



# GroveStreams Website Guide

---

## Table of Contents

Table of Contents .....	1
User Profile.....	5
Editing Your Profile.....	5
Changing Your Password .....	5
Forgotten Password .....	5
Managing Your Account .....	5
Subscription .....	6
Usage.....	6
Invoices .....	6
Transactions .....	6
Branding Groups, White Labeling, and OEMing .....	6
Creating a New Brand .....	7
Editing a Branding Group .....	7
General.....	7
Home Page .....	8
Custom Home Page .....	8
Generic Home Page.....	8

Links .....	9
Email Addresses .....	9
Issues Email Address.....	9
No-Reply Email Address.....	9
Applying to Organizations .....	9
Users .....	10
No-Reply Email Address.....	10
Changing the Home Page and Start Page Logo .....	10
Changing the grovestreams.com URL .....	10
Organizations .....	11
Who Can See Your Organization?.....	11
Adding Users .....	12
User Rights .....	12
Making Your Organization Public .....	13
Organization Blueprints.....	14
How to Create a Blueprint.....	14
Blueprint Use Case .....	15
Blueprint Tips and Best Practices.....	15
System Blueprints.....	16
Component and Streams .....	16
Virtual Streams, Other Calculations and Real-Time.....	17
Phase I Calculations (real-time): .....	18
Phase II Calculations (near real-time): .....	20
Derived Streams (Phase II Calculations).....	20
System Reliability – The Process Queue.....	23
Jobs .....	24
Job Example .....	24
Notifications.....	24

System Notifications .....	24
User Notifications .....	25
Jobs Notifications.....	25
File Imports .....	25
GS Stream Format .....	27
JSON Feed Import Format.....	27
Plain Delimited Import Format .....	30
Backup and Restore.....	33
Searching for Components.....	34
Stream Groups .....	34
Stream Group Inclusion Algorithm .....	38
Runnables, Processes, and the Scheduler.....	38
Queries.....	40
Query Schedules .....	40
Query Security.....	40
API Keys.....	40
api_key Examples.....	42
API Editor .....	44
Viewing an API Secret Key.....	45
Time Zone Support.....	46
Dashboards .....	47
Adding Components and Streams to a Widget.....	48
Changing a Widget's Settings.....	48
Chart Widget Settings .....	48
Grids Widget Settings.....	50
Maps Widget Settings .....	50
RSS Widget Settings .....	51
Stream Feed Form Widget .....	51
Embeddable Graphs and Grids .....	52

Export to a Spreadsheet.....	53
Component View .....	54
Stream View .....	54
Export Stream Feed Range to a Spreadsheet.....	55
Subscribing to a Stream's RSS Feed .....	55
Quick View .....	55
Active Events Viewer.....	55
Custom Application Development .....	56
Mobile Application.....	56
Component Studio .....	57

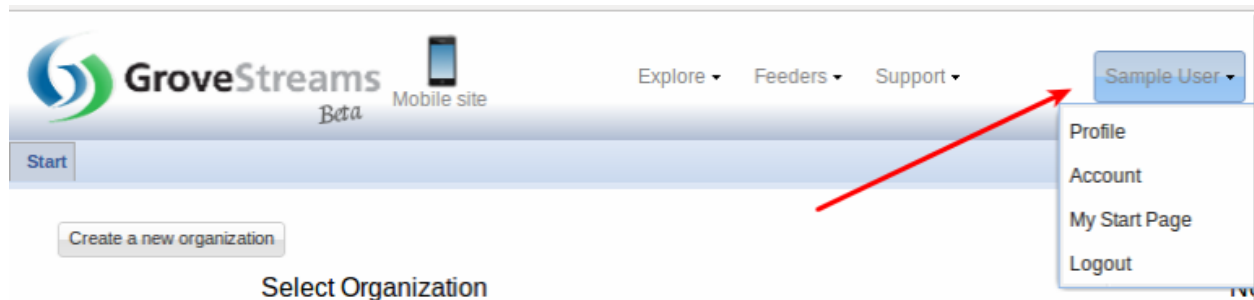
## User Profile

A user profile contains information such as the user's display name and email address. Most profile information is optional with the exception of the email address, password and time zone.

## Editing Your Profile

To edit the user profile profile:

1. Sign-in



2. Click on your name in the top right corner and choose the **Profile**

## Changing Your Password

1. Sign-in
2. Click on your name in the top right corner and choose **Profile**
3. Click the **Change Password** button

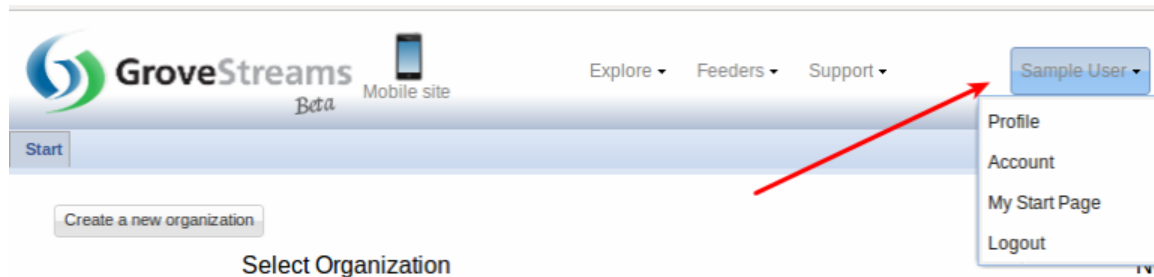
## Forgotten Password

1. Navigate to [www.grovestreams.com](http://www.grovestreams.com)
2. Enter the email address you used to sign-up with GroveStreams
3. Click the **Forgot Password** link. GroveStreams will email you a new password. You can reset/change the new password after you use it to log into GroveStreams.

## Managing Your Account

To manage your account:

1. Sign-in
2. Click on your name in the top right corner and choose the **Account**



## Subscription

The subscription tab allows you to setup your payment method. GroveStreams does not store your credit card information. GroveStreams uses one of the largest providers of credit card-based processing services, Authorize.Net, to manage and store sensitive billing information.

Authorize.Net allows for shipping information to be entered but is not required.

## Usage

The usage tab allows you to view the metrics that are used for billing calculations.

## Invoices

The invoices tab lists all previous invoices and your ongoing invoice. Ongoing invoices are processed daily so that you can view your current month's costs as they accrue each day.

Invoices are billed monthly.

## Transactions

The transactions tab displays all credit card charges and refunds that have been applied to your account.

## Branding Groups, White Labeling, and OEMing

GroveStreams supports rebranding, otherwise known as white labeling or OEMing.

Rebranding can be custom configured by creating a Branding Group and associating one or more organizations with it.

A Branding Group allows GroveStreams to be branded as another company. Rebranding allows for:

- The Replacing of home page text and the name 'GroveStreams'
- Custom Logos to be displayed
- Custom 'from', 'info', and 'issue' emails
- Custom URL
- The removal of the Explore, Feeders, and Support menu items located at the top of non-Studio pages
- The removal of the Explore, FAQ, Developers, and Contact Us links located at the bottom of non-Studio pages

## Creating a New Brand

1. Sign-in to GroveStreams
2. Click on your user name located at the top right and choose **Account**.
3. Select the Branding Groups tab and then click the **New** button

## Editing a Branding Group

### General

- **Brand Name:** Give your Brand a name. This name will replace "GroveStreams" within pages and emails.
- **Brand Subdomain:** Enter a unique subdomain for your brand. The subdomain will be prepended to grovestreams.com (*your\_subdomain.grovestreams.com*). GroveStreams applies the branding settings when it is used within the browser.
- **Rebrand URL Completely:** Select this option if you want to replace the grovestreams.com part of the url with a custom domain.
- **Brand Domain:** Your custom domain. Leave off the protocol (i.e. http://). The domain entered here is required to be registered with a domain provider such as godaddy.com and the DNS record modified so that the new domain points to grovestreams.com. Examples – assume you're subdomain is "samplecompany", possible domains are:

- samplecompany.com
- [www.samplecompany.com](http://www.samplecompany.com)
- samplecompany.acme.com
- www.samplecompany.acme.com

## Home Page

GroveStreams gives you the choice of creating your own custom home page or using a generic home page.

### *Custom Home Page*

Creating a custom home page requires programming a custom web page that allows the user to:

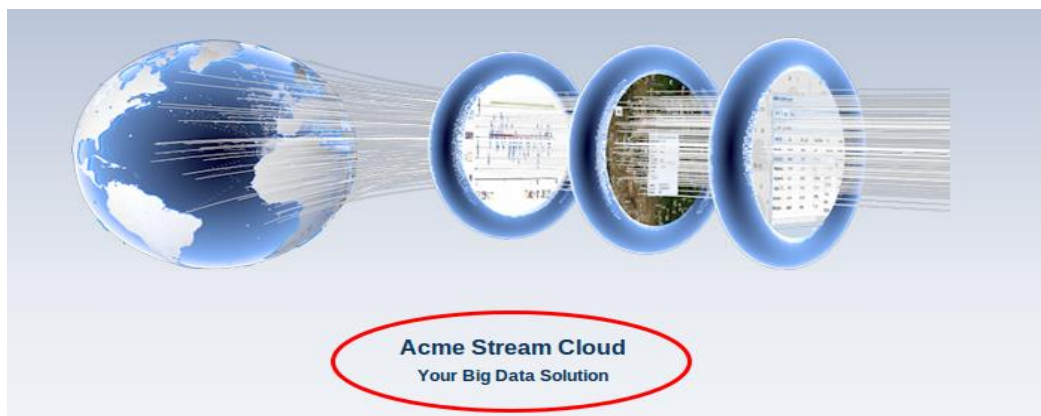
- Create an account
- Verify Captcha entries and confirm their email address
- Log into GroveStreams

See the GroveStreams [User API resource](#) for more information about programming your own web page for managing user accounts.

Custom Home Page URL: Enter your custom home page URL here. Leave this blank if you are not programming your own home page.

### **Generic Home Page**

Optionally change the text under the generic sign-in page graphic:





Your home page link is displayed at the bottom of the General tab.

Home Page Link: Use this URL to send your users to your branded pages.

## **Links**

The link tab allows you to provide links to pages that have been replaced. Optionally override any or all of these links.

**Important:** If you override the GroveStreams Terms or Privacy links, then your own Terms and Privacy pages must contain links back to the GroveStreams Terms and Privacy pages.

## **Email Addresses**

Optionally change email addresses.

### *Issues Email Address*

Issues reported by users from within the GroveStreams website are forwarded as emails to [info@grovestreams.com](mailto:info@grovestreams.com). Enter an alternate email address below if you would like to have the issues sent there.

### *No-Reply Email Address*

GroveStreams sends emails during the user sign-up, email reset, event notifications and other similar processes. GroveStreams uses [no-reply@grovestreams.com](mailto:no-reply@grovestreams.com) as the “From” email address.

GroveStreams uses Amazon SES services to manage its email infrastructure. To avoid spammers, Amazon SES requires all “From” email addresses to be registered with it, inside the GroveStreams SES Account.

You must contact GroveStreams via email at [info@grovestreams.com](mailto:info@grovestreams.com) and request that your ‘From’ email address be added to the GroveStreams safe email address list. GroveStreams will register your email address with Amazon SES. Amazon SES will send a confirmation email to the email address you are registering. Your email address must be able to receive new emails and you must be able to read them because you will need to open the confirmation email from Amazon and follow the directions within the email to confirm your address.

## **Applying to Organizations**

Add organizations that you would like to fall under this Branding Group. You must be an owner or an administrator of any organization that you add.

You can change an organization’s logo here.

## Users

This tab has settings to determine what to do when a user signs up while under a Branding Group. The Branding Group can be configured to automatically add users to an existing organization or create an organization for the user when they sign up.

- Allow Users to Create Organization: If you would like users running under this brand to create organizations, then select the **Allow Users to Create Organizations** checkbox
- When Users Sign Up:
  - Do Nothing: Allow users to sign up, but don't create or add them automatically to any organizations.
  - Auto-Assign to a New Organization: A new organization will be created using the uploaded Blueprint and the newly registered user will be added to the organization. Select who will be the Owner and Payer of the organization. If the newly registered user is not the Owner of the new organization, then the organization will not be part of the Branding Group and the user will not see the organization on their start page unless they navigate to their start page without the branding subdomain as part of the URL.
  - Auto-Assign to an Existing Organization: Automatically assign newly registered users to a selected organization. Users will automatically be assigned to the "Users" group with rights.

## *No-Reply Email Address*

### Changing the Home Page and Start Page Logo

You may change the home page and start page **logo** after you submit your new Branding Group.

Select the new Branding Group in the list and click **Change Brand Logo**

### Changing the grovestreams.com URL

This section describes how to brand the URL to a different domain other than *your\_subdomain.grovestreams.com* to *your\_subdomain.your\_domain.com*

Changing the URL that users see in their browser address field requires modifying a new or existing domain zone file CNAME. New domains can be registered from providers such as GoDaddy.com.

Here is a quick example of how to add a CNAME to a Go Daddy account:

- Let's assume you'd like your URL to be devicecloud.acme.com
- First, follow the directions above to create a branding group with a subdomain called **devicecloud** which results in your branding group URL being <http://devicecloud.grovestreams.com>.
- Enter http://devicecloud.acme.com in the **Custom Home Page URL** field on the **Home Page** tab
- Log in to your Go Daddy account, locate the domain name you want to use, and select **Launch**
- Locate the CNAME (Alias) table, and click **Quick Add**
- For the Host name enter the subdomain you created above for the branding group. For example "**devicecloud**" will be your Host name for devicecloud.acme.com
- The **Points to** must be the URL of your branding group. In this example it will be devicecloud.grovestreams.com (do not add www. Or "http://").
- For **TTL** (time to live) select **1 Hour**
- Click **Save Zone File**

Note: Changes usually take 1 hour. However, it may take up to 48 hours for these changes to take effect.

## Organizations

An organization is a group of components, dashboards, maps, users and other entities. It can remain private or it can give anonymous guests, with assigned rights, access. An Organization can be a home, business, a government agency, a health care provider, a utility, or any other type of entity.

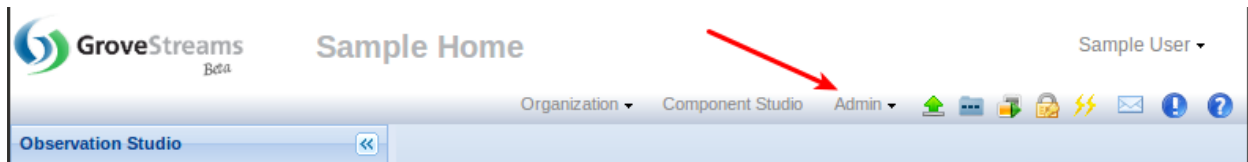
After a user creates a free user account in GroveStreams, they can be invited to many other GroveStreams organizations or they can create their own organizations.

### Who Can See Your Organization?

Only you, the owner, has access to a new organization until more users are added or the organization is configured as a 'Public' organization.

## Adding Users

Enter your organization and select **Admin - Organization Users** from the menu to give more users access to your organization. Users must have a GroveStreams user account prior to adding them to an organization.



## User Rights

Each user can has certain rights that are granted by an organization owner or administrator. Rights are set during the adding of a user or by selecting a user and clicking the **Edit** button.

Is Owner: Users assigned as owners have all rights and are responsible for usage payments to Grove Streams, LLC.

Is Administrator: Users assigned as administrators have all rights except the ability to:

- Delete the organization
- Change the ownership of the organization

There are three types of rights:

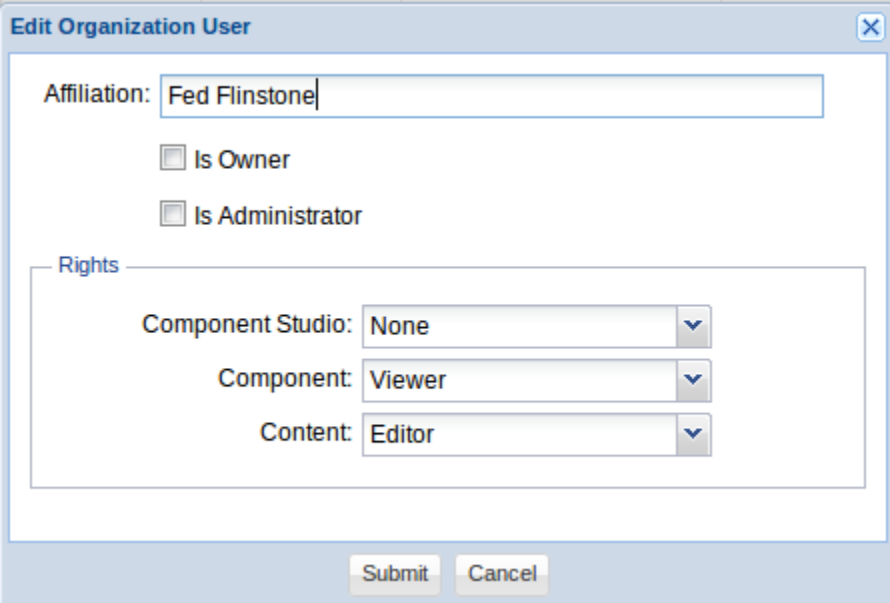
1. None: The user has no rights to this area.
2. Viewer: The user has view only rights to this area.
3. Editor: The user has creating, editing and deleting rights to this area.

Rights can be assigned to each of the three main areas of GroveStreams:

1. Component Studio: Includes
  - Event Viewer
  - System and Job notifications
  - Import Profiles, Stream Groups, Runnables
2. Observation Studio – Components:
  - Components

- Streams
3. Observation Studio – Content:
- Maps
  - Dashboards

Only Owners and Administrators have the ability to manage API Keys.



**Edit Organization User**

Affiliation: Fed Flinstone

Is Owner

Is Administrator

**Rights**

Component Studio: None

Component: Viewer

Content: Editor

Submit Cancel

### Making Your Organization Public

Organizations can be configured to be 'Public'. Organizations configured as Public organizations, give users access to GroveStreams without requiring them to register with GroveStreams. These users are referred to as Guests throughout this document.

Enter an organization and select **Admin - Organization Public/Private Settings** from the menu to allow Guest users access to an organization.

Select the rights for Guest users for the organization.

Anonymous user sessions will never timeout.

## Organization Blueprints

Organizations can be created with an organization blueprint. Blueprints allow users to get up and running quickly without having to know how to model their environment. They also can be used to copy organizations without copying stream data.

### How to Create a Blueprint

1. Create a GroveStreams organization
2. Model your environment within your organization by creating all of the entities you would like to include in your blueprint such as:
  - a. Component Templates
  - b. Components
  - c. Units
  - d. Time Filters
  - e. Dashboards
  - f. Maps
3. After you have created your organization and tested it, you can create the blueprint by choosing the menu item **Admin - Organization Create Blueprint**
  - a. The blueprint is exported as a text file and is downloaded to your local computer
4. The blueprint can be used when creating new organizations by selecting **Advanced** and choosing **Create with a custom blueprint**. Select the custom blueprint file and then click **Create Organization**

Every organization entity will be included within a blueprint with the exceptions of:

- Stream Data, with the exception of default Point Stream data.
- Organization Administration Data
  - Custom Logo
  - Organization Users
  - Public/Private Settings

- Organization name and address
- Notification Limits

### **Blueprint Use Case**

Let's assume you are a Smart Plug distributor that sells smart plugs and offers a cloud website service (GroveStreams) to view usage information. You would like to have new smart plugs automatically appear in your customer's new GroveStreams organization. Your users are not highly technical and cannot easily model items within GroveStreams.

You can create an organization that works with your smart plugs, complete with Component Templates, Dashboards, Maps, Derived Streams, Events, Notifications, and much more.

You can then export that organization as a blueprint. The blueprint can be used during the creation of each of your user's organizations to get them up and running quickly without any configuration or modeling being performed by them.

Each of your customer's smart plugs will automatically appear in their GroveStreams organization and, if the blueprint was modeled correctly, automatically appear within predefined dashboards – all without your users having to model or configure anything.

### **Blueprint Tips and Best Practices**

- Model your organization so that components can auto-register themselves and automatically appear in your dashboards. For example:
  - Create component templates and have your stream feeds reference those component template IDs
  - New internal UIDs are assigned to every entity when they are created using a blueprint. Avoid relying on internal UIDs during feed uploads. Use component IDs, Stream IDs, and other IDs for other associated metadata within your stream feed (i.e. associate your stream with a unit ID instead of a unit UID). This will allow your stream feeds to be uploaded to any organization.
  - Reference Stream Groups instead of specific streams within your dashboards and maps. This will allow newly registered components and streams to automatically appear within your dashboards and maps

- Setup a Scheduled Runnable that will periodically update your stream groups
- Event Notifications: Your new organization will not contain any users with the exception of you, the owner. Any users that were associated with email or text notifications will no longer be associated with them unless their email addresses and phone numbers were explicitly entered for each notification.
- Testing your blueprint
  - Create a new organization with your new blueprint
  - Point your devices to your new organization. This is typically done by changing the organization and API key tokens within your device that is uploading into GroveStreams.
  - Test that your components auto-register and that you can view them in your dashboards and maps within the organization that was created with the blueprint.

### System Blueprints

GroveStreams provides some system blueprints that can be chosen during organization creation to assist with the auto-creation of some entities such as common units, cycles and rollup-calendars.

### Component and Streams

A Stream represents a collection of one or more data points. Each data point can be associated with a single time stamp or a time range. Timestamps are accurate to the millisecond.

There are three types of streams:

- Regular Streams: Each data point is associated with a datetime. Datetimes can be random. There can be tens of millions of data points for each stream. This is the most common stream type.
- Interval Streams: Each data point is associated with a start datetime (inclusive) and an end datetime (exclusive). There can be tens of millions of interval data points for each stream.
- Point Streams: There is only one data point. This value can change frequently or infrequently, but only one data point is stored.

Stream data types can be numbers, dates and times, text, booleans (yes/no or true/false), longitude, latitude or elevation. Each Stream can have time filters, constraints, gap filling methods, a unit and other



items applied to its feed data as it is uploaded. Stream data can be uploaded via the GroveStreams' API or a stream can be defined as a derived stream. It is not required for a component to upload all of its streams at the same time with the exception of mobile components. Mobile components are required to upload their longitude and latitude streams at the same time.

A component is a "thing" that is also a container for a group of streams and events that share a similar location. For example, a component might be a sensor that monitors temperature and humidity. Temperature would be one stream and humidity would be another stream. If the sensor were mobile, its component could also have latitude and longitude streams that record the location of the sensor component over time.

A component is an abstract concept that represents physical "things" or non-physical "things". Examples of components:

- Physical "things":
  - Utility Meters
  - Smart Plugs
  - Sensors (Temperature, Accelerometer, etc.)
  - Motion Detectors
  - Cardiac Telemetry Devices
  - Cellular Phones
  - Vehicles, Planes, Boats
- Nonphysical "things"
  - A computer program that tracks computer memory and CPU in a server farm
  - A computer program that polls a relational database for information
  - A computer program that uploads financial information such as sales figures, expenses, stock quotes, etc.

## **Virtual Streams, Other Calculations and Real-Time**

The term real-time is thrown around a lot within the Internet of Things (IoT), M2M, and device communities. It can mean different things to different systems. It's important that GroveStreams users and organizations understand what we define as real-time or near real-time.

We consider any processing that takes place during a component feed upload to be done in "real-time". We also consider any calculated results that are available immediately after a feed has successfully uploaded as "available in real-time".

So what exactly does GroveStreams do in real-time?

GroveStreams has developed a patent pending two phase calculation engine. Phase I processes component data as it is uploaded within the same upload transaction. Phase II calculations are near real-time calculations. Phase II calculations run periodically, usually every few seconds, but it depends on the number of calculations and the complexity of the calculations. Phase I and Phase II calculations don't just calculate a single result, they calculate a range of results over a period of time intervals. That is, the result of Phase I and II calculations are GroveStreams' streams. And these results can be used just like any other GroveStreams stream. They can be graphed, rolled-up, have events setup on them, and be used for other phase II calculations.

To clarify: let's dissect a Phase I calculation so that you know exactly what is done in real-time.

### **Phase I Calculations (real-time):**

A lot of processing takes place when a component feed is uploaded into GroveStreams. Let's walk through the processes as they occur so that you understand what is done in real-time. This example will be for a one second interval stream associated with a rollup calendar:

1. An HTTP feed PUT call is made to upload 10 one second intervals for a component stream
2. Each value is placed into its correct interval time slot using the time passed in with the feed. The interval start date is inclusive. The interval end date is exclusive.
3. The optional time filter constraint is applied for every upcoming process and rollup calculation. Samples that are not included with the Time Filter will be ignored for all calculations.
4. Optional Min/Max or Floor/Ceiling constraints are applied to each interval.
5. Optional Gap filling logic is applied.
6. Event detection takes place. Event action packages, such as notifications or http calls, are processed.

The above steps are great; they organize and process the stream data the way you have configured it to do so in real-time. But, here's where things get really interesting. A group of statistical virtual streams become available in real-time if a rollup calendar has been associated with the stream. These virtual streams can be used throughout GroveStreams just like any other stream.

What exactly is a virtual stream? Let's go through an example. Suppose you've associated the one second stream above with a rollup calendar defined in this manner: Rollup one second intervals into five

minute intervals. Rollup the five minute intervals into hour intervals; hours into days; days into months; months into quarters; quarters into years.

Once a stream has been associated with a rollup calendar like the one described above, at least 66 statistical virtual streams are available in real-time. Virtual streams like:

- 5Minute Averages
- 5Minute Sums
- 5Minute Maximums
- 5Minute Maximum date and time occurrences
- 5Minute Minimums
- 5Minute Minimum date and time occurrences
- 5Minute Gap Counts
- 5Minute Nongap Intvls
- 5Minute Interval Counts
- 5Minute First interval values
- 5Minute Last interval values

Each of these statistics are available in real-time for each rollup calendar level immediately after the upload transaction has completed. All that the user, you, had to do was associate a rollup calendar with a stream. You didn't have to write a complex script that includes time variables. So what can you do with virtual streams? A lot! Here are some examples of what's available in real-time:

1. You can graph 15 minute temperatures from a temperature sensor along with its hourly, daily, and annual highs and lows in the same graph.
2. You can setup an event on a device uploading 1 second water usage that will send you a notification if the total water usage for the current year exceeds a certain amount. Do you know how many seconds there are in a year? Over 30 million! Don't worry, you can still have access to that statistic and the other annual statistics immediately after the upload has completed.
3. In combination with a time filter, energy and other commodity folks, can extrapolate time-of-use billing determinants in real-time.

Very nice! But it gets better. GroveStreams supports batch GETs for component feed data. You can assemble batch calls to get a range of stream values, including virtual stream values from any rollup cycle, from components across your organization in one call. Batch calls are very fast. They're used throughout the GroveStreams web user interface such as within dashboards.

## Phase II Calculations (near real-time):

Some calculations are more complex and can involve data from other components across your organization. These types of calculations could be too lengthy to perform during a feed upload so they are performed periodically after feeds are uploaded. GroveStreams' phase II calculations are component derived stream calculations. Most SCADA applications have this concept. Their derived streams are sometimes referred to as calculated channels. Derived streams are interval streams. Each resulting interval is calculated using a user defined expression. Derived streams are just like any other GroveStreams stream.

Derived streams can be derived from any other organization component stream (real, virtual or derived). When using a virtual stream, the rollup cycle is selected along with the statistical function for that cycle and an optional interval offset. The combination of these settings makes our derivation engine very flexible. It allows for near real-time calculations such as rolling averages and comparing values from streams with different intervals sizes across time ranges.

Those of you working within the energy industry can now see the advantages of the two phase calculation engine. Phase I can extract the time-of-use billing determinants in real-time and Phase II can use those results to calculate block costs for a user defined billing cycle whether the cycle is every minute or every month or both!

For the non-energy crowd, the GroveStreams calculation logic crosses industries and can be used for any industry wherever real-time results are needed including industries with a lot of devices such as the health care industry.

Phase I & II calculations are configured as part of the component definition. Right click on a component within the component tree on the left of Observation Studio and choose **Edit**. More information regarding component and stream modeling can be found in the GroveStreams Modeling Guide.

## Derived Streams (Phase II Calculations)

Streams can be derived from other sources besides from direct feed uploads.

### From RSS Feed:

Streams can be derived from internal or external RSS feeds. GroveStreams allows for a simple filter to be applied to RSS feed results so that certain values can be extracted from an RSS feed.

RSS feeds are a popular way of bringing external data into GroveStreams for mashing up with other streams. Many RSS feeds are available on the Internet including:

- Weather data
- Stock quotes
- Currency Exchange Rates
- External, 3rd party sensor data

GroveStreams automatically updates RSS derived streams hourly.

The GroveStreams sandbox organization provides an example of an RSS derived stream.

#### From Aggregation:

Streams can be derived from aggregating anywhere from two to thousands of other streams. Calculate statistics such as the maximums, minimums, sums, averages, and gap counts for each interval in the aggregation time range. Aggregation is started either manually or it can be scheduled to run periodically since it can be a lengthy process.

#### From Expression:

A Derived expression stream is calculated using a user defined expression that may include other streams within an organization. Calculations occur periodically (usually every few seconds) and only occur when all dependents have data available for the time period being calculated. Different stream roll-up cycles and interval offsets can be used inside a derivation expression.

Derived streams are just like other streams except that data is derived by the GroveStreams' derivation engine. They can be graphed and monitored in dashboards. Derived streams can be derived from other derived streams.

Expression variables are streams. If a variable is an interval stream, then the user can choose the interval offset for that variable. For example, if an hourly interval with a time span of 2:00 pm to 3:00 pm is being calculated, a dependent variables' offset can be set to -1 and that variables' data for time span 1:00 pm to 2:00 pm will be used in the calculation. Offsets are useful for things like calculating rolling averages or comparing monthly costs.

Expressions can contain any stream data types, conditional "if" operators (<=,>=, <,>, !=, ==, &&, ||) and many functions (i.e., trig functions, log functions, exponents, absolute value, random number, and many more).

If any dependent interval is a Gap, then the derived interval will be a Gap.

Example of a derived stream expression that calculates one second 3 point rolling averages from component1.stream1:

Variables:

<b>Variable</b>	<b>Offset</b>	<b>Stream</b>	<b>Cycle</b>	<b>Cycle Function</b>
n_minus_0	0	component1.stream1	Second	-
n_minus_1	-1	component1.stream1	Second	-
n_minus_2	-2	component1.stream1	Second	-

Expression:

$(n + n\_minus\_1 + n\_minus\_2) / 3$

Other usages of Derived Streams:

- A Stream that filters out the Winter on Peak kilowatt usage
- A Stream that converts Fahrenheit to Celsius
- A Stream that calculates summer weekend afternoon peak temperatures
- A Stream that sums all the energy for all plant meters (aggregates a collection of component streams)
- A Stream that calculates energy cost every minute, hour, day, month, and year
- A Stream that measures the distance between a component and another component every second
- A Stream that applies a diversity factor

Derived stream variables can be different data types. A stream defined as text can be derived from the sum of two other streams of types long and double. GroveStreams will attempt to do the data type conversion wherever it is possible. If the conversion is not possible, the resulting interval will be considered a Gap.

Only dependent stream samples that have matching datetimes will be used to calculate a sample for derived regular and point streams. An interval stream's end date will be used as the sample time during derivation for dependent streams that are interval streams.

More information regarding derived stream can be found in the GroveStreams Modeling Guide.

## System Reliability – The Process Queue

GroveStreams has an internal process queue. Requests to execute small or large processes, such as imports, are made internally through a process queue manager. The process queue manager is responsible for:

- Ensuring a process gets executed
  - Process information is persisted to the store and backed up as each is requested
  - Servers with available resources will attempt to run the task
  - Multiple runs may occur for failed tasks on different servers
- Determining if a process request:
  - Can run immediately. The queue manager makes this determination by looking at the size of the request and how many server resources are available. For example, for file imports, it will look at the amount of bytes being imported and how full the process queue is to determine when the import should run.
  - Must run later when there are more resources available. This can occur if a maximum number of processes are already being processed or the request is a large request
  - Must run as a Job (more about jobs below) if the request is a large request such as a large file import
- Determines how many attempts should be tried for requests that keep failing

Process Examples:

- File Imports
- Stream Aggregation
- Stream Group Updates
- Runnable Executions

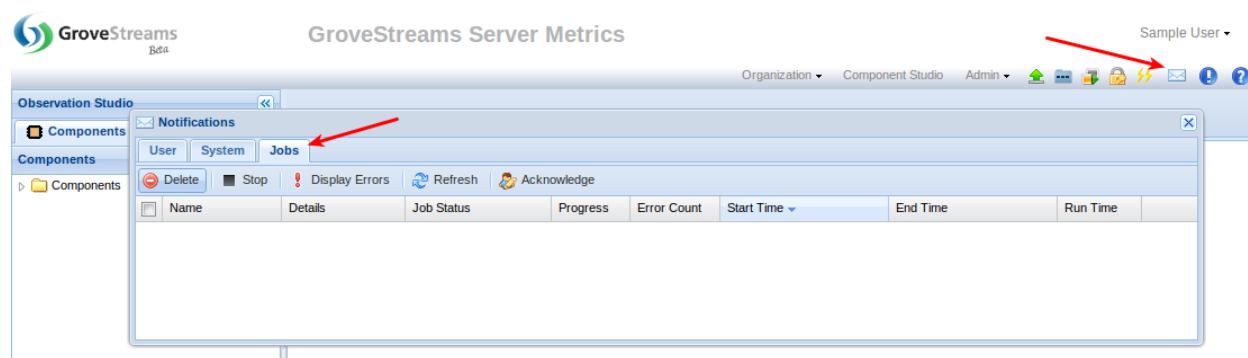
Notifications will appear within an organization's system notification viewer as processes are completed for processes that were started manually. Any errors that may have occurred during a process execution will arrive as a notification.

The process queue can be viewed by selecting: **Admin – Runnables – Process Queue** button on the right.

## Jobs

GroveStreams is designed for small and large customers. Large customers may need extremely large amounts of processing power for certain processes. GroveStreams can auto-detect if a process request is too large to run on one server. If it is too large then the process will run as a Job. A Job will break the request into smaller tasks that run on many servers at the same time in an attempt to complete the job as fast as possible.

Jobs can be monitored and canceled from within the Notifications viewer.



## Job Example

An example of a Job is a large file import. An import request will run as a Job if it is importing a large amount of data. The import Job will be split into many tasks and run on many servers at the same time. The number of server tasks is determined by dividing the number of lines being imported by the number of servers and how many tasks each server can run. Each server is responsible for running one or more tasks, each of which is responsible for the lines it has been assigned. A task will be re-executed on a different server if an error occurs during a task execution to ensure a high degree of reliability.

## Notifications

Notifications are used to provide feedback to users for processes that are occurring in the background.

Click on the **Notifications** button within your GroveStreams organization to view notifications.

There are three types of notifications:

### System Notifications

System notifications are notifications sent from GroveStreams and can be viewed by all users that have Component Studio rights. Note that a System notification will be removed for all users if one user deletes it. Examples of System notifications are:



- Feed and other API errors or warnings. Errors and Warnings are saved as system notifications during a feed upload or any other API call to your organization
- Process queue errors
- Stream event action package processing errors. Examples are email notification errors or http notification errors.

## User Notifications

Stream events can be configured to send notifications as GroveStreams User notifications.

## Jobs Notifications

Job notifications indicate the status and progress of jobs. Queued or running jobs can be canceled from within the job notification viewer. Job errors can also be viewed.

Note that large jobs could generate millions of errors. Try to delete large jobs with a lot of errors as soon as possible to keep your storage costs low.

## File Imports

GroveStreams supports the importing of stream data via one or more text files.

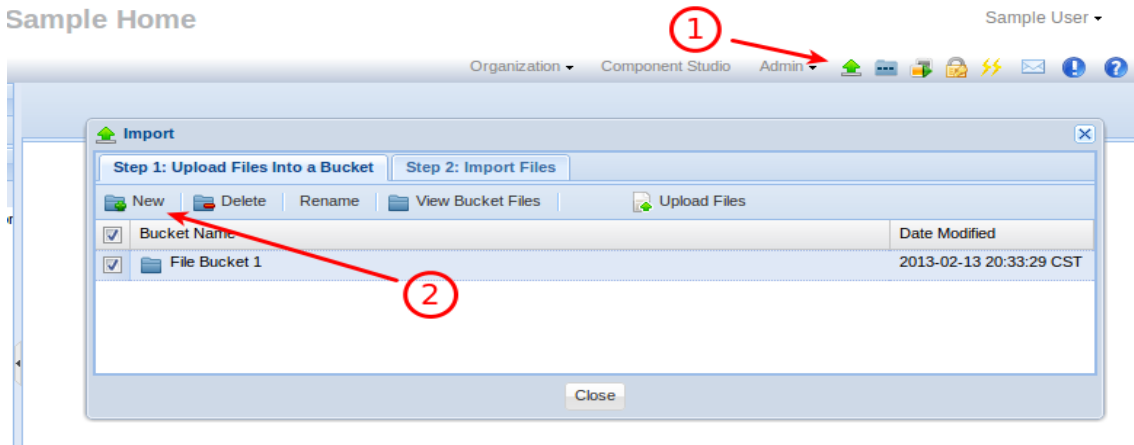
The size of the import can vary from one file with one line, one file with many lines, to thousands of files. All files within the selected file bucket will be imported during a single import. Large imports will automatically be run as Jobs across many servers.

Import information can be saved as an Import Profile so that the same import information does not have to be re-entered each time an import is executed.

Import Profiles can be scheduled to run periodically with the GroveStreams scheduler (see below).

To import one or more files:

1. Create a file bucket
  1. Open the Import Window: Click the **Import** button or select **Admin - Import**
  2. Create a file bucket if you haven't done so already. Select **New** and give your file bucket a name

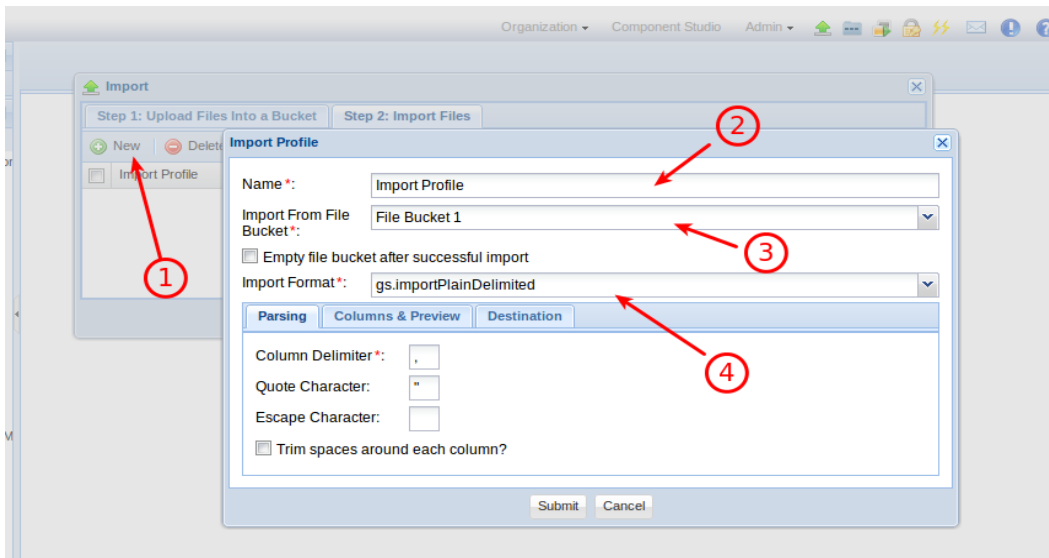


2. Upload one

or more files into the File Bucket

3. Create an Import Profile

1. Select the **Import Files** tab and click the **New** button
2. Enter an Import Profile name
3. Select the File Bucket to import from
4. Choose the Import Format (more about formats below)
5. Click **Submit**



Execute the Import Profile by clicking **Run Import Profile**

There are currently three types of text formats supported:

### GS Stream Format

This is the same format as if the file were generated from Admin-Export-Streams. It has 4 header rows followed by rows of stream data. The first row is the name of each component, The second row is the ID of each component (IDs are optional). The third row is the name of each stream and the fourth row is the ID of each stream. The following rows are stream data. The first column is the datetime in UTC format followed by a column for each stream. This file can contain streams from many components.

Example:

```
, "Component1", "Component2", "Component2"
, "Component1ID", "Component2ID", "Component2ID"
, "Temperature", "Humidity", "Door State"
, "temp", "humidity", "state"
2017-07-10T16:02:18.000, 35.4, 50.4, "OPEN"
2017-07-10T16:02:19.000, 38.4, 60, "CLOSED"
2017-07-10T16:02:12.000, 36.2, 55.2, "OPEN"
```

### JSON Feed Import Format

The JSON Feed format is exactly the same format of the [GroveStreams PUT Feed API](#)

at the component level. It allows the importing of multiple components during the same import. There must not be any new line characters within each file except after each component json definitions.

Example:

```
{
  "stream": [
    {
      "streamId": "light",
      "data": [
        0.10000000149011612,
        1.100000023841858
```

```

]
},
{
  "streamId": "temperature",
  "data": [
    0.10000000149011612,
    1.100000023841858
  ]
},
{
  "streamId": "low_battery",
  "data": [
    true,
    false
  ]
}
],
"time": [
  1376083085818,
  1376086685818
],
"defaults": {
  "name": "Temperature Sensor_0"
},
"componentTemplateId": "sensorTemplate",
"componentId": "Temperature Sensor_0"
} << newline character here (/n) >> {
  "stream": [

```

```
{
  "streamId": "light",
  "data": [
    0.10000000149011612,
    1.100000023841858
  ]
},
{
  "streamId": "temperature",
  "data": [
    0.10000000149011612,
    1.100000023841858
  ]
},
{
  "streamId": "low_battery",
  "data": [
    true,
    false
  ]
}
],
"time": [
  1376083085818,
  1376086685818
],
"defaults": {
  "name": "Temperature Sensor_1"
```

```
},  
"componentTemplateId": "sensorTemplate",  
"componentId": "Temperature Sensor_1"  
} << newline character here (/n) >>
```

## Plain Delimited Import Format

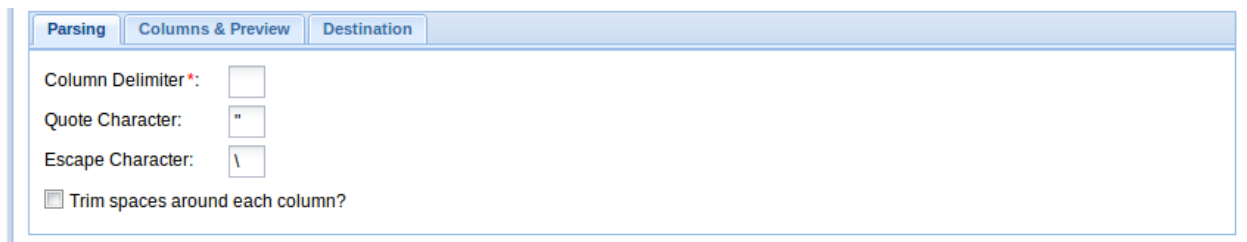
Plain delimited import formats are text files with columns that are separated by a delimiter character. This type of format imports data into a single component's streams. Each column in the import file will import into a stream. Each row corresponds to a sample datetime. One column must be a date and time column.

It is required that you choose a File Bucket that contains at least one file that you would like to import before configuring the **Columns & Preview** tab. This is because the **Columns & Preview** tab interrogates an import file to determine how many columns (with the entered delimiter) there are and it will also be used for the import **Preview** grid.

One column has to be designated as the datetime column. The datetime column contains the datetime of each sample.

The Parsing tab allows you to enter:

- A Column Delimiter. This field is mandatory. This character is the character that separates each of the columns.
- A Quote Character. This field is optional. It will tell the import to ignore delimiter characters within text that is surrounded with the given quote characters.
- An escape character. This field is optional. It is the character to use for escaping a delimiter or



quote

The Columns & Preview tab is responsible for how columns will be parsed during an import.

Fields:

- **Date Column:** The column within the import file that is the datetime column. Zero is the first column.
- **Time zone:** If your date column values do not contain the time zone identifier then select the correct time zone for this import otherwise choose “Use Time Zone in Date Format”
- **Date Format:** This field allows you to choose or enter any user-defined patterns for date-time formatting. Select from the list or enter your own. GroveStreams uses Java date time formatting. See [here](#) for more information.
- **Import File Columns:** The Import File Columns grid allows you to configure how columns should be imported.
  - **Include Column:** Select no to ignore the column during importing
  - **Stream Name:** This will be the name of the stream if you choose “New Component” on the Destination tab
  - **Stream Value Type:** This will be the value type of the stream if you choose “New Component” on the Destination tab
  - **Regex Expression:** Optional. Regex allows you to enter a user-defined pattern to be applied to each value within the column during the import. See more information [here](#) regarding regex. GroveStreams uses the Java String [ReplaceAll](#) when applying regex patterns. Regex patterns can be complicated but they give you a lot of flexibility during your imports.
  - **Regex Replace:** Optional. The value to be substituted for each regex match.
  - **Apply Expression:** Optional. A formula expression that is applied to each value during the import. This is the same expression engine that is used to process derived streams, events, and dashboard widget values. Refer to derivation for more details about expression capabilities. Examples:
    - value \* 10
    - If (value<0, 0, value)
- **Import Preview:** The preview grid allows you to apply your current configuration settings and determine if they are correct. It also ensures the datetime column is being parsed correctly with the entered parsing format. The preview datetime column will report any parsing errors.
  - Click **Refresh** to see how your current configuration settings will be applied
  - Click **Preview** File to see the first few lines of the file that the preview grid is using

Parsing Columns & Preview Destination

Date Column \*: 3 Time zone \*: Use Time Zone in Date Format  
 Date Format \*: dd/MMM/yyyy:HH:mm:ss Z

**Import File Columns**

	0	1	2	Date Time	4	5	6
Include Column?	Yes	No	No		Yes	Yes	No
Stream Name	IP Address				Method & URL	HTTP Result C...	
Stream Value T...	STRING	STRING	STRING		STRING	SHORT	STRING
Regex Express...							
Regex Replace							

**Import Preview**

Refresh Preview File

	0	1 (X)	2 (X)	Date Time	4	5	6 (X)
	174.53.181.101	-	-	2013-05-29 08:...	PUT /api/feed?...	200	-
	174.53.181.101	-	-	2013-05-29 08:...	PUT /api/feed?...	200	-
	174.53.181.101	-	-	2013-05-29 08:...	PUT /api/feed?...	200	-

The Destination tab allows you to choose which component the data will import into.

Parsing Columns & Preview Destination

Import Into:

- New Component (with random streams)
- New Component Based on Template
- Existing Component



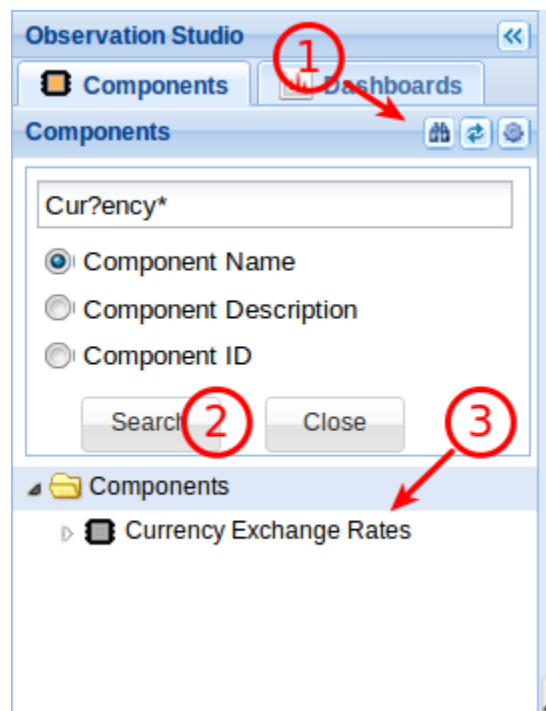
## **Backup and Restore**

Organizations can be backed up to a local compressed zip file on your computer and later restored from that zip file. The backup zip file will contain a blueprint for your organization and a stream export file for each component in your organization. The stream format is the same format as the GS Stream format used in the stream file export and import.

You may restore to a new organization or restore to an existing organization. This may be a lengthy process if your organization is large. Restoring streams will run as a Job since it is a lengthy process.

## Searching for Components

1. Click the **Search** toolbar icon within the Observation Studio component browser. Wild card '\*' and '?' characters are supported.
2. Click the **Search** button.
3. A list of component results is returned.



## Stream Groups

Stream groups are just what they imply - a group of streams!

Stream groups are mostly used to automatically include newly registered component streams within existing reports, maps, and streams derived from aggregating other streams.

Stream Groups are built to handle large collections of streams. Do not try and graph a Stream Group, within a dashboard, that could have hundreds, thousands, or millions of streams. *Large* Stream Groups are mainly used for aggregating large numbers of streams.

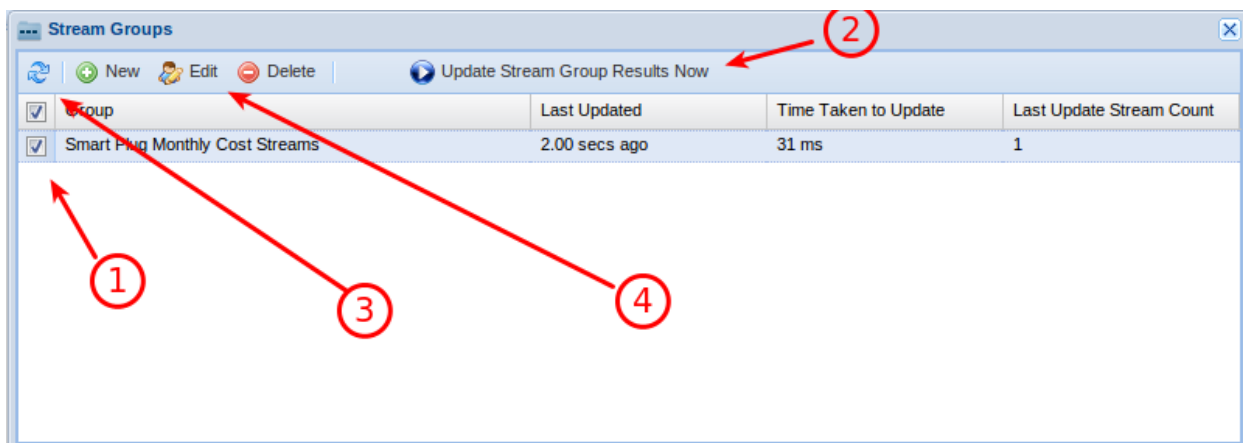
Streams become part of a stream group by matching a set of criteria configured for a group. Criteria such as:

- Stream wildcard name filters
- Stream id wildcard filters
- Which folder or component the stream belongs to

Streams do not automatically become members of a Stream Group after it has been created. The **Update Stream Group Results Now** button needs to be clicked or a scheduled Runnable can be created to run periodically so that group membership becomes automatic. Be sure not to schedule Stream Group updates to run too close together. If a group is including a large number of streams, it could cause the runnable update runtime to be more than your schedule cycle and system errors will be generated that state that a runnable failed because it already exists within the queue.

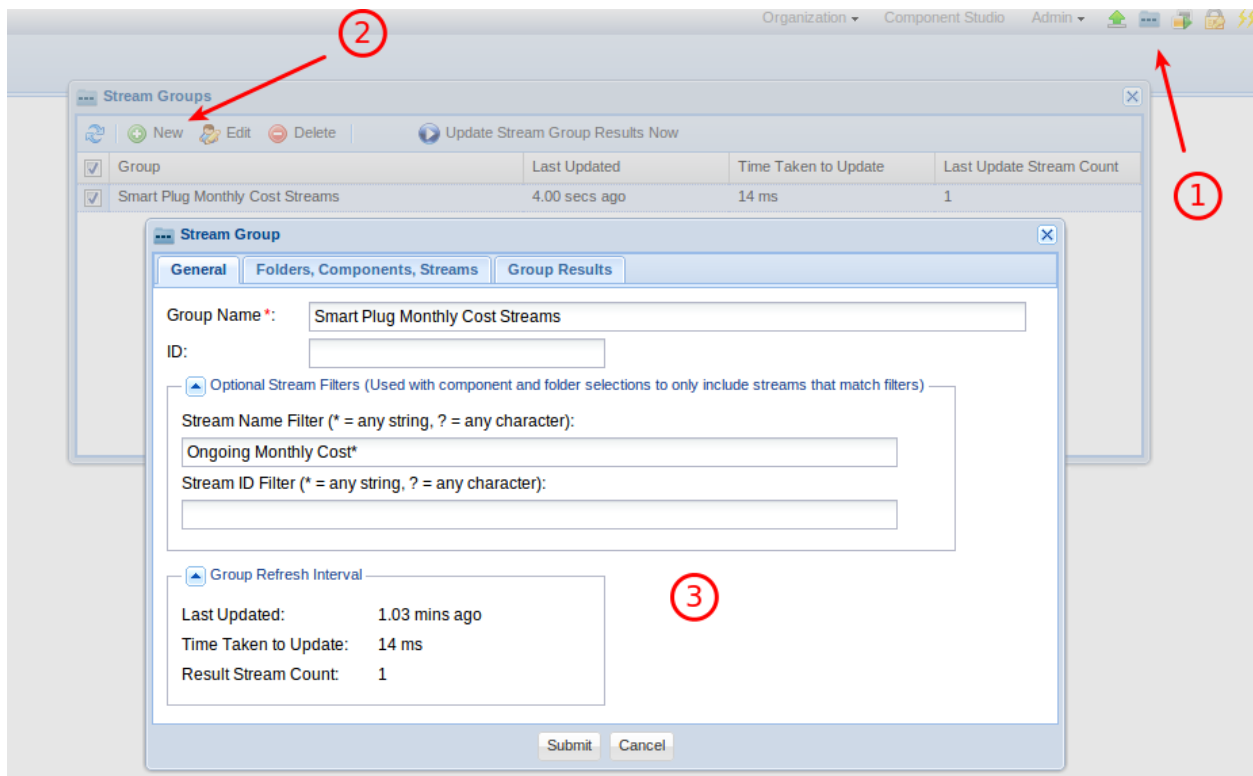
To create a Stream Group, click the **Stream Group** button within the studio toolbar:

1. Select the Stream group
2. Click **Update Stream Group Results Now** to include streams within the group
3. Click the **Refresh** button to see how long it took to run the update and to see how many streams now belong to the group
4. Click **Edit** to edit your Stream Group



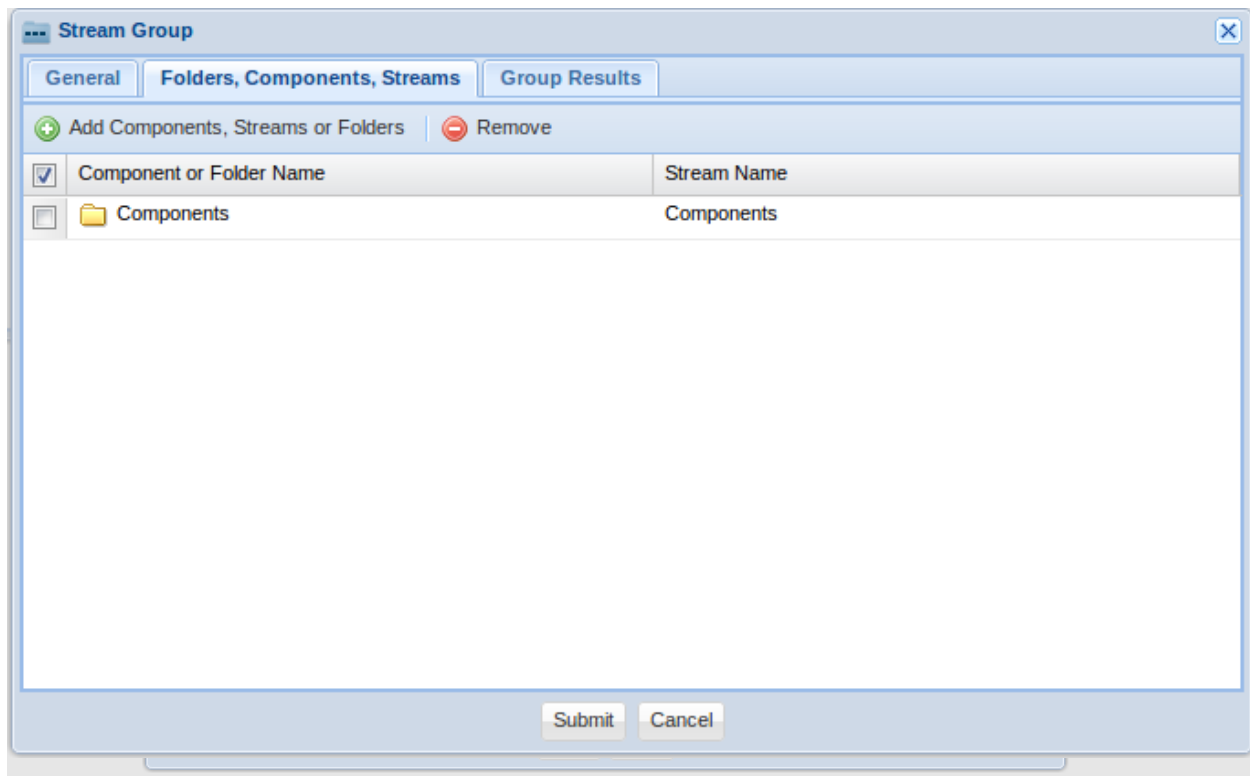
To create a new Stream Group:

1. Click the **Stream Groups** toolbar button
2. Click the **New** button
3. Enter General Information:
  - Group Name: The name of the Stream Group
  - ID (Optional): The ID of the Stream Group. Mainly used for API purposes
  - Stream Name Filter (Optional): Enter the name of streams that will be added to this group. Wildcards are supported. Enter '\*' for any string. Enter '?' for any character. For example, if you would like to include any strings that have a name that starts with "light" you would enter "light\*".
  - Stream ID Filter (Optional): Enter the ID of streams that will be added to this group.



On the Folders, Components, Streams tab:

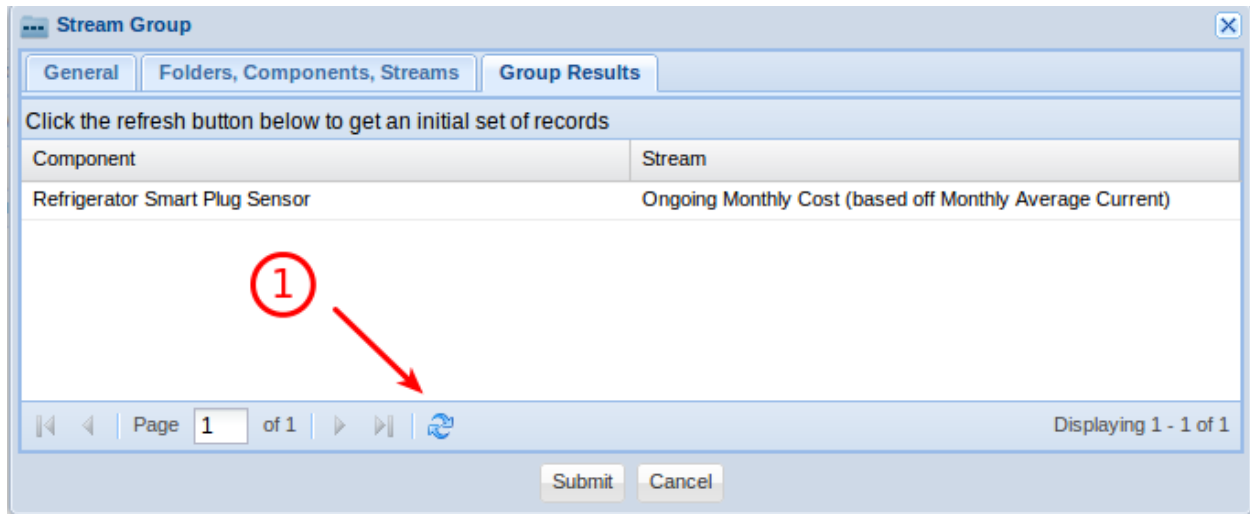
- Add Components, Stream, and Folder that you would like to restrict the search to.



- All folder subdirectories are assumed to be included for each selected folder.

Group Results Tab:

1. Click the **Refresh** button at the bottom of the tab to see a list of streams that currently belong to the Stream Group.



## Stream Group Inclusion Algorithm

- All streams that are selected explicitly on the “Folder, Components, Streams” tab are automatically included (name and id wild cards are not applied) in the group.
- All streams that are selected via folders and components on the “Folder, Components, Streams” tab will have the name and id wild cards applied to them, and if a stream passes the wild card filter, it be added to the group. Note that if no name or id wild card has been entered, then all of the streams within the folder and components will be added to the group.

## Runnables, Processes, and the Scheduler

A Runnable is a group of processes. Runnables can be executed in batch or scheduled to run periodically.

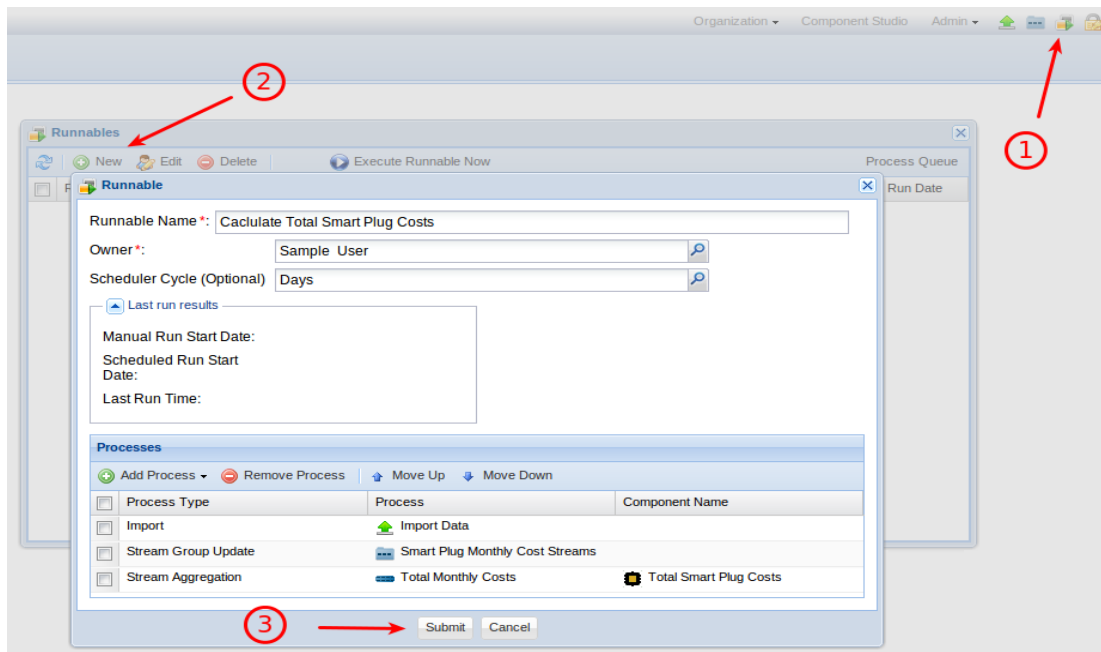
A Runnable is a great way to schedule a set of processes that need to occur periodically such as the importing of files, performing stream aggregation and updating stream groups so that newly registered components are picked up.

All of a Runnable's processes will be placed into the Process Queue when the Runnable executes and is processed as server resources become available.

Any errors that occur during processing are delivered as System notifications.

1. Click on the **Runnables** button within a studio toolbar to create a runnable

2. Click **New** to create a new Runnable
3. Click **Submit** to save your new Runnable



- **Runnable Name:** The name of your Runnable
- **Owner:** The user that owns this Runnable. Some processes may need a user. The Runnable owner will be used for those processes. An example is an import that might create components. The Runnable owner will be assigned as the owner of the new component
- **Scheduler Cycle:** Select how often the Runnable should run. Ensure that the cycle is not too small. If it is smaller than the time it takes to complete a Runnable execution, the organization will get a system notification indicating that a process could not be added to the queue since it already exists within the queue. Also, you are charged by your usage of GroveStreams. The more process CPU that is used, the more your usage costs will be.
- **Processes:** Add processes that are part of this runnable. Processes will be executed synchronously in the order that they occur in the list. All processes will run even if a prior process failed.

## Queries

Grove Streams supports its own Structured Query Language (SQL). A user defined Query entity contains an SQL statement and can be saved and secured within the Tools content store. A result set is saved in the GS cloud whenever a Query SQL statement is executed. The result set is cached and can be received at any time without executing the SQL statement. This is useful for long running queries or queries that need to run at a specific day and time

## Query Schedules

A Query schedule can be configured within the Query definition.

A Running User must be selected as part of the schedule. The Running User's time zone will be used while executing the Query SQL statement and **the Running User's content security rights will be used to filter the results of the query** so that the cached results only contain items the user has Read rights for.

A query can also be scheduled within a Runnable as part of collection of other processes (see above).

## Query Security

Queries are secured the same way all Tool Content Store items are secured.

The Time Zone and Content rights are used during a cache refresh for the following user:

- Manual Cache Refresh from Observation Studio: Uses the Query's Scheduled Running User if a Schedule is being used. Otherwise, it uses the user who started the refresh. The user who started the refresh needs Execute rights for the Query object.
- Scheduled Refresh: The cache will be refreshed using the Schedule's Running User rights.
- Within a Runnable: The cache will be refreshed using the User selected within the Runnable.

Only users with Read access to the Query can view the Query's result set.

## API Keys

API keys are used to secure API calls for each GroveStreams organization.

The API Key Editor allows you to create and configure a secret key token with specific rights. The token is passed with each HTTP API call to GroveStreams.

A single api\_key token can grant rights to the entire organization API or it can be defined to only grant one specific action to a specific resource instance from a specific IP address.

Examples of resources are components, streams, and dashboards. An example of a resource *instance* is a specific component such as "Weather Station Number 5".



An API secret key is used by devices to:

- Prevent hard-coding of a user email address and password on a device
- Avoid session timeouts
- Provide the capability to grant only rights to a specific device in case the secret key becomes compromised

API secret keys are also used by GroveStreams embeddable charts and RSS feeds.

The secret key is passed to the server either as a URL argument or as a cookie with the name "api\_key". Be aware that if a session token is passed with an api\_key secret key at the same time, the session token will be used for authentication and for determining access rights and the api\_key will be ignored.

The api\_key server architecture is very scalable. An organization can have many keys - one for each component or stream if that's what their security demands require.

Child resources inherit their parent's rights. There may exist a condition where a parent resource has certain rights but one or more of its children has other rights set. For example, GET only rights might be given to all component entities, but then GET, DELETE, PUT, POST rights might be given to one specific stream. When rights are given to a child resource, those rights will override any parent rights. For the previous example, GET rights will be allowed for all streams except the specified stream will have GET, DELETE, PUT, POST rights.

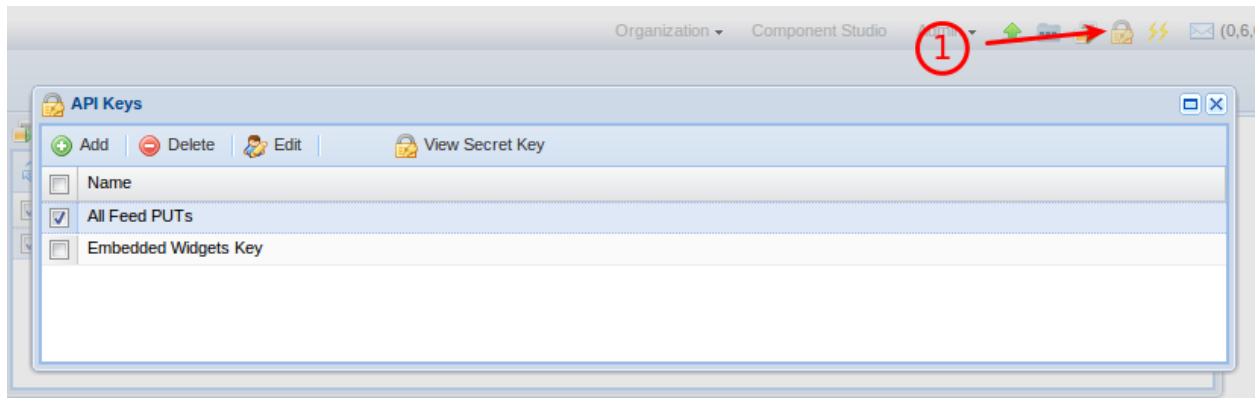
Valid Actions are:

- GET
- PUT
- POST
- DELETE

Include a list of API originating IP addresses with the API key to only allow calls from those addresses. All calls from any addresses will be allowed if no addresses are listed. IP addresses are comma delimited.

Only GroveStreams administrators have rights to create or modify api\_keys.

1. Click the **API Key** toolbar button from within the studios to manage API Keys.



## api\_key Examples

- Grant all actions to the entire API:
  - resource: "\*"
  - actions: "GET, PUT, POST, DELETE"
- Grant no actions to entire API (makes api key invalid):
  - resource: ""
  - actions: ""
- Grant read-only rights for all streams within an embedded widget:
  - resource: "component/\*/feed"
  - actions: "GET"
  - resource: "component/\*/stream"
  - actions: "GET"
- Grant GET only action to entire API:
  - resource: "\*"
  - actions: "GET"
- Grant GET only action to all streams and feeds of a specific component instance (component/{compUId}):

- resource: "component/8d6bb905-0e71-4321-9b0c-f8fe85cbc77f/\*"
- actions: "GET"
- Grant GET only action to all children of a specific component instance except restrict to PUT,POST to one specific stream's feed (component/{compUid}/feed/{streamUid}:
  - resource: "component/8d6bb905-0e71-4321-9b0c-f8fe85cbc77f/\*"
  - actions: "GET"
  - resource: "component/8d6bb905-0e71-4321-9b0c-f8fe85cbc77f/feed/567242a6-e800-3510-8812-1fc6f0218ea1:"
  - actions: "PUT,POST"
- For Component Feeds (commonly used for RSS Feed rights)
  - Grant GET only action to all component's feeds (component/{compUid}/feed):
    - resource: "component/\*/feed"
    - actions: "GET"
  - Grant GET only action to one specific component stream's feed(component/{compUid}/feed/{streamUid}):
    - resource: "component/8d6bb905-0e71-4321-9b0c-f8fe85cbc77f/feed/567242a6-e800-3510-8812-1fc6f0218ea1"
    - actions: "GET"
  - Grant GET only action to all stream feeds of a specific component (component/{compUid}/feed/\*):
    - resource: "component/8d6bb905-0e71-4321-9b0c-f8fe85cbc77f/feed/\*"
    - actions: "GET"
- For Batch Feeds (commonly used for uploads from devices)
  - Grant GET only action to all batch feeds
    - resource: "component/\*/feed/\*"
    - actions: "GET"
  - Grant GET and PUT only actions to one specific component stream's feed used in a batch call (feed/{compUid}/stream/{streamUid}):

- resource: "feed/8d6bb905-0e71-4321-9b0c-f8fe85cbc77f/stream/567242a6-e800-3510-8812-1fc6f0218ea1"
- actions: "GET, PUT"
- Grant GET and PUT only action to all stream feeds of a specific component used in a batch call(feed/{compUid}/\*):
  - resource: "feed/8d6bb905-0e71-4321-9b0c-f8fe85cbc77f/\*"
  - actions: "GET, PUT"


## API Editor

**Edit API Keys**


API Key Name \*:

Addresses (comma separated ip addresses that can use the key. No addresses implies any address.):

**Resources**

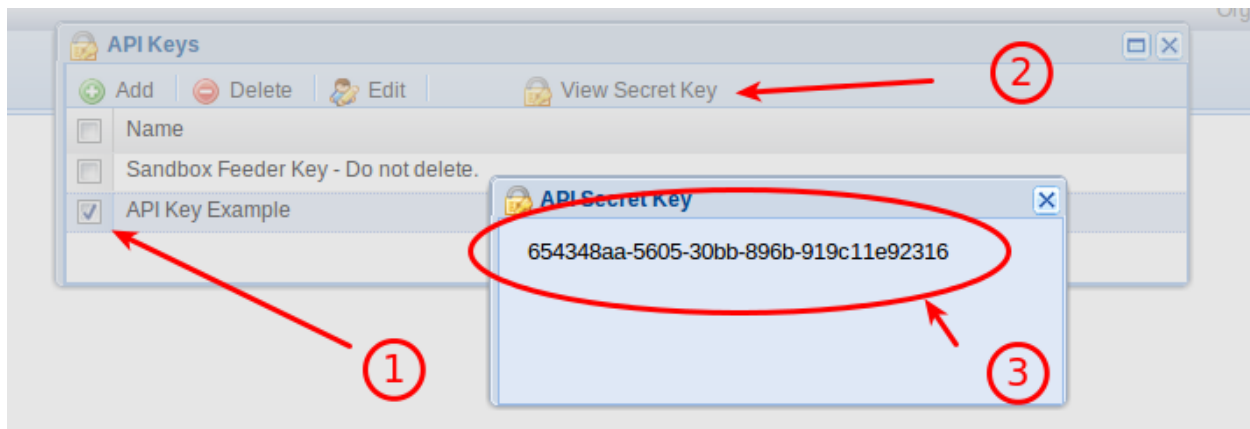
Add Instance ▾ Add Type ▾  Remove

<input type="checkbox"/>	Resources	GET	PUT	POST	DELETE
<input type="checkbox"/>	feed	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	component_template	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	component	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Submit Cancel  Help

## Viewing an API Secret Key

1. Select an API Key
2. Click **View Secret Key**
3. Select and copy your secret key to the clipboard (ctrl-c) and use it on your device, embeddable chart, or RSS feed



## Time Zone Support

Time zone support is very flexible. All components must reference a time zone. All cycles reference a time zone or point to a component's time zone.

All times are formatted and presented in the time zone that your browser has been configured to use as they are requested.

Stream rollup calculations use the time zone and cycle reference date to determine the actual UTC time to associate with the rollup interval times.

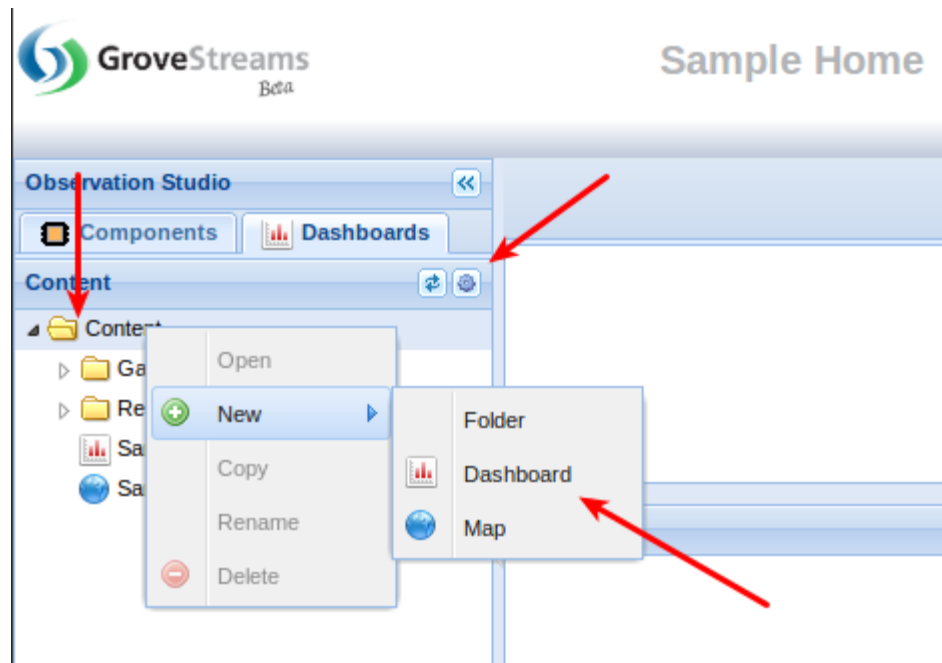
The ability to fix a time zone to a cycle or to a component allows for a rollup to occur within a specific time zone (such as the corporate headquarters). For example, an oil company may want to monitor daily real-time oil production from each well. Their definition of a day might be the corporate headquarters time zone day. They would set their time zone in each rollup cycle, forcing each well's daily oil production to be calculated using the corporate headquarters definition of a day. If the same oil company wanted to track production in each well's time zone, then they would just point each rollup cycle time zone to the component's time zone.

## Dashboards

Dashboards allow stream data or component locations to be viewed in near real-time.

To create a dashboard:

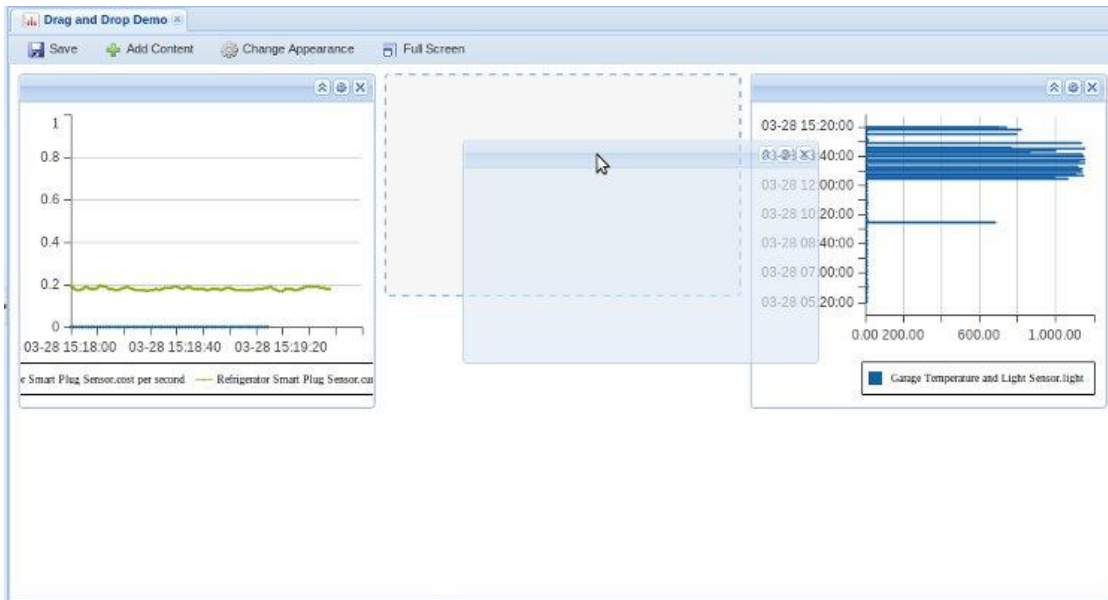
1. Login and enter an organization
2. Within Observation Studio, right click a folder on the Dashboards tab and select **New – Dashboard**. If your viewer does not support right clicks, then select a folder and click the cog icon to get the menu to appear.
3. Give your Dashboard a name



Click **Add Content** to add chart, grid, RSS or map widgets.

Click **Change Appearance** to adjust the number of columns.

Drag and drop your widgets to the location you would like them:



## Adding Components and Streams to a Widget

There are two techniques to add data to your widget:

1. Simply drag and drop a stream or component from the Components tab on the left onto a widget. Many widget settings are set automatically when the first object is added to a widget from dragging and dropping it.
2. Click the Cog icon at the top of the widget and add components or streams from the Data tab

## Changing a Widget's Settings

Click on the Cog icon at the top of the widget to change settings such as:

- Line color and thickness
- Widget's height
- Widget's title
- Data
- Time Range or refresh rate

## Chart Widget Settings

### General Tab

- Title: Optional. The title that will appear above your widget.



- Height: The height of your widget
- Range Type:
  - Heartbeat: Polls and displays the latest data
  - Fixed Range: Displays data for a fixed range
- Polling: How often the widget is refreshed. This field is only available if Range Type is Heartbeat.
- View Last: How many samples to display
- Cycle: Optional. When a cycle is selected:
  - The widget graph will have an axis tick for every interval within the cycle. The cycle option is very useful for identifying gaps.
  - Some stream feeds will be dynamically converted to interval feeds with the cycle's interval size and time zone. The browser's time zone will be used if the cycle is not referencing a time zone.
  - Regular and point streams will be converted to interval streams for graphing.
  - Each interval's end datetime is used for graphing.

### Data Tab

The data tab is where streams are added to a widget. Add stream groups if you would like streams to be dynamically added to this widget as they are included in the stream group (see Stream Groups above for more information regarding Stream Groups).

You can display any of a Stream's virtual streams if it has been associated with a rollup calendar during its definition. Virtual stream data is available as soon as a stream's feed has been uploaded into GroveStreams.

To include a virtual stream:

- Select a stream
- Change its Cycle
- Change its Statistic

### Graph Tab

- Vertical Axis Title: Optional.
- Vertical Axis Number Format. Allows the vertical axis number to be formatted. Examples:

- 0,000 : Display commas every three digits. Show no decimal places.
- 0,000.000 : Display commas every three digits. Show three decimal places

## Grids Widget Settings

The grid widget displays actual sample values, previous values, the delta, trending direction, and much more.

Grid columns can be dragged and dropped to new locations.

### General Tab

- Title: Optional. The title that will appear above your widget.
- Height: The height of your widget
- Range Type:
  - Heartbeat: Polls and displays the latest data
  - Fixed Range: Displays data for a fixed range
- Polling: How often the widget is refreshed. This field is only available if Range Type is Heartbeat.

### Data Tab

The data tab is where streams are added to a widget. Add stream groups if you would like streams to be dynamically added to this widget as they are included in the stream group (see Stream Groups above for more information regarding Stream Groups).

### Columns

Select the columns to be viewed with the grid.

### Cycle Columns

Select a cycle to enable virtual stream value selections. When a cycle is selected:

- Some stream feeds will be dynamically converted to interval feeds with the cycle's interval size and time zone. The browser's time zone will be used if the cycle is not referencing a time zone.
- Regular and point streams will be converted to interval streams for graphing.

## Maps Widget Settings

GroveStreams uses Bing Maps for displaying component locations. A component can be defined to be "fixed" or "mobile". Both types of component locations can be viewed within a dashboard.

Along with the component's location, the map widgets will display where active events are occurring for each component.

There are two types of Map Widgets:

1. Area Map: Displays components and events that are within a defined map boundary.
2. Track Map: This widget will track the location of a single component and its events. The Component will remain in the center of the map. Drag and drop a component onto this widget or click on the cog icon, select **Settings**, and select a component.

## RSS Widget Settings

The RSS widget allows the viewing of internal or external RSS feeds within a GroveStreams dashboard. Drag components or streams onto the RSS widget or click on the cog icon, select **Settings** to select each URL.

## Stream Feed Form Widget

This widget displays each stream's last value and allows users to append on a new stream value with the sample time being the time on your computer (rounded to the second) when you click the button.

The combination of the Stream Feed Form widget and the Feed PUT (all on the URL) request streams parameter (rsid) will allow your device to accept commands and settings from a dashboard through a firewall while keeping transaction rates the same.

It's also a great way for users to hand enter values that are needed in derivation formulas.

- The form on the left had several streams dragged onto it without any widget settings changed.
- The form on the right is the same set of streams with some formatting changes
- Validation rules can be applied and a marker will indicate when the rule fails and a **Submit** will not be allowed.
- Any stream can be dragged onto a single form: Point, Regular, Interval.
- Streams can belong to different Components
- The editing control is determined by the stream type: Boolean is a Radial Control. Number is a Number Field. Text is a text field. DateTime is a time picker.
- The Regex property on streams that are Text/String value types, is used for validation. Search the web for Regex and validation.
- Like most of our other widgets, this widget is:

- embeddable (**Your API key will need to allow a Feed PUT!**)
- exportable to a spreadsheet
- Here are some sample style settings:
  - Label: **font-size:20px; font-family: Florence, cursive, Times, serif; font-weight:bold;font-style: italic;line-height: 35px;color:red;**  
  
The line-height style is used to help center the label.
  - Field: **font-size:20px; font-family: Florence, cursive, Times, serif; font-weight:bold;font-style: italic;color:red; background-color: yellow; color:red;background-image: none;**

## Embeddable Graphs and Grids

Some widgets can be embedded in external websites much like YouTube videos can. Click on the widget cog icon and select **Embed this Widget**.

API Secret Key: Paste a secret API key here that will give rights to read a stream's meta data and feed (both are needed by most embeddable widgets).

Example of an API key that gives read-only access to all stream meta data and all stream feed data within an organization:

**Edit API Keys**

API Key Name \*:

Addresses (comma separated ip addresses that can use the key. No addresses implies any address.):

**Resources**

Add Instance ▾ Add Type ▾ Remove

<input type="checkbox"/>	Resources	GET	PUT	POST	DELETE
<input type="checkbox"/>	component*/feed	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	component*/stream	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Help

Height: The height, in pixels, for the embedded widget

Width: The width, in pixels, for the embedded widget

Show Border: Whether to show a border around the widget

Connection Type: How the widget should communicate with GroveStreams

Click **OK**, then copy and paste the html java script code into your external website. Any changes made to this widget will ripple through to the embedded widget. For example, if new streams are added or line colors are changed, those changes will occur in the embedded widget when the page it is embedded on is refreshed.

## Export to a Spreadsheet

Some widgets support exporting their data to a spreadsheet. Click on the widget cog icon and select **Export to Spreadsheet**.

Exporting to a spreadsheet requires that a spreadsheet program, such as Microsoft Excel, be installed on your computer.

## Component View

A component view displays component information, its location and the latest values for each of its streams. To open a component view right click on a component within the component browser on the left side of observation studio and click **View Component**. If your viewer does not support right click, click on the cog icon within the component browser to make the pop-up menu visible.

## Stream View

To open a stream view, right click on a stream within the component browser on the left side of observation studio and click **View Stream**. Stream views can also be opened by clicking on streams within a component view or within a dashboard widget.

A stream view is a great way to analyze, add, edit, or delete a stream feed data.

Gaps, or null values, will be identifiable by no value appearing.

Virtual Streams:

- Virtual streams can be viewed and graphed within a stream view if a rollup calendar has been associated with an interval stream. Select your rollup cycle and then click the **Data Points** button to view a virtual stream.
- Virtual streams, within the viewer, support drilling down from a rollup cycle down to the next cycle (i.e. Drill down from days to hours). This is useful for uncovering anomalies within a feed at a high level and then zooming in to the actual base cycle value to determine the cause of the anomaly. To drill into a cycle, choose any cycle besides the base cycle and click on a point within the graph.
- For minimum and maximum virtual streams, the datetime of each occurrence within a rollup cycle will also be displayed within the grid.
- Virtual stream gap counts are also available for viewing and graphing for each rollup cycle.

Drag and drop other streams or click the **Compare** button to compare a stream to similar streams within the same view.

Click the map icon next to each sample to view the component's location at the time the sample was taken.

Select **Chart Type** at the bottom to change the chart.

## Export Stream Feed Range to a Spreadsheet

Click the **Export to Spreadsheet** button on the left grid of the stream view to export the current range of samples to a spreadsheet.

## Subscribing to a Stream's RSS Feed

Click the **Subscribe to Feed** button on the left grid of the stream view to the current stream's RSS feed. Include a secret API key if the RSS feed will be used outside of the organization that the stream belongs to.

## Quick View

The quick viewer is located at the bottom of observation studio. It quickly displays the latest component or component stream information when a component or a stream has been selected.

The quick view can be minimized or closed by clicking the **Minimize** or **Close** buttons in its title bar.

## Active Events Viewer

Active component events can be viewed by clicking the **View Active Events** button within the studio toolbar.

Events are configured as part of a component's definition. Right click on a component and choose **Edit** to add, change or remove any events.

The active events viewer will display many attributes about an event including:

- The datetime that the event was triggered
- How much time has elapsed since the event started
- The name and category of the event
- The stream the event was triggered on
- The stream value that triggered the event

Click on the component icon to view the current location of a component within a map

Click on the event icon to view the location with a map where the component was located when the event started.

Select one or more events and click the **Delete** button to cancel and delete any active events. Deleting an event will prevent any Stop Trigger Actions from occurring such as sending an end event notification, email, or SMS texts if they have been configured for that event.

## Custom Application Development

GroveStreams is more than an application; it is a platform that custom applications can be built on top of.

Almost all of GroveStreams' capabilities are exposed through its public API. The current browser and mobile applications have been built using the public API.

Use the GroveStreams public API to build your own custom application.

### Cross site scripting:

- The GroveStreams browser and mobile applications use the REST based API directly from within a browser' client side java script. Because of cross site scripting security concerns, this technique may not work for all custom applications depending on the type of calls being made. If cross site scripting is an issue that you are experiencing, then move the GroveStreams API calls from your browser to your server.

More information about development and the GoveStreams API can be found by choosing **Support – Developers – Overview** from the menu located at the top of any non-studio page.

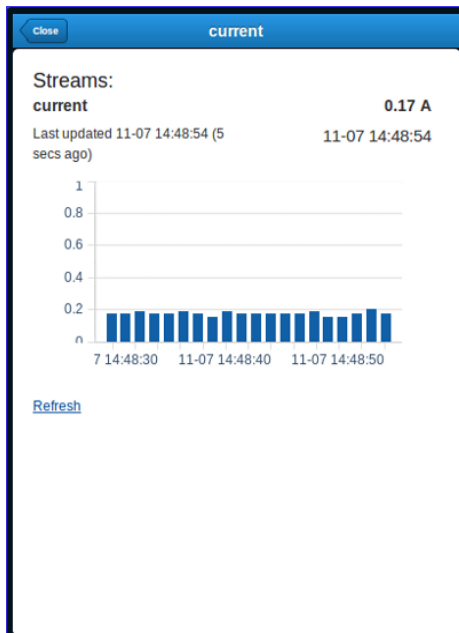
## Mobile Application

GroveStreams provides a mobile application. The mobile application requires that your mobile device have an Internet connection and a browser that supports HTML 5.

The mobile application displays component, stream, and notification information in views that are compatible with mobile devices.

Open your browser and navigate to the [www.grovestreams.com](http://www.grovestreams.com) and click on the **Mobile site** link at the top of the page to use the mobile application on your mobile device.





## Component Studio

Component Studio is where component templates, units, time filters, rollup calendars, and other entities are modeled.

Typically, only advanced users or administrators are given rights to component studio.

Click the **Component Studio** button with the observation studio to open component studio.

See the GroveStreams modeling guide for more information regarding Component Studio.