Task Scheduler

Last Modified on 06/11/2025 3:30 pm CDT

Tool Search: Task Scheduler

The Task Scheduler tool allows users to schedule and manage automated system tasks. For example, the Task Scheduler can be used to regulate and execute the following types of tasks:

- **Built-in system tasks** - maintenance, data exchange, Campus processes
- **Custom maintenance tasks** - user-created tasks (*e.g.*, attendance, grading, user management)
- **Reporting** - including district, state, custom, SQL, etc.

Use the Task Scheduler tool to set the dates and times for running SQL queries or Prism URL calls from the Campus database. Recurring tasks may be set to run hourly, daily, weekly or on monthly rotations. One-time tasks may also be scheduled.

Several areas of the Campus product rely on the *taskScheduler* for functional purposes, including:

- **Batch Queue-** The Batch Queue / Admin Queue tools use the task scheduler to schedule batch reports.

- **Data Warehouse** - The task scheduler schedules and triggers periodic refreshes of the Data Warehouse tools.

- **Point of Sale** - Districts using Point of Sale tools and functionality will see a scheduled task for messaging between the student system and the POS terminal application. This task updates balances to allow the POS terminals and a patron's journal account to reflect the same balance. This task is read-only and does not allow end user modifications; it is for viewing purposes only.

- **Voice and/or Email Messenger** - The task scheduler allows message delivery scheduling and maintenance. Attendance and behavior messages are scheduled and triggered by user-defined settings in the Task Scheduler tool.

Technical Details and Integration

The Task Scheduler tool represents the *taskScheduler* element of the campus.xml file. The *taskScheduler* element executes various system tasks required for optimal use of the Campus

system. Infinite Campus Hosting staff enables the *taskScheduler* element and creates built-in tasks, as necessary for Campus system use.

When the *taskScheduler* is enabled, a district can schedule and manage automated tasks through the Task Scheduler tool.

The *taskScheduler* is enabled when its value is set to "true" within the campus.xml file.

If using an application server pool, only one server in the pool should have *taskScheduler* enabled. Infinite Campus hosting generally enables *taskScheduler* on the first application server in a server pool.

Users who define and schedule tasks in the Task Scheduler need full access rights (**RWAD**) to the tool.

Rights to the Task Scheduler tool should be restricted. Modifying Task Scheduler settings could seriously affect server performance and data processes. Preferably, only a system administrator should have rights to this tool.

Scheduled Task List Editor

The Scheduled Task List editor on the Task Scheduler tool displays a list of existing, active, and inactive tasks.

The task list displays a green or red circle to indicate the task's status.

- A green circle indicates the task is currently active.
- A red circle indicates the task is currently inactive. If a task is red (inactive), the Stop/Pause Execution check box is flagged on the task or the *taskScheduler* is not enabled.

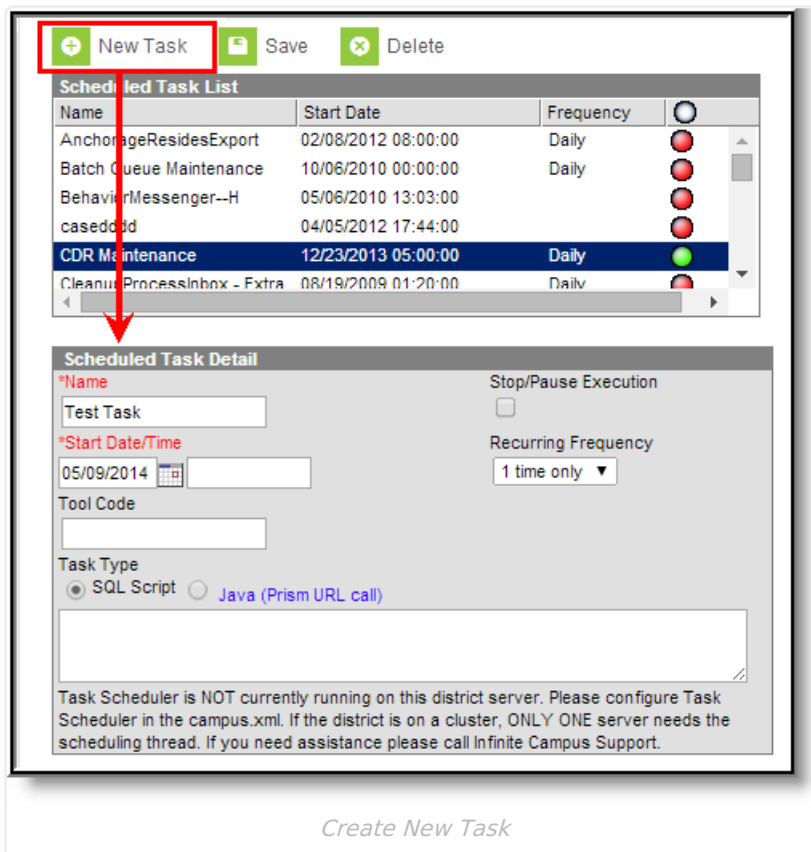At the bottom of the Schedule Task Detail editor, a message indicates whether or not the *taskScheduler* is enabled.

Schedule Task List Editor

Create a New Task

New tasks are only created for custom district/state procedures. Tasks needed for functionality of other Campus tools already exist within Campus.

> The Scheduled Task List editor shows both custom and built-in tasks. Certain built-in tasks are for viewing purposes only and cannot be modified.

Create New Task

1. Select the **New Task** icon from the action bar. The **Scheduled Task Detail** editor displays.
2. Enter information in the required **Name** and **Start Date/Time** fields. Indicate **AM** or **PM** for time field.
3. Select a **Task Type (SQL Script or Java)**.
4. Enter the commands/script in the text box below the **Task Type**.
5. Complete additional non-required fields as necessary.
6. Click the **Save** button when finished.

Field Definitions

Field	Definition
Name	The name of the scheduled task.
Stop/Pause Execution	If flagged, this checkbox indicates the task is not actively occurring. Inactive tasks will not occur, even if the current date is after the Start Date/Time set on the task and a Recurring Frequency has been set.
Start Date/Time	The time of day the task should be triggered for the first time. When a task is initially created, the time (and/or date) should be set at least five minutes in the future to avoid server clock discrepancies. Indicate AM or PM with a time or set it to military time.

Field	Definition
Recurring Frequency	The pattern of repetition the task should follow. Options include: • Hourly • Daily • Weekly • Monthly • 1 Time Only (indicates task will not repeat)
Tool Code	This field is only for the auto-populated tool code of centralized printing tasks for reporting purposes. The user should not enter or modify anything in this field.
Task Type	Indicates the type of task, either SQL Script or Java (Prism URL call).

Batch Queue Maintenance Task

The Batch Queue Maintenance task is an automated task that deletes old reports from the Batch Queue / Admin Queue tools. If this task applies to a district or state, it will appear in the Task Scheduler.

Batch Queue Maintenance Task

This task will clear completed reports from the Batch Queue / Admin Queue tools after they have exceeded the maximum number of days they are allowed to remain. The maximum number of

days is set in the **Batch Queue Expiration Days** field of the System Preferences.

When a report is deleted from the Batch Queue / Admin Queue tools by this task, the details of its generation still appear on the Batch Queue / Admin Queue tools for tracking purposes (though the report itself was deleted).

The Batch Queue Maintenance task's default setting is to execute every night at midnight. However, execution settings for this task may be modified as necessary. This task can be deleted from the Scheduled Task List; however, it will be automatically rebuilt the next time the application server is restarted. If this task should be disabled, the **Stop/Pause Execution** checkbox should be marked.

View Reports Scheduled for Deletion

Each report displayed in the Batch Queue / Admin Queue tools will display its specific deletion date in the **Expires After** field of the Batch Queue Detail editor. This is the date the report will be removed from the queue, as long as the Batch Queue Maintenance task is executed. If the maintenance task is not executed on that date, the report will be deleted the next time the maintenance task runs.

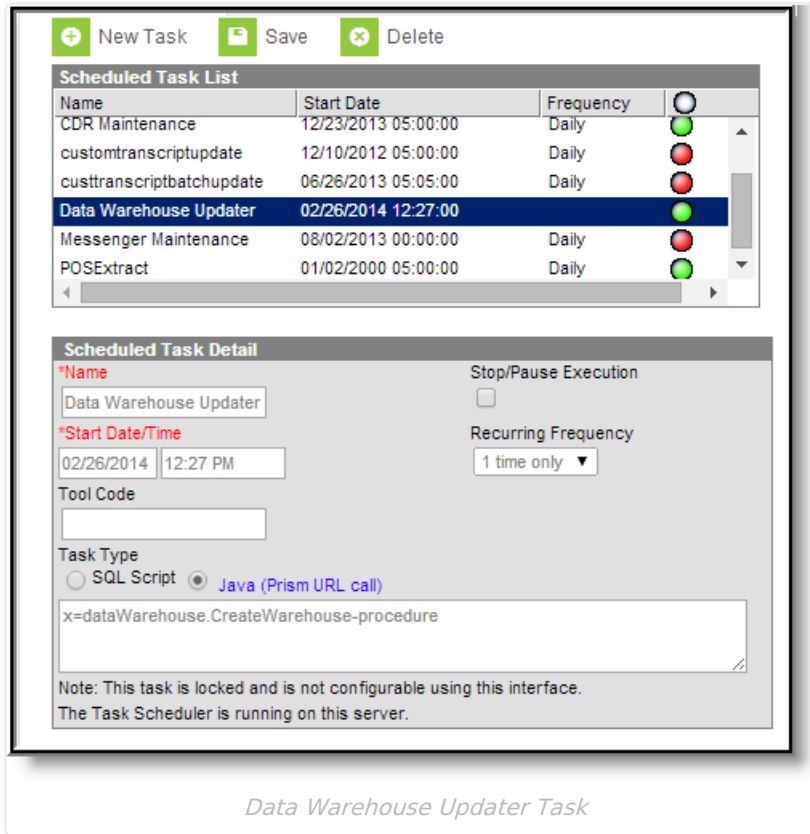*Example of a Report Scheduled for Deletion*

The Batch Queue Maintenance task will NOT delete any report that has been manually excluded from the deletion process. These reports will have the **Never Expires** checkbox marked and the **Expires Date** crossed out within the Batch Queue Detail editor of the Batch Queue / Admin Queue tool.

When a report is deleted from the Batch Queue / Admin Queue tools by this task, the details of its generation will still appear on the Batch Queue History tab for tracking purposes (though the report itself was deleted).

Data Warehouse Tasks

A scheduled task called the Data Warehouse Updater Task may exist within the Task Scheduler. It assists data backup on behalf of the Data Warehouse Settings tools.

Data warehousing users will see a read-only **Data Warehouse Updater** scheduled task. When applicable, Infinite Campus staff will insert this task for a customer.

Data Warehouse Updater Task

Custom District Tasks

The following examples are custom district tasks that may be useful for Campus system administrators.

> Infinite Campus highly recommends testing these custom tasks in a non-production site before implementing them on a live site.

This section includes example queries for the following custom district tasks:

- Update Unknown Absences to Unexcused Absences
- Update Custom Tabs with Custom District Data

Update Unknown Absences to Unexcused Absences

This query can be used to update all existing unknown absences to unexcused absences.

The district must change the codes at the end of the SET/CASE statement to the codes appropriate to the district.

```
UPDATE a
SET a.excuseid = case COALESCE(x.status, a.status)
WHEN 'A' THEN (SELECT AE.excuseid FROM AttendanceExcuse AE WHERE a.calendarid = ae.calendarid AND ae.[code]
 = 'A')
WHEN 'T' THEN (SELECT AE.excuseid FROM AttendanceExcuse AE WHERE a.calendarid = ae.calendarid AND ae.[code]
 = 'T')
END
FROM dbo.Attendance a
LEFT OUTER JOIN dbo.AttendanceExcuse x ON x.excuseID = a.excuseID AND x.calendarID = a.calendarID
INNER JOIN calendar c ON c.calendarid = a.calendarid
INNER JOIN schoolyear sy on sy.endyear = c.endyear
WHERE (a.excuse is null and x.excuse is null) and a.crosssitesection = 0
AND sy.active=1
```

Update Custom Tabs with Custom District Data

This query is an example of updating a custom tab with data from a custom district table (generally,this data is from another software application and has been inserted into the Custom District table by matching personIDs).

```
INSERT INTO customstudent (personid,attributeid,[value],[date])
SELECT personID,287,1,getdate()
FROM MyCustomStudentTable mcst
WHERE mcst.specificValue = 'M'
```