

# opEvents 2.4.x - Customising Table Columns

- [Feature Description](#)
- [Views That Support Custom Columns](#)
- [Available Columns](#)
- [Configuration](#)
  - [Configuration Files](#)
  - [Enabling the Feature](#)
  - [Configuration](#)
  - [Property Attributes](#)
    - [Attribute Descriptions](#)
  - [Adding and Removing Columns](#)
- [Verification](#)
- [Event Context](#)

## Feature Description

This is an advanced feature that allows customers to modify the data presented in tables. The default table content has been carefully selected based on Opmantek's numerous years of network management experience. This said there are cases where customers would like to alter the table data to display information that is important to their organisation. Most opEvents pages with table data can be customised to display different content. This feature allows table columns to be removed or added as required.



If when following these steps the View that is being modified does not behave normally; remove the custom configuration file and normality should be restored.

## Views That Support Custom Columns

- [Actions](#)
- [Raw Logs](#)
- [Archive Logs](#)
- [Nodes](#)
- [Summary Reports](#)
- [Edit Nodes](#)
- [Dashboard](#)
- [Events](#)
- [Event Context \(Node Summary\)](#)

## Available Columns

Columns properties that may be added can found here: [opEvents Normalised Event Properties](#)

## Configuration

### Configuration Files

Each view has a separate configuration file that will be found in the following directory (by default this directory does not exist; reference the next step):

/usr/local/omk/conf/table\_schemas

View	Configuration file
Actions	opEvents_action-log.json
Raw Logs	opEvents_raw-log.json
Archive Logs	opEvents_archive-log.json
Nodes	opEvents_node-log.json
Summary Reports	opEvents_summary-reports.json
Edit Nodes	opEvents_node-ip.json

Current Events	opEvents_current-events.json opEvents_active-events.json
Events by Node (Dashboard)	opEvents_index_events.json
Events	opEvents_current-events.json

## Enabling the Feature

In order to enable this feature the following must be done.

- Create a directory called `/usr/local/omk/conf/table_schemas`
- Copy the specific view configuration file that requires modification from `/usr/local/omk/lib/json/opEvents/table_schemas/` into `/usr/local/omk/conf/table_schemas`.
  - Only the necessary json files should be copied to the `/usr/local/omk/conf/table_schemas` directory as having unnecessary config files in this directory will result in future upgrades being unpredictable.



### Caution!

Enable this feature with care.

Future opEvents upgrades will need to be watched carefully as tables and event properties can change across versions. Based on this an upgrade has the potential to break the functionality of a custom table configuration. If this feature is enabled it is highly recommend to upgrade in a test environment prior to upgrading the production environment.

## Configuration

The configuration files are json files that have a specific syntax that must be observed. Add the desired property to the json file in the order it should appear in. The table will be constructed left to right based on attributed that are read from the top down.

## Property Attributes

Each property that is added will require a set of attributes. This is an example of the attributes that belong to the 'comment' property.

```
{
  "name": "comment",
  "label": "Comment",
  "cell": "string",
  "search": false,
  "editable": false
}
```

## Attribute Descriptions

- name: Name of the event property ([opEvents Normalised Event Properties](#))
- label: The column name that will render in the web page.
- cell: The cell type, usually this will be "string". For other options please contact [support@opmantek.com](mailto:support@opmantek.com).
- search: Always false
- editable: Always false

## Adding and Removing Columns

To remove a column simply remove the associated section from the applicable json file. To add a column add a new section in the json file. The column placement will be relative to the order it is put in the json file.

The example below is the `opEvents_action-log.json` file. The version on the left is the default version. The version on the right adds an Event ID column between the data and event columns, it also removes the comment column.

```

// VERSION=0.4.0
[
  {
    "name": "date",
    "label": "Date",
    "search": false,
    "cell": "string",
    "direction": "descending",
    "editable": false
  },
  {
    "name": "event",
    "label": "Event",
    "cell": "string",
    "search": false,
    "editable": false,
    "cell": "LookupUrl",
    "replace_name": "id",
    "base_url_stash_key": "event_base_url"
  },
  {
    "name": "node",
    "label": "Node",
    "search": false,
    "editable": false,
    "cell": "LookupUrl",
    "base_url_stash_key": "node_base_url"
  },
  {
    "name": "action",
    "label": "Action",
    "cell": "string",
    "search": false,
    "editable": false
  },
  {
    "name": "details",
    "label": "Details",
    "cell": "string",

    "editable": false
  },
  {
    "name": "comment",
    "label": "Comment",
    "cell": "string",
    "search": false,
    "editable": false
  }
]

// VERSION=0.4.0
[
  {
    "name": "date",
    "label": "Date",
    "search": false,
    "cell": "string",
    "direction": "descending",
    "editable": false
  },
  {
    "name": "eventid",
    "label": "Event ID",
    "cell": "string",
    "search": false,
    "editable": false
  },
  {
    "name": "event",
    "label": "Event",
    "cell": "string",
    "search": false,
    "editable": false,
    "cell": "LookupUrl",
    "replace_name": "id",
    "base_url_stash_key": "event_base_url"
  },
  {
    "name": "node",
    "label": "Node",
    "search": false,
    "editable": false,
    "cell": "LookupUrl",
    "base_url_stash_key": "node_base_url"
  },
  {
    "name": "action",
    "label": "Action",

    "search": false,
    "editable": false
  },
  {
    "name": "details",
    "label": "Details",
    "cell": "string",
    "search": false,
    "editable": false
  }
]

```

For example, the image below shows the default menu bar for the Actions view in opEvents.

Date -	Event	Node	Action	Details	Comment
--------	-------	------	--------	---------	---------

After editing the opEvents\_action-log.json file the extra column was added to the end of the row labeled "New Custom Column". This process can be replicated to display custom information in as many columns as you would like on this page.

Date -	Event	Node	Action	Details	Comment	New Custom Column
--------	-------	------	--------	---------	---------	-------------------

## Verification

It is not necessary to restart any daemons. After editing the associated json file simply load (or reload) the view in question.

## Event Context

For the Event Context node summary data can be displayed using `opevents_gui_event_node_summary_list` config option in `opCommon.nmis`

By default these properties are displayed in the gui, any changes will require the `omkd` daemon to be restarted.

```
'opevents_gui_event_node_summary_list' => ["name", "group", "location", "customer", "businessService", "host"]
```