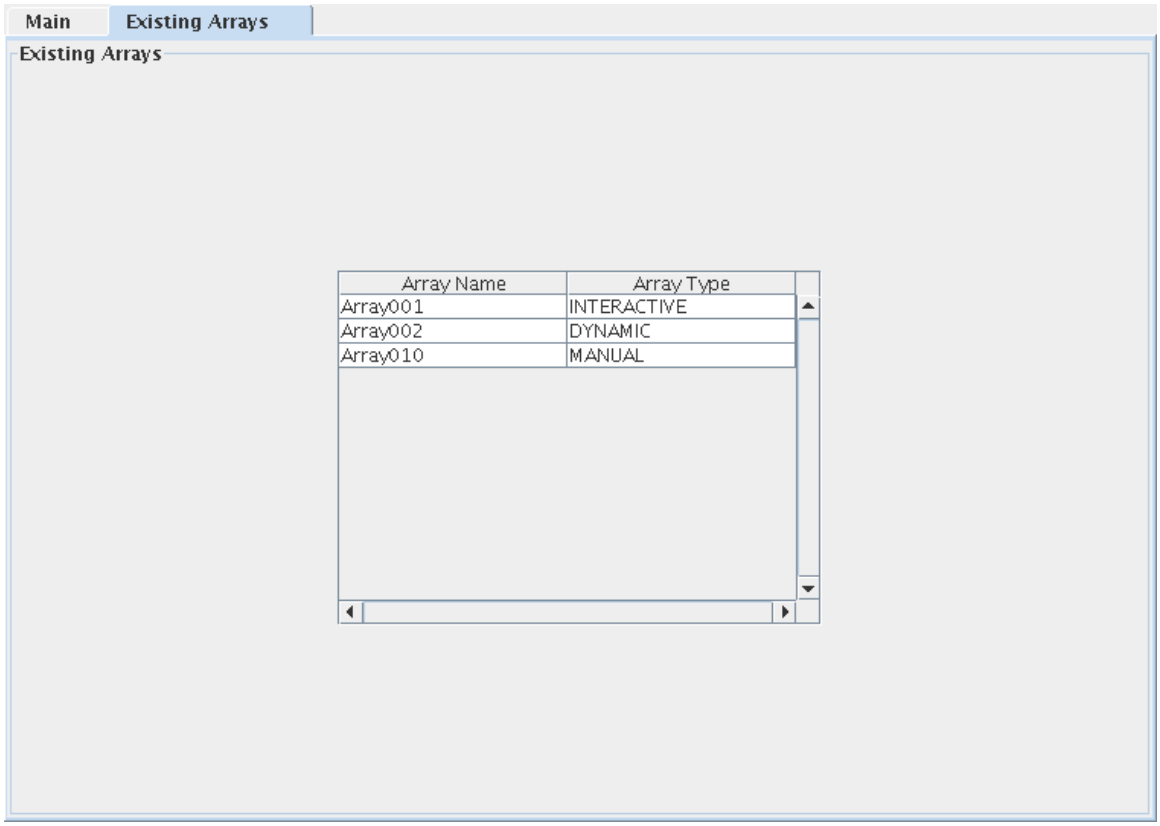

Figure 2: Existing Arrays Panel showing one of each type of array

4.2. Array Creation

As seen in Figure 1, the create array section of the scheduling panel becomes enabled when one of the top 3 buttons is pressed. The panel then shows the available antennas (in green), the available photonic references, the correlators and the available Scheduling Policies. To create an array select which of these resources are needed for the array, then press the 'Create' button. To select multiple antennas use the Ctrl or Shift keys with a left mouse click.

A photonic reference is the combination of a laser reference (1st LO photonic Reference Synthesizer Test Module - LORTM or Laser Synthesizer - LS) and the Central Variable Reference (CVR) connected to it. This also includes the Photonic Reference Distributor PRD. A photonic reference is required for any array which will use the Front End (Single Field Interferometry and TotalPower). It is not required for Tower Holography or Optical Pointing. A maximum of one photonic reference may be selected.

A Scheduling Policy is only applicable for Dynamic Scheduling (when it is, in fact, mandatory).

After the 'Create' button is pressed the whole panel will be disabled until the array is created or a message is returned saying why the array cannot be created.

4.3. Interactive Scheduling

4.3.1. Array Panel Overview

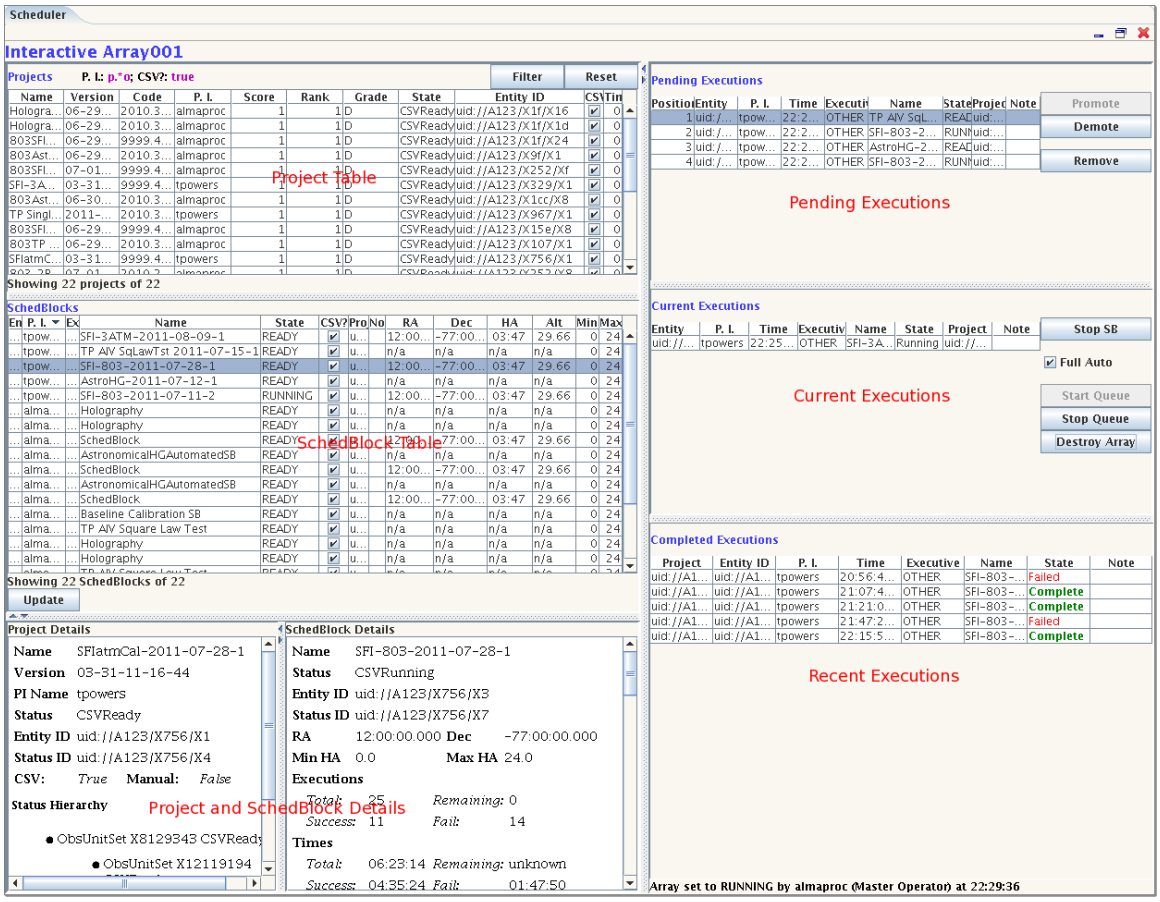

Figure 3: Array Panel for interactive scheduling, annotated to show the main parts

The display for an Interactive Array (Figure 3) has the following principal parts:

4.3.2.Project Table.

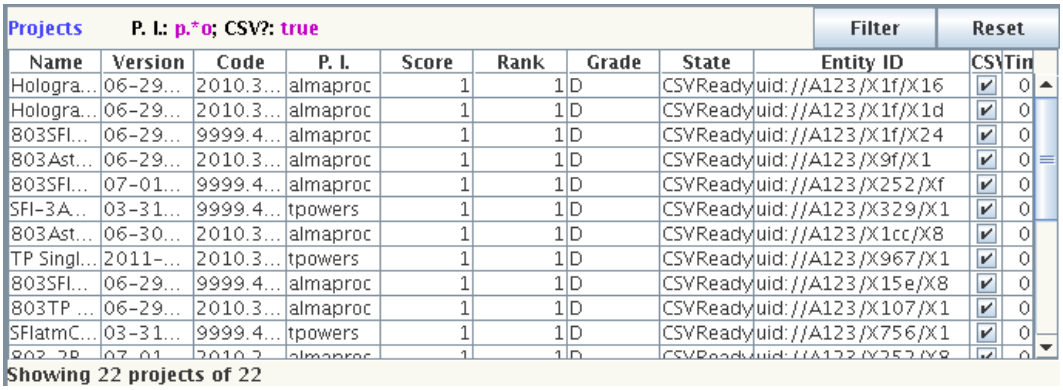

Figure 4: Project Table

This displays information on the Projects which the Scheduler knows about, and which are suitable for this array (currently, the criteria are Manual Mode or non-Manual Mode). The table can be sorted on any column by clicking that column's header. Additionally, there is a powerful set of filters which allow the operator to control which subset of projects is displayed (see 4.4.1.Selecting And Queuing).

4.3.3.SchedBlock Table.

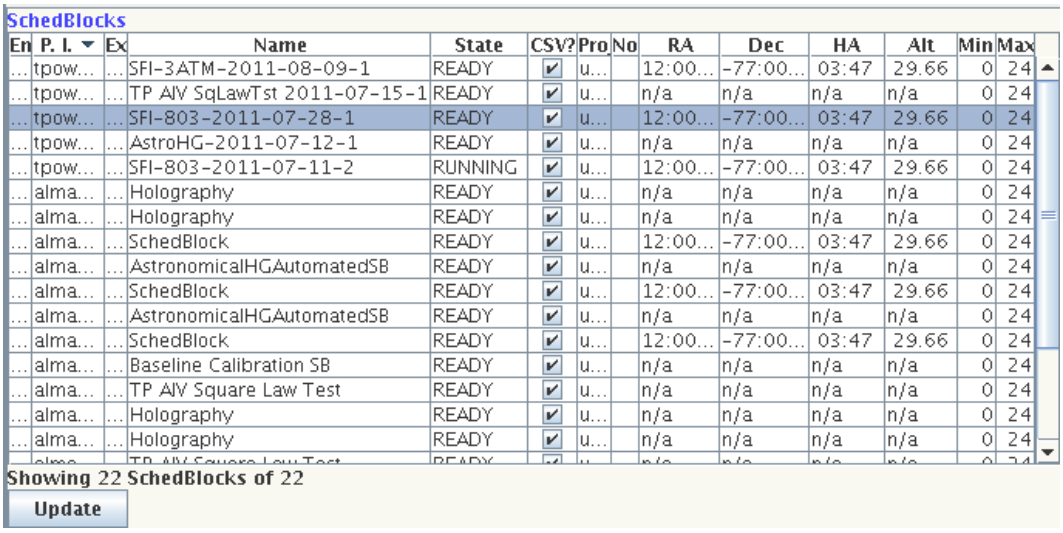

Figure 5: SchedBlock Table

This shows the SchedBlocks which are part of the Projects displayed in the Project Table. By default, the table shows all the SchedBlocks for the Projects displayed in the ProjectTable, however, selecting one or more Projects in the

Project Table will cause the SchedBlock Table to display the SchedBlocks for just those projects.

4.3.4.Project and SchedBlock Details

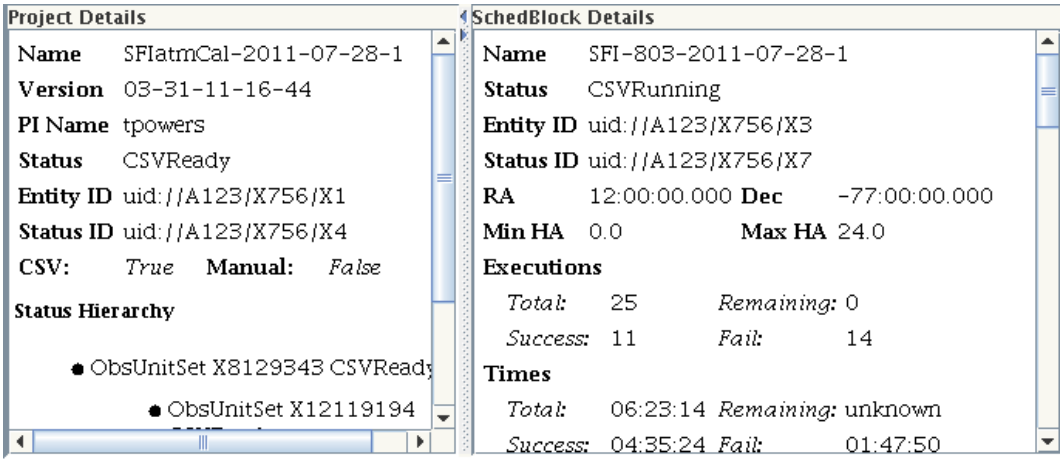

Figure 6: Project and SchedBlock Details

This shows details of the currently selected Project and SchedBlock. In the case of the SchedBlock this includes a summary of the target, outlines of past executions, and, for Dynamic Schedulers, a summary of the scheduling scores for the SchedBlock.

4.3.5.Pending Executions.

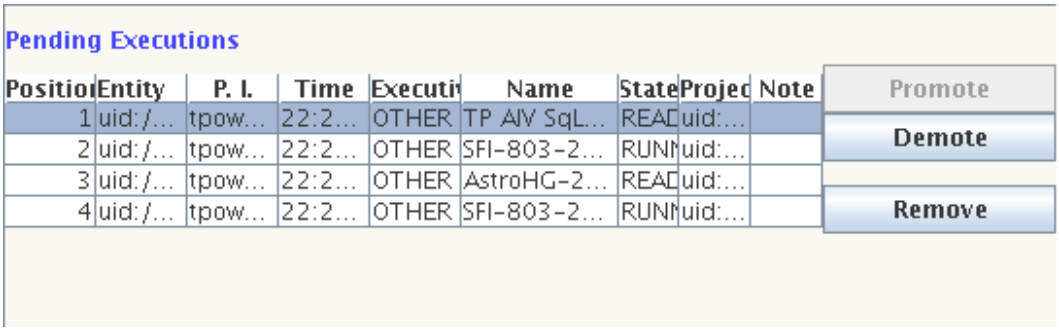

Figure 7: Pending Executions

Shows the queue of pending SchedBlock executions, along with the time at which they were queued for execution. A SchedBlock execution in the queue can be manipulated by using the Promote, Demote and Remove buttons shown alongside the queue.

4.3.6.Current Executions.

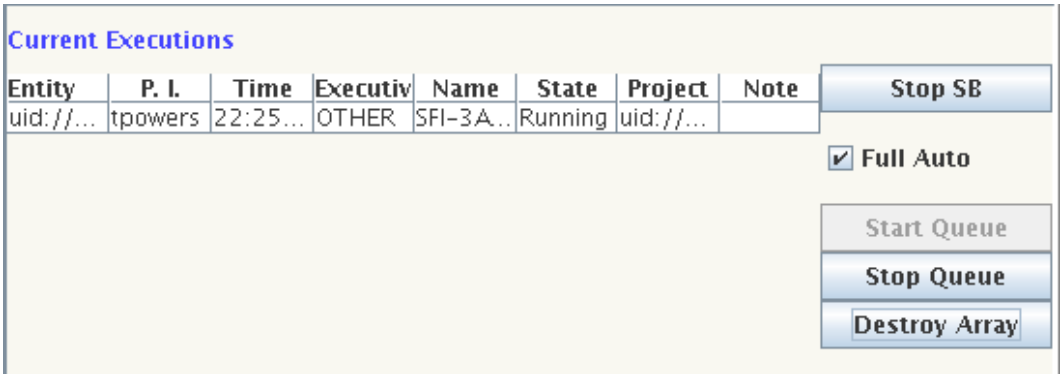

Figure 8: Current Executions

Shows the current SchedBlock executions (if any) and any which have finished observing but are waiting for their data to be archived. By default, an Interactive Scheduler starts with SchedBlock execution deactivated. To start observing, the operator should press the “Start Queue” button.

4.3.7.Recent Executions

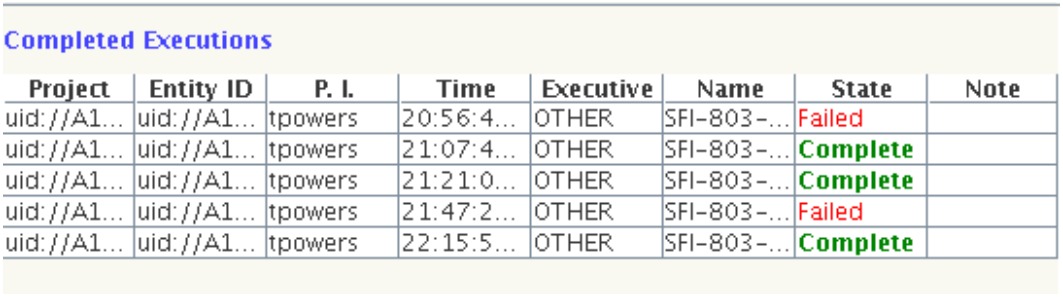

Figure 9: Recently Completed Executions

Shows SchedBlock executions which have completed while this plugin has been open. Allows the operator to review recent activity. Note that this information is stored in the Array Panel, and will be lost once the Array Panel is closed.

The horizontal and vertical dividers between these components can be dragged with the mouse to allow you to adjust the display to suit your preferences. Also, some dividers have small arrows on, clicking which will collapse parts of the display to allow you to focus on other parts. The collapsed panels can be brought back by clicking the other arrow. Multiple Array Panels can be open on the same Array, allowing the user to manage their screen real-estate: for example one Array Panel showing the Project and SchedBlock tables and one showing the executions.

4.4. Basic Operation – Interactive Scheduler.

The basic method of working with the Interactive Scheduler is to select SchedBlocks which are to be observed and to queue them for execution. Once the observing queue is started (by pressing the “Start Queue” button), each SchedBlock will be taken from the queue and executed in turn.

More SchedBlocks can be added to the queue as it is running, and the queue can be managed as described above. Note that the same SchedBlock can appear in the queue more than once, including consecutively.

4.4.1.Selecting And Queuing

To add SchedBlocks to the execution queue, firstly select one or more SchedBlocks in the SchedBlock Table, then click the right mouse button within the table. The pop-up menu which appears has two options for queueing SchedBlocks.

To queue the selected SchedBlock(s), select the “Queue Selected SchedBlock(s)” option. To queue the SchedBlock which was under the cursor at the time the menu was popped-up, select the “Queue SchedBlock <*entity-id*>” (the entity-id will be that of the SchedBlock which will be queued).

In order to help manage the list of SchedBlocks in the table, you can:

1. Sort the table on any column by clicking on the column header. Clicking again will reverse the direction of sorting.
2. Resize the columns by clicking and dragging the dividers between the column headers.
3. Rearrange the columns by dragging and dropping the headers.
4. Reduce the number of SchedBlocks displayed by either selecting a number of projects in the Project Table (in which case the SchedBlock table will show only the SchedBlocks for the selected projects), or by filtering the Project Table (q.v.).

Like those in the SchedBlock Table, the columns in the Project Table can be sorted, resized and rearranged. In addition to this, search criteria can be specified for any set of columns in the Project Table, and only the rows which match all of these criteria will be shown. This process is termed “filtering”.

To specify the filters for the Project Table, press the “Filter” button which is located just above the Project Table. A dialogue will appear with an entry for each column of the Project Table. Each entry consists of three parts:

1. A label indicating to which column in the Project Table this filter will apply.

2. A text field into which you should type a regular expression. Only rows in which the value in this column matches the regular expression will be displayed.
3. A checkbox which controls whether to use this filter or not (note that this is automatically set when you type something into the textfield described above).

Filtering happens as you type, so you can easily check that you are getting the correct results. The current filtering is also summarised just above the Project Table.

The filtering is individual to each Array Panel, so if you open the plugin again, a new set of filters will be created and used on that second plugin, independently from those on the first plugin.

Things to note about filtering:

- Matching is done in a case insensitive way (so “bob”, “BOB” and “Bob” will all match the value “Bob”).
- The search filters are regular expressions, one notable result of this is that a single asterisk is NOT a valid regular expression. To match any string use the string “.*” (without the quotation marks). You can always just switch off a particular filter.
- Boolean values are shown in the tables as check boxes which are either ticked or not. However, their values are “True” and “False”. So, as an example, to show only those projects which are CSV projects, set the CSV filter to “true” (remember, it is not case sensitive).
- If you enter a malformed regular expression, it will turn red (be aware that this can happen when you're in the middle of typing as half a regular expression will probably not be syntactically valid– if it does, just keep typing the rest of the expression and things should work out).

4.4.2.SchedBlock Execution.

As mentioned above, an Interactive Array is created with SchedBlock execution switched off. To switch it on, press the “Start Queue” button. When you do this, the first SchedBlock in the Pending Executions queue will be taken and put to the Control subsystem for execution. In the Array Panel, it will move from the Pending Executions display to the Current Executions display, and its status will be changed to Running. To monitor progress of the observation, you should open the DataFlow plugin.

Once the SchedBlock has finished execution, it's status will change to Archiving until such time as the Control subsystem notifies the Scheduler that the observation data has been archived, when the state of the Execution will be changed to Complete and the Execution will move to the Recently Completed display.

At the same time as the current SchedBlock execution is moved to Archiving, the next SchedBlock is taken from the Pending Executions queue and execution of that SchedBlock begins. This continues until either the queue is empty or observing is suspended by pressing the Stop Queue button.

In the case of the Stop Queue button being pressed, the current SchedBlock execution will complete, but after that no new SchedBlock will be taken from the Pending Queue until observing is restarted (by pressing the Start Queue button).

If observing stops because the Pending Executions queue is empty, simply add a new SchedBlock to the queue and observing will start straight away.

Should a SchedBlock execution be unsuccessful, the state of the Execution will be changed to Failed and the Execution will move to the Recently Completed display. As with a successful execution, the next Execution will be started.

4.5. Dynamic Scheduling

4.5.1.Overview

The Dynamic Scheduler plug-in has all the features of the Interactive Scheduler, with the following extras:

- The SchedBlock table shows each SchedBlock's most recent score and rank as well as their previous score and rank (if available).
- Whenever the Dynamic Scheduler tries to take a SchedBlock from the queue, if that queue is empty the scoring and ranking of SchedBlocks will be triggered.

The scoring and ranking is a two stage process. Firstly viable SchedBlocks are selected and then each of these is evaluated and assigned a score (the higher the better) based on how good it would be to observe this SchedBlock now. If the SchedBlock Table is set to sort rows based on the current rank then the SchedBlocks will be sorted into their resultant order.

4.5.2.Scheduling Policies

Whenever the SchedBlocks are scored, they are scored according to the current Scheduling Policy. This policy determines the criteria used to select viable SchedBlocks,

the factors used to score the selected SchedBlocks and the weighting to give each such factor when the overall score is calculated for a SchedBlock. The available Scheduling Policies are defined in an XML file called the Policy File.

This Policy File can contain multiple Scheduling Policies. Each Scheduling Policy has two main sections: Selection Criteria and Scorers. The following is a sample Policy File:

```
<?xml version="1.0" encoding="UTF-8"?>
<Policies>
    <SchedulingPolicy name="testPolicy1">
        <SelectionCriteria>
            <OpacitySelector/>
            <MoonAvoidanceSelector/>
            <SunAvoidanceSelector/>
            <HourAngleSelector/>
        </SelectionCriteria>
        <Scorers>
            <HourAngleScorer>
                <weight>1.0</weight>
            </HourAngleScorer>
            <SciScorer>
                <weight>1.0</weight>
            </SciScorer>
            <TsysScorer>
                <weight>1.0</weight>
            </TsysScorer>
        </Scorers>
    </SchedulingPolicy>

    <SchedulingPolicy name="testPolicy2">
        <SelectionCriteria>
            <HourAngleSelector/>
            <SunAvoidanceSelector/>
            <MoonAvoidanceSelector/>
        </SelectionCriteria>
        <Scorers>
            <SciScorer>
                <weight>1</weight>
            </SciScorer>
            <HourAngleScorer>
                <weight>2</weight>
            </HourAngleScorer>
        </Scorers>
    </SchedulingPolicy>

</Policies>
```

The `weight` parameter of the scorers is mandatory. The overall score for each selected SchedBlock is the weighted sum of the individual scorers.

The meaning of each Selector is:

OpacitySelector
: Selects only SchedBlocks for which we can see anything at all at their representative frequency;

MoonAvoidanceSelector
: Selects SchedBlocks which are not too close to the moon;

SunAvoidanceSelector
: Selects SchedBlocks which are not too close to the sun;

HourAngleSelector
: Selects only SchedBlocks which are within 4 hours of their meridian;

SchedBlockGradeSelector
: Selects only SchedBlocks which are of Science Grade A, B or C.

The available Scorers are:

SciScorer
: Considers the Science Grade and Science Ranking assigned to the SchedBlock's Project and assigns high scores to high ranked A grade projects and low scores to low ranked C grade projects;

HourAngleScorer
: Scores highly for SchedBlocks which are at or close to their meridian;

TsysScorer
: Scores highly for SchedBlocks which will have a good Tsys compared to their theoretical best.

Notes:

- When we talk about the position of the SchedBlock, we mean the position of its representative target on the sky. SchedBlocks without a representative target can not be dynamically scheduled;
- If the SchedBlock's project has not been assigned a grade by the APRC, then the SchedBlockGradeSelector and the SciScorer may well produce unexpected results;

- Each individual scorer returns a score from zero to one

The Policy File is read from the location specified in the `scheduling.properties` file. The location is specified in there by the `dsa.policy.file` property.

4.6. Getting New or Updated Projects and SchedBlocks

Behind the scenes, there is a component called the ArchiveUpdater which monitors the ObsProjects, SchedBlocks and Status objects stored in ALMA's databases. If it detects any changes, it will update the Scheduler's records of what Projects &c are available for scheduling. These records are stored in the Scheduling Working Database (or SWDB). Changes to the ObsProjects, SchedBlocks or their Statuses should be reflected in the SWDB within two minutes (though this can be changed, so your installation may have a different time between updates).

The Array Panel does not update automatically in response to SWDB updates (changing displays under the user's feet proved to be a bad thing), but pressing the Update button at the bottom left of the Scheduling Table will update the Project and SchedBlock Tables. Note that this could result in Projects or SchedBlocks disappearing from the tables as a result of them no longer being ready to schedule.

4.7. Creating a Manual Array

The plug-in associated with a manual scheduler is very similar to that for an interactive scheduler The Project and SchedBlock search and selection facilities are the same as for interactive scheduling, except that the Projects found will be those marked as Manual Mode in the ALMA-OT and the SchedBlocks found will be only those in such Projects. Select a SchedBlock in exactly the same way as for interactive scheduling (though note that only one SchedBlock can be selected at once). To Configure the array for the selected SchedBlock, use the pop-up menu on the SchedBlock table (activated by clicking the right mouse button).

In order to use the manual array, CCL commands should be entered in your preferred Python console. Documentation on the CCL is available both within a python session or at the system command line. For the latter, type “pydoc CCL” on a regular terminal where the software is installed and the modules you can choose from will be listed. Then for more detailed information on specific modules type “pydoc CCL.<*mod name*>” (replacing <*mod name*> with the actual module’s name). The same documentation is available in the Python Console by typing “import CCL” and “help(CCL)”.

4.8. Closing a Scheduler plug-in.

To close a scheduler simply close the window by clicking on the red X in the top right hand corner.

NOTE: This does not destroy the array!

4.9. Destroying an array

To destroy an array, either click the 'Destroy Array' button on the Current Executions part of the Array Panel, or use the mouse's right click menu on a selected array in the Existing Arrays tab. See Figure 2.

When a scheduler plug-in is still open, the status of an array will be displayed there. Plug-ins for destroyed arrays will have all their controls disabled.

5. List of Known Issues

1. Restarting the queue while the previous SchedBlock is still active will not work.
2. Currently there is no check against trying to run a SchedBlock marked as Running. Whilst a Running SchedBlock can appear in the Pending queue, it must not be running when it gets selected from the top of the queue. Note that the Scheduler is smart enough to cope with two (or more) consecutive executions of the same SchedBlock – it clears the Running state for the first execution before starting the second.
3. Opening an Array Panel while a SchedBlock is running will fail to show the current execution. Once a new SchedBlock execution starts, the display will work as expected.
4. Certain failures in execution (particularly those which happen before a SchedBlock execution gets properly underway), will not appear in the Completed Executions table.

6. Next official update

R9.0