

# Task Scheduler

Last Modified on 06/08/2026 9:56 am 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 a monthly rotation. One-time tasks may also be scheduled.

Task Name	Start Date	Start Time	Frequency	Active/Stopped	Edit
ac-pref	05/05/2025	01:51 PM		ACTIVE	Edit
Alternative Program Calendar Maintenance	07/10/2018	05:25 PM	Daily	STOPPED	Edit
Alternative Program Student Maintenance	11/28/2018	09:35 PM	Daily	STOPPED	Edit
AZ SAIS Automated Download	08/31/2012	12:00 AM	hourly	STOPPED	Edit
AZ SAIS Automated Upload	08/31/2012	12:00 PM	daily	STOPPED	Edit
Batch Queue Maintenance	06/25/2012	12:00 AM	Daily	STOPPED	Edit

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.

Many of the columns in the task list can be filtered and/or searched to help locate a specific

task(s).

Task Name	Start Date	Start Time	Frequency	Active/Stopped	Edit
<input type="text"/>	month/day... <input type="text"/>		<input type="text"/>	(All) <input type="text"/>	
AZ SAIS Automated Upload	08/31/2012	12:00 PM	daily	STOPPED	Edit
Batch Queue Maintenance	06/25/2012	12:00 AM	Daily	STOPPED	Edit

## 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.

## Create a New Task

New tasks are only created for custom district/state procedures. Tasks needed for the 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.

Task Scheduler System Settings > System Processes > Task Scheduler

Task Name	Start Date	Start Time	Frequency	Active/Stopped	Edit
	month/day...			(All)	
	09/16/2025	08:00 AM		ACTIVE	Edit
***	09/11/2025	01:46 PM		ACTIVE	Edit
ac-pref	05/05/2025	01:51 PM		ACTIVE	Edit
Alternative Program Calendar Maintenance	07/10/2018	05:25 PM	Daily	STOPPED	Edit
Alternative Program Student Maintenance	11/28/2018	09:35 PM	Daily	STOPPED	Edit
AZ SAIS Automated Download	08/31/2012	12:00 AM	hourly	STOPPED	Edit
AZ SAIS Automated Upload	08/31/2012	12:00 PM	daily	STOPPED	Edit

**New**

**Scheduled Task Detail**

**Name: (Required)**

**Start Date/Time (Required)**

**Tool Code:**

**Task Type: (Required)**

**Command to run:**

**Stop/Pause Execution**

**Recurring Frequency:**

**Save** **Cancel**

1. Select the **New** 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
<b>Name</b>	The name of the scheduled task.

Field	Definition
<b>Stop/Pause Execution</b>	<p>If flagged, this checkbox indicates the task is not actively occurring.</p> <p>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.</p>
<b>Start Date/Time</b>	<p>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.</p> <p>Indicate AM or PM with a time or set it to military time.</p>
<b>Recurring Frequency</b>	<p>The pattern of repetition the task should follow. Options include:</p> <ul style="list-style-type: none"> <li>• Hourly</li> <li>• Daily</li> <li>• Weekly</li> <li>• Monthly</li> <li>• 1 Time Only (indicates task will not repeat)</li> </ul>
<b>Tool Code</b>	<p>This field is only for the auto-populated tool code of centralized printing tasks for reporting purposes.</p> <p>The user should not enter or modify anything in this field.</p>
<b>Task Type</b>	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.

Task Name	Start Date	Start Time	Frequency	Active/Stopped	Edit
<input type="text"/> ▼	month/day... 📅 ▼		<input type="text"/> ▼	(All) ▼	
AZ SAIS Automated Upload	08/31/2012	12:00 PM	daily	STOPPED	Edit
Batch Queue Maintenance	06/25/2012	12:00 AM	Daily	STOPPED	Edit
bothRender	03/25/2025	04:00 PM		ACTIVE	Edit
Calendar Rollover	06/12/2020	09:00 PM		STOPPED	Edit
centralReport	03/25/2025	03:55 PM		ACTIVE	Edit
Clean Old Data	12/27/2024	02:02 PM		ACTIVE	Edit

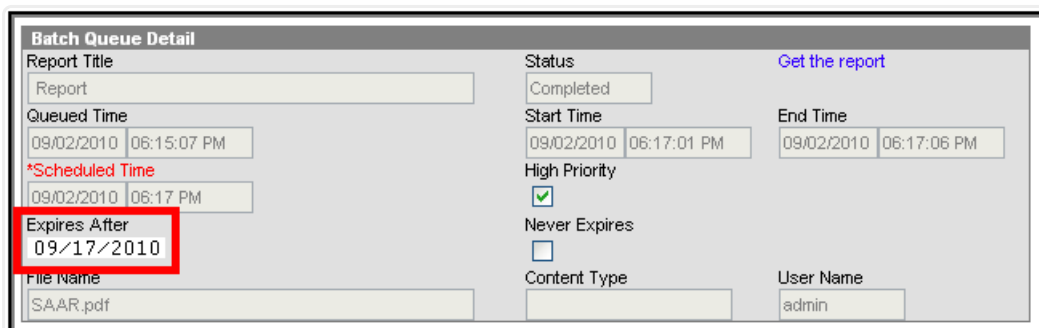
This task will clear completed reports from the [Batch Queue](#) / [Admin Queue](#) tools after they have exceeded the maximum number of days they are allowed to remain. The maximum number of days is set in the **Batch Queue Expiration Days** field of the [System Preferences](#).

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.



Batch Queue Detail	
Report Title	Status <a href="#">Get the report</a>
Report	Completed
Queued Time	Start Time
09/02/2010 06:15:07 PM	09/02/2010 06:17:01 PM
*Scheduled Time	End Time
09/02/2010 06:17 PM	09/02/2010 06:17:06 PM
Expires After	High Priority
09/17/2010	<input checked="" type="checkbox"/>
File Name	Never Expires
SAAR.pdf	<input type="checkbox"/>
	Content Type
	User Name
	admin

*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](#) 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.

The screenshot shows a web-based Task Scheduler interface. At the top, there are buttons for 'New Task', 'Save', and 'Delete'. Below is a 'Scheduled Task List' table:

Name	Start Date	Frequency	Status
CDR Maintenance	12/23/2013 05:00:00	Daily	Green
customtranscriptupdate	12/10/2012 05:00:00	Daily	Red
custtranscriptbatchupdate	06/26/2013 05:05:00	Daily	Red
<b>Data Warehouse Updater</b>	<b>02/26/2014 12:27:00</b>		<b>Green</b>
Messenger Maintenance	08/02/2013 00:00:00	Daily	Red
POSExtract	01/02/2000 05:00:00	Daily	Green

Below the list is the 'Scheduled Task Detail' section for the 'Data Warehouse Updater' task:

- Name:** Data Warehouse Updater
- Stop/Pause Execution:**
- Start Date/Time:** 02/26/2014 12:27 PM
- Recurring Frequency:** 1 time only
- Tool Code:** [Empty field]
- Task Type:**  SQL Script  Java (Prism URL call)
- Task Code:** x=dataWarehouse.CreateWarehouse-procedure

Note: This task is locked and is not configurable using this interface. The Task Scheduler is running on this server.

*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 Pages with Custom District Data](#)

## Update Unknown Absences to Unexcused Absences

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

Note: This query was updated to account for cross-site enrollment, where absence information is matched between two different locations without the need for them to be manually updated.

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 Page with Custom District Data

This query is an example of updating a custom page 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'
```