OpenBoard Dashboard API

Author and technical contact: Paul Haesler
paul.haesler@data61.csiro.au

Version History

Version	Release Date (d/m/y)	Changes
1.0	9/10/2015	Rebranded as OpenBoard. Restarted version number at 1.0
2.0	?/?/2016	Change views from cartesian product of frequencies, locations and themes into navigable hierarchies. Backward incompatible to version 1.0, so incrementing version number to 2.0.

Openboard API

This document describes the Openboard dashboard API.

1 Authentication

Some API methods may require authentication.

Cookie-based authentication takes the form of a cookie named “sessionid” containing an encrypted session identifier (the cookie is obtained via the login method described below).

Because the API calls are being served from a different domain to the front end HTML, this requires correctly set CORS (and P3P for IE) headers to comply with browser security.

Header-based authentication takes the form of an HTML header: X-Dashboard-Session-Id which contains an encrypted session identifier, as returned by the login method described below.

1.1 Login

URL: api-root/login

GET args:

Argument	Description
username	The username of the user to login
password	The password of the user to login.

AUTHENTICATION: Not required.

RESPONSE description: A string containing the session id. Also sets a sessionid cookie which can then be sent with subsequent API calls.

COMMENTS: Arguments can be sent as either GET or POST parameters. Incorrect username/password results in a 403 FORBIDDEN response.

POSSIBLE FUTURE DIRECTIONS: Input arguments should only be accepted as POST parameters. Using GET parameters leaks passwords into the Apache log, which is a security risk.

1.2 Logout

URL: api-root/logout

GET args: None

RESPONSE description: An empty list.

COMMENTS: After calling logout, the sessionid is no longer valid for use in other API calls.

1.3 Change Password

URL: api-root/change_password

GET args:

Argument	Description
old_password	The current password of the currently logged in user.
new_password	The value to change the current user's password to.

AUTHENTICATION: required

RESPONSE description: An empty list. Will set a new sessionid cookie which can then be sent with subsequent API calls. Incorrect old_password results in a 403 FORBIDDEN response.

COMMENTS: Arguments can be sent as either GET or POST parameters.

POSSIBLE FUTURE DIRECTIONS: Input arguments should only be accepted as POST parameters. Using GET parameters leaks passwords into the Apache log, which is a security risk.

2 Available dashboard views

Dashboard views define an hierarchical navigation structure of pages within a dashboard application. Each view typically defines a page in a dashboard application. There may be multiple top-level views which may be selected from a top level menu. A view may have subviews which can be navigated to directly from a subview, or from a widget. A canonical navigation hierarchy is maintained for bread-crumbing, etc. A view may have additional properties that are meaningful in the context of a particular dashboard application.

2.1 Get Top Level Views

URL: *api-root*/top_level_views

GET args: None

AUTHENTICATION: May be required, depending on system configuration.

REQUEST example: *apt-root*/top_level_views
RESPONSE description: A list of View Descriptions for the top-level views the user has access to (or that are publicly accessible if not authenticated). A View Description is an object with the following keys:

Key	Type	Description
name	string	A display name for the view. Used in bread crumbs and menus.
label	string	A short “slug” label for the theme, for use in API call parameters.

RESPONSE example:

```
[
	{
		"name": "General",
		"label": "general"
	},
	{
		"name": "Air and Water",
		"label": "environment",
	},
	{
		"name": "Roads, Rail and Ports",
		"label": "infrastructure",
	},
	...
]
```

3 Get Icon Libraries

URL: *api-root*/icons

AUTHENTICATION: May be required, depending on system configuration.

GET args: None

REQUEST example: *apt-root*/icons

RESPONSE description: An object, where each key represents an icon library. Each key is a short “slug” string used to identify the icon library elsewhere in the API. The value for each key is a list of Icon Definitions. An Icon Definition is an object containing the following keys:

Key	Type	Description
library	string	The slug name identifying the icon library. Should be the same for every icon in a given library, and should match the key in the object holding all icon libraries.
value	string	A short “slug” label identifying the icon elsewhere in the API.
alt_text	string	The alt text to be displayed with the icon image.

RESPONSE example:

```
{
    "weather_icon_scale": [
        {
            "alt_text": "Sunny",
            "value": "sunny",
            "library": "weather_icon_scale"
        },
        {
            "alt_text": "Clear",
            "value": "clear",
            "library": "weather_icon_scale"
        },
        …
      ],
      ….
}
```

4 Get View

URL: *api-root*/view/<view_label>

AUTHENTICATION: May be required, depending on system configuration and choice of view.

REQUEST example: *apt-root*/view/air_pollution

RESPONSE DESCRIPTION: A View Definition, which is an object with the following keys:

Key	Type	Description
crumbs	List of View Description Objects	A list of View Descriptions (see above) tracing the navigation hierarchy from a top-level view to the current view. The first View Description in the list represents the top level view of the hierarchy, the last represents the requested view – the view being defined.
children	List of View Descriptions	The child views of this view.
type	string	The name of the view type. Typically used by the front-end to select between page layout templates.
show_children	boolean	If true, then a menu should be provided for direct navigation to child views (if any exist). If false, then child views can only be navigated to via widgets. Guaranteed to be consistent across all views of a given type.
properties	object	An associative array of key-value pairs for additional implementation-specific properties of the view.
widgets	List of Widget Definitions	A list of objects defining the widgets to be displayed in this view, sorted by display precedence of the category and subcategory, and sorted in order of precedence for display within each subcategory.

The category and subcategory can be used by front ends to lay out widgets in related groups. Some categories can may have special significance to a particular front end implementation. For example the “General”/“Baseline” subcategory was used in the NSW Premier's dashboard prototype for widgets displayed in a specially formatted manner at the bottom of the dashboard (Weather, Events and News feed).

4.1 Widget and Tile Definitions

A Widget Definition is an object with the following keys:

Key	Type	Description
name	string	The display name of the widget
subtitle	string	The display subtitle of the widget (can be null)
category	string	The category of the widget. (Particular front end implementations have used this for determining colour pallet for the widget; or the column in which the widget is displayed.)
category_aspect	integer	Used by some front-end implementations to determine the relative width of widgets in a particular category (e.g. when category determines display column.) Guaranteed to be constant within a single category.
subcategory	string	The subcategory of the widget.
about	string	HTML formatted text explaining the data contained in the widget – for direct display to users.
actual_frequency	string	The "actual frequency" of the data displayed in the widget. This is the text currently displayed in the bottom right-hand corner of the widget.
refresh_rate	integer	How often the front end should requery the API for updated data for the widget. In seconds.
label	string	A short "slug" label for the widget, for use in API call parameters.
source_url	string	An external URL link, to which the user can be directed for further information about the data presented in the widget. Should be a website operated by the agency supplying the data.
source_url_text	string	The text to display in the hyperlink pointing to source_url.
child_view	string	(Optional) The label of a view which can be navigated to through this widget. Would typically be a child view of the containing view, or at least in the same navigation hierarchy, but this is not required.
child_view_text	string	(Optional) The text to display in the hyperlink pointing to the child view.
display	Object (Display Definition)	Description of the format of the data for the widget and how it is to be presented.

4.1.1 Display Definition

A Display Definition is an object with the following keys:

Key	Type	Description
expansion_hint	string	The text to be displayed on the control to expand the widget. Will be null if no expansion tiles.
deexpansion_hint	string	The text to be displayed on the control to collapse the expanded widget. Will be null if no expansion tiles.
tiles	List of Tile Definition objects.	A list of tiles. A widget must have at least one default (non-expansion) tile, and may have an expansion tile as well. The default tiles are always listed first. (Some front-end implementations may expect additional restrictions, e.g. that there is always one and only one default tile.)
raw_data_sets	List of Raw Data Set Definition objects	A list of raw data sets (CSV files) available for download for the widget. Optional – not supplied if no raw data sets defined.

4.1.2 Raw Data Set Definition

A Raw Data Set Definition is an object with the following keys:

Key	Type	Description
label	string	A short "slug" label for the raw data set, for use in API call parameters.
columns	List of Column Definition objects	List of columns that appear in the raw data CSV file, in the order in which they appear. Each column definition includes a "heading" (which is the exact string that will appear in the first (heading) row for that column) and may optionally include a "description" which is a longer string explaining the contents for the column.

4.1.3 Tile Definition

A Tile Definition is an object with the following required keys:

Key	Type	Description
type	string	The type of the tile. Supported types are listed below. The Tile Definition may have other additional keys, depending on the tile type. This is also described below.
expansion	boolean	True for the expansion tile, false for the default (non-expansion) tile.
aspect	integer	Implementation-specific field indicating the relative width of the tile.

The allowed tile types are:

Display Tile type	**Description**
single_main_stat	A tile with a single main statistic shown, and optionally one or more secondary statistics.
double_main_stat	A tile with two statistics shown with equal weighting. Only used on default tiles currently.
text_template	A tile containing a text template into which statistic data can be inserted. E.g. A sentence like “Of ***12*** services from ***5*** agencies, representing ***65%*** of total traffic, ***85%*** are online.” where the text in bold italics are statistic data.
single_list_stat	A tile containing a single statistic, which must be a display list statistic (defined below under statistic definition). Mostly used for expansion tiles, but can appear in default tiles as well.
multi_list_stat	A tile containing at least one and up to four display list statistics. It may also contain a single non-display-list statistic, which must be the first statistic listed if present.
priority_list	A list of statistics, to be displayed in the order in which they were defined.
urgency_list	A list of statistics, ordered by traffic light value, e.g. bad/red statistics first.
list_overflow	If the default tile is of type priority_list, urgency_list, newsfeed or single_list_stat, then the expansion tile can be of type “list_overflow”. The tile should then be displayed with all list items from the default tile list that didn't fit on the default tile.
calendar	A calendar tile – may only contain one statistic which must be of event_list type.
graph	A graph is rendered in the tile.
grid	Statistics are displayed in a rectangular grid.
graph_single_stat	A graph is rendered in the tile, along with one non-display-list statistic.
grid_single_stat	Statistics are displayed in a rectangular grid. An additional non-display-list statistic is displayed outside the grid.
newsfeed	A tile containing a single statistic of type string_kv_list.
news_ticker	A tile containing a single statistic of type string_list. Used for a scrolling news feed.
tag_cloud	A tile containing a tag-cloud. Contains a single numeric_kv_list statistic, the keys of which represent the tags to be displayed and the value of which represents the relative size of the tags.
time_line	A tile containing a time-line. Contains a single hierarchical_event_list statistic.
map	A map is rendered in the tile.

Additional keys for the TileDefinition based on tile type are as follows:

Tile Type(s)	Key	Type	Description
All except: graph, grid, list_overflow, map	statistics	List of Statistic Definition objects.	List of statistics to be displayed in the tile.
single_list_stat, multi_list_stat, priority_list, urgency_list	list_label_width	integer	How wide the label column of the list should be. (as the percentage of the width of the whole tile.)
text_template	template	string	The text template. Statistic values are represented in the template by "%{statistic_url}".
grid, grid_single_stat	grid	Object (Grid Definition)	Description of how the grid is to be displayed, and what data is to be displayed in it.
graph, graph_single_stat	graph	Graph Definition Object	Describes what data is to be plotted on the graph, and how it is to be displayed.
map	map	Widget Map Definition object	Describes what geoapatial data is to be plotted on the map.

4.2 Widget Map Definitions

A Widget Map Definition is an object with the following keys:

Key	Type	Description
layers	List of Map Layer Definition Objects	Map feature layers to be displayed on the map.
window	Geo-Window Definition Object	The bounding rectangle of the map.
label	string	A short "slug" label for the graph, used in the Get Graph API call response.

4.3 Graph Definitions

A Graph Definition is an object with the following keys:

Key	Type	Description
graph_type	string	The type of the tile. Supported types are listed below. The Graph Definition may have other additional keys, depending on the tile type. This is also described below.
heading	string	Text to be displayed above (or below) the graph as an overall heading for the graph. (May be null)
label	string	A short "slug" label for the graph, used in the Get Graph API call response.
display_options	Display Option Object	Display options. Contents depend on graph type, as described below.

The allowed graph types, and the additional keys in the Graph object associated with them, are:

graph_type	description	Additional Graph Definition Keys	descriptions
line	A line (or multi-line) graph.	vertical_axis	The vertical axis of a line graph is assumed to be numeric. This option is an object containing an optional label for the axis (key "label", a string value), and whether the zero point must be displayed (key "always_show_zero", a boolean value). (e.g. if the values are in the range 12-15, can the vertical axis simply span the values 12-15 (false), or should it run from 0-15 (true).)
		secondary_vertical_axis	Optional. Specifies a secondary vertical axis that some dataset(s) are to be plotted against. Same format as vertical_axis.
		horizontal_axis	The horizontal axis of a line graph has a type (key "type", string. Allowed values: "numeric", "date", "time", "datetime") and an optional label (key "label", a string).
		lines	A list of Graph Dataset Definition objects, defining the lines (datasets) to be plotted on the graph.
histogram	A (vertical) histogram or clustered histogram.	numeric_axis	Describes the numeric (vertical) axis of the histogram. Uses the same format as the "vertical_axis" key for line type graphs, as described above.
		secondary_numeric_axis	Optional. Specifies a secondary numeric (vertical) axis that some dataset(s) are to be plotted against. Same format as numeric_axis.
		clusters	A list of Graph Cluster Definition objects, representing the histogram clusters.

graph_type	description	Additional Graph Definition Keys	descriptions
		bars	A list of Graph Dataset Definition objects representing the individual bars in each cluster.
bar	As for histogram, but displayed horizontally.	As for histogram.	
pie	A pie chart, or set of related pie charts.	pies	A list of Graph Cluster Definition objects representing the separate pies in a multiple pie chart (contains a single entry for a single pie chart). As for clusters in a bar/histogram graph.
		sectors	A list of Graph Dataset Definition objects representing the pie sectors in each pie. As for bars (in histogram and bar type graphs) and lines (in line type graphs) above.

A Graph Display Option object contains keys specific to the graph type, as follows:

graph_type	Option	Type	Description
line	lines	string	How the lines joining the datapoints are to be displayed. Allowed values are: straight: Datapoints should be joined by (jagged) straight lines. bezier: Datapoints should be joined by smooth curves. none: Datapoints should not be joined.
	points	string	How the individual datapoints are to be displayed: none: Individual datapoints should not be displayed. (NB. If both “lines” and “points” are “none” the graph will always be blank!) triangle/circle/square: Individual datapoints should be displayed as small instances of the respective shape. vertical-bar: Each datapoint is displayed as a thin vertical bar (similar to a histogram).
	shaded	boolean	If true, area under the line is to be coloured in. Cannot be true if lines is “none”. Shading of lines goes in order lines are defined (i.e. largest dataset should be defined first, so it can be shaded first with subsequent smaller datasets shaded on top.)

graph_type	Option	Type	Description
	single_graph	boolean	If true, display on a single graph. If false, display as a series of smaller individual graphs. N.B. Single-graph true is incompatible with vertical-bar points.
	rotates	boolean	Can only be true if single_graph is false. Display rotates through the individual graphs, one at a time.
	point_colour_map	Colour Map Definition Object	Indicates how points should be coloured, based on their vertical axis value. Optional. If null, then use the colours in the Graph Dataset Definition. N.B. Must be null if “points” is “none”.
bar/histogram	stacked	boolean	If true, should be rendered as a stacked bar/histogram chart. (i.e. the bars for each cluster are stacked on top of each other instead of being placed side by side.
	single_graph	boolean	If true, display on a single graph. If false, display as a series of smaller individual graphs.
	rotates	boolean	Can only be true if single_graph is false. Display rotates through the individual graphs, one at a time.
pie	-		*No special display-options for pie graphs have been identified yet. The Display Option object will be empty.*

4.3.1 Colour Map Definition Objects

A Colour Map Definition is an object with the following keys:

Key	Type	Description
name	string	A name identifying the colour map. (Can be used by the front end to map colour hints to a designer-specified palette.)
map	List of Colour Mapping Objects	Each Colour Mapping object contains two keys: “min_value” (numeric): The minimum vertical axis value for which this colour should be used. “colour” (string): A colour hint for how to display datapoints in the relevant range. The list of colour mapping objects will be sorted in ascending order of min_value. The first colour mapping object in the list may have a null min_value.

A Graph Dataset Definition is an object with the following keys:

Key	Type	Description
label	string	A short “slug” label for the dataset, used in the Get Graph Data API method.
name	string	The display name for the dataset.
colour	string	Non-binding colour hint for the graph element representing the dataset.
hyperlink	string	An external URL to link for this dataset (optional - may be null).
use_secondary_vertical_axis	boolean	If true, the dataset is to be plotted against the secondary vertical axis. N.B. This key is called “use_secondary_numeric_axis” for graphs of type “histogram” and “bar”.

A Graph Cluster Definition is an object with the following keys:

Key	Type	Description
label	string	A short “slug” label for the cluster, used in the Get Graph Data API method.
name	string	The display name for the dataset.
hyperlink	string	An external URL to link for this dataset (optional - may be null).

4.4 Grid Definitions

A Grid Definition is an object with the following keys:

Key	Type	Description
corner_header	string	Heading for the top left corner of the graph (where the heading column and heading row intersect). Can be null. Can only be not null if both show_column_headers and show_row_headers are both true.
show_column_headers	boolean	If true a heading row is displayed on the top of the grid, containing the labels for each column.
show_row_headers	boolean	If true a heading column is displayed on the left of the grid, containing the labels for each row.
columns	List of Column Definition objects	Descriptions of the columns to be displayed in the grid, listed in display order from left to right.
rows	List of Row Definition objects	Descriptions of the rows to be displayed in the grid, listed in display order from top to bottom.

A Column Definition is an object with the following key:

Key	Type	Description
header	string	The heading for the column.

A Row Definition is an object with the following keys:

Key	Type	Description
header	string	The heading for the row.
statistics	list of Statistic Definition objects	A description of the statistics to be displayed in the cells of this row. The statistics are listed in display order (left to right) and the number of statistics will match the number of columns for the grid. Statistics in grids should not be display list statistics.

4.5 Statistic Definitions

A Statistic Definition is an object containing the following keys:

Key	When	Type	Description
label	Always	string	A short “slug” label for the statistic, for use in the widget data API.
type	Always	string	The type of statistic. Allowed values are listed below.
name	For all tile types except “grid”.	string	The name of the statistic. Usually for display (see below). May be blank if the statistic is to be unlabeled or name_as_label is false.
display_name	For all tile types except “grid”.	boolean	If true, the name above is the display label for the statistic. If false, the statistic will be supplied with a dynamic display label in the widget data API
precision	For statistics of type “numeric” and “numeric_kv_list”.	Integer	The number of decimal places the data for the statistic will have. E.g. 0 means an integer. 1 means 0.1 or 124.5. 2 means 13.44, etc.
unit	For statistics of type “numeric” and “numeric_kv_list”.	Object (Unit Definition)	How the number is to be displayed.
traffic_light_scale	Always	Object (Traffic Light Definition)	The traffic light scale (colour coding) to use for the statistic. Null if no traffic light scale applies for the statistic.
trend	For statistic types except “string_list” and “long_string_list”.	boolean	If true, an up/down/steady trend arrow is to be displayed with the statistic.
icon_library	For statistic types except “string_list” and “long_string_list”.	string	If not null, the library name of an icon library. The icon library is described by the Get Icon Libraries API call.

Key	When	Type	Description
rotates	Only displayed if true.	boolean	For display list statistics, “rotates”=true means that only as many list items as can be displayed at once will be displayed, but the items displayed will be gradually rotated through the full set of list items supplied. For non-display-list statistics, “rotates”=true means that a list of data items will be supplied, but only one will be displayed at a time. The displayed item gradually rotates through the full set supplied.
hyperlinkable	For display list statistics, and statistics with rotates=true	boolean	If true, an external URL can be optionally supplied for each list item.
numbered_list	For display list statistics.	Boolean	If true, list items should be displayed numbered.
footer	Always	boolean	If true, the statistic is to be displayed as a “footer” across the bottom of the tile. Cannot be true for display list statistics, and can only be true for at most one statistic per tile.

Allowed statistic types are listed below, showing which statistic types constitute “display list statistics”:

Statistic Type	Description
Not display list statistics	
string	A (short) string value.
long_string	A long string value
numeric	A number.
am_pm	An am-pm indicator.
Display list statistics	
string_kv_list	A list of key value pairs where the key and value are both strings.
numeric_kv_list	A list of key value pairs where the key is a string and the value is numeric. Statistics of this type should also define a unit and a precision, which apply to all values in the list.
string_list	A list of strings.
long_string_list	A list of long strings
event_list	A list of key value pairs where the key is a date an the value is a string. Can only appear in a calendar type tile.

Statistic Type	Description
hierarchical_event_list	A list of key value pairs where the key consists of a datetime and a “level” (i.e. second, minute, hour,day, month,quarter,year); and the value is a string. Can only appear in a time_line type tile. The datetime value of each list item is rounded to the associated level.

4.5.1 Unit Definitions

A Unit Definition is an object that may contain the following keys:

Key	Type	Description
prefix	string	Text to be displayed BEFORE the numeric value, e.g. “$” in “$34.55”. Only present if there is a prefix.
suffix	string	Text to be displayed AFTER the numeric value, e.g. “%” in “4.5%”. Only present if there is a suffix
underfix	string	Text to be displayed UNDER the numeric value. Only present if there is an underfix.
signed	boolean	Only present if true. If true, a “+” sign is displayed when the value of the statistic data is positive. The “+” sign appears before any prefix. Note that a “-” sign is always displayed if the value of the statistic data is negative – the value of “signed” key in the unit does not effect this.
si_prefix_rounding	integer	Only present if non-zero. If specified, the number supplied by the API (with precision as specified) should be rounded by the front-end to the indicated number of significant digits, and an SI unit prefix will be displayed between the number and any defined unit suffix. Eg 1. Precision=0, si_prefix_rounding=2 1→ 1, 234 → 230, 5328 → 5.3k, 45236457 → 45M Eg 2. Precision=2, si_prefix_rounding=4 0.03 → 30m, 12.34→12.34, 234.77→234.8, 2345156.43→23.45M

Note that all keys in a Unit Definition are optional. An empty object is a valid Unit Definition.

4.5.2 Traffic Light Definitions

A Traffic Light Definition is an object with the following keys:

Key	Type	Description
scale	string	The name of the traffic light scale.
codes	List of Traffic Light Code Definition objects	List of the possible colour codes within the scale, listed in order from “best” to “worst”. (This order is significant for urgency_list tiles.)

A Traffic Light Code Definition is an object with the following keys:

Key	Type	Description
colour	string	Non-binding colour hint for the traffic light code.
value	string	A short “slug” value used to identify the traffic light code in the widget data API.

RESPONSE example:

```
{
    "show_children": false,
    "properties": {
        "theme": "All",
        "frequency": "Current",
        "location": "Sydney"
    },
    "crumbs": [
        {
            "name": "All",
            "label": "tall_migration"
        },
        {
            "name": "Current",
            "label": "tall_frt_migration"
        },
        {
            "name": "Sydney",
            "label": "tall_frt_lsyd_migration"
        }
    ],
    "widgets": [
        {
            "subtitle": null,
            "subcategory": "Baseline",
            "source_url": "http://www.abc.net.au/news/nsw",
            "actual_frequency": "Real time",
            "category": "General",
            "about": "<p>News Headlines from ABC news RSS feed for NSW.</p>",
            "category_aspect": 1,
            "name": "News Headlines",
            "refresh_rate": 320,
            "url": "news",
            "source_url_text": "ABC News",
            "display": {
                "tiles": [
                    {
                        "statistics": [
                            {
                                "display_name": true,
                                "name": "",
                                "numbered_list": false,
                                "hyperlinkable": true,
                                "footer": false,
                                "label": "headlines",
                                "type": "string_list"
                            }
                        ],
                        "type": "news_ticker",
                        "aspect": 1,
                        "expansion": false
                    }
                ],
                "deexpansion_hint": "Show less",
                "expansion_hint": "Show more"
            }
        },
        {
            "subtitle": null,
            "subcategory": "Baseline",
            "source_url": "http://www.bom.gov.au/",
            "actual_frequency": "Real time",
            "category": "General",
            "about": "<p>All weather observations and forecast data are for the
Sydney Observatory weather station and are sourced directly from the Bureau of
Meteorology open data API.</p>\r\n<p>The seasonal average trend is obtained by
comparing the current temperature to the long term (e.g. over all years since
records begain) average temperature for the month.</p>",
            "category_aspect": 1,
            "name": "Weather",
            "refresh_rate": 300,
            "url": "weather",
            "source_url_text": "Bureau of Meterology website",
            "display": {
                "tiles": [
                    {
                        "statistics": [
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "Now",
                                "footer": false,
                                "trend": false,
                                "precision": 1,
                                "label": "current_temp",
                                "type": "numeric",
                                "unit": {
                                    "suffix": "\u00b0C"
                                }
                            },
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "Seasonal average",
                                "footer": false,
                                "trend": true,
                                "precision": 1,
                                "label": "seasonal_average",
                                "type": "numeric",
                                "unit": {
                                    "suffix": "\u00b0C",
                                    "signed": true
                                }
                            },
                            {
                                "icon_library": "weather_icon_scale",
                                "display_name": true,
                "name": "Today",
                "footer": false,
                "trend": false,
                "label": "today_short_forecast",
                "type": "string"
            },
            {
                "icon_library": null,
                "display_name": true,
                "name": "Max",
                "footer": false,
                "trend": false,
                "precision": 0,
                "label": "today_max",
                "type": "numeric",
                "unit": {
                    "suffix": "\u00b0C"
                }
            },
            {
                "icon_library": null,
                "display_name": true,
                "name": "Min",
                "footer": false,
                "trend": false,
                "precision": 0,
                "label": "today_min",
                "type": "numeric",
                "unit": {
                    "suffix": "\u00b0C"
                }
            }
        ],
        "type": "single_main_stat",
        "aspect": 1,
        "expansion": false
    },
    {
        "statistics": [
            {
                "icon_library": "weather_icon_scale",
                "display_name": true,
                "name": "Today",
                "footer": false,
                "trend": false,
                "label": "today_long_forecast",
                "type": "string"
            },
            {
                "icon_library": "weather_icon_scale",
                "display_name": false,
                "name": "Day 1 Forecast",
                "footer": false,
                "trend": false,
                "label": "day_1_forecast",
                "type": "string"
            },
            {
                "icon_library": null,
                "display_name": false,
                "name": "Day 1 Max",
                "footer": false,
                "trend": false,
                "precision": 0,
            "label": "day_1_max",
            "type": "numeric",
            "unit": {
                "suffix": "\u00b0C"
            }
        },
        {
            "icon_library": null,
            "display_name": false,
            "name": "Day 1 Min",
            "footer": false,
            "trend": false,
            "precision": 0,
            "label": "day_1_min",
            "type": "numeric",
            "unit": {
                "suffix": "\u00b0C"
            }
        },
        {
            "icon_library": "weather_icon_scale",
            "display_name": false,
            "name": "Day 2 Forecast",
            "footer": false,
            "trend": false,
            "label": "day_2_forecast",
            "type": "string"
        },
        {
            "icon_library": null,
            "display_name": false,
            "name": "Day 2 Max",
            "footer": false,
            "trend": false,
            "precision": 0,
            "label": "day_2_max",
            "type": "numeric",
            "unit": {
                "suffix": "\u00b0C"
            }
        },
        {
            "icon_library": null,
            "display_name": false,
            "name": "Day 2 Min",
            "footer": false,
            "trend": false,
            "precision": 0,
            "label": "day_2_min",
            "type": "numeric",
            "unit": {
                "suffix": "\u00b0C"
            }
        },
        {
            "icon_library": "weather_icon_scale",
            "display_name": false,
            "name": "Day 3 Forecast",
            "footer": false,
            "trend": false,
            "label": "day_3_forecast",
            "type": "string"
        },
        {
```

```
                                "icon_library": null,
                                "display_name": false,
                                "name": "Day 3 Max",
                                "footer": false,
                                "trend": false,
                                "precision": 0,
                                "label": "day_3_max",
                                "type": "numeric",
                                "unit": {
                                    "suffix": "\u00b0C"
                                }
                            },
                            {
                                "icon_library": null,
                                "display_name": false,
                                "name": "Day 3 Min",
                                "footer": false,
                                "trend": false,
                                "precision": 0,
                                "label": "day_3_min",
                                "type": "numeric",
                                "unit": {
                                    "suffix": "\u00b0C"
                                }
                            }
                        ],
                        "list_label_width": 50,
                        "expansion": true,
                        "aspect": 1,
                        "type": "priority_list",
                        "columns": 1
                    }
                ],
                "deexpansion_hint": "Show less",
                "expansion_hint": "Detailed weather"
            }
        },
        {
            "subtitle": null,
            "subcategory": "Baseline",
            "source_url": "http://www.events.nsw.gov.au/",
            "actual_frequency": "Daily",
            "category": "General",
            "about": "<p>The event calendar is manually maintained. Todays
events are automatically updated daily from the manually maintained
calendar.</p>",
            "category_aspect": 1,
            "name": "Today's Events",
            "refresh_rate": 7200,
            "url": "events",
            "source_url_text": "Events NSW",
            "display": {
                "tiles": [
                    {
                        "statistics": [
                            {
                                "display_name": true,
                                "name": "Today",
                                "numbered_list": false,
                                "hyperlinkable": true,
                                "footer": false,
                                "label": "today",
                                "type": "string_list"
                            }
                        ],
                        "type": "single_list_stat",
                        "aspect": 1,
                        "expansion": false,
                        "list_label_width": 100
                    },
                    {
                        "statistics": [
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "Event Calendar",
                                "numbered_list": false,
                                "hyperlinkable": true,
                                "trend": false,
                                "footer": false,
                                "label": "calendar",
                                "type": "event_list"
                            }
                        ],
                        "type": "calendar",
                        "aspect": 1,
                        "expansion": true
                    }
                ],
                "deexpansion_hint": "Show less",
                "expansion_hint": "See all"
            }
        },
        {
            "subtitle": "Arrive/Depart Central on time",
            "subcategory": "Public Transport",
            "source_url": "http://www.sydneytrains.info/service_updates/",
            "actual_frequency": "Sample Data",
            "category": "Service Delivery",
            "about": "<p>Manually entered daily summary data supplied by
Transport NSW</p>",
            "category_aspect": 1,
            "name": "Trains",
            "refresh_rate": 3600,
            "url": "train_arrivals",
            "source_url_text": "Sydney Trains service updates",
            "display": {
                "tiles": [
                    {
                        "statistics": [
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "",
                                "footer": false,
                                "trend": true,
                                "precision": 0,
                                "label": "all_on_time",
                                "type": "numeric",
                                "unit": {
                                    "suffix": "%"
                                }
                            },
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "YTD",
                                "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "all_on_time_ytd",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                },
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "Target",
                    "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "all_on_time_target",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                }
            ],
            "type": "single_main_stat",
            "aspect": 1,
            "expansion": false
        },
        {
            "statistics": [
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "T1 North Shore, Northern & Western",
                    "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "t1_ontime",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                },
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "T2 Inner West & South",
                    "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "t2_ontime",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                },
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "T3 Bankstown",
                    "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "t3_ontime",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                },
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "T4 Eastern Suburbs & Illawarra",
                    "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "t4_ontime",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                },
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "T5 Cumberland",
                    "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "t5_ontime",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                },
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "T6 Carlingford",
                    "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "t6_ontime",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                },
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "T7 Olympic Park",
                    "footer": false,
                    "trend": false,
                    "precision": 0,
                    "label": "t7_online",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                }
            ],
            "list_label_width": 70,
            "expansion": true,
            "aspect": 1,
            "type": "priority_list",
            "columns": 1
        }
    ],
    "deexpansion_hint": "Show less",
                "expansion_hint": "Show more"
            }
        },
        {
            "subtitle": null,
            "subcategory": "Public Transport",
            "source_url": "http://www.sydneytrains.info/rss/",
            "actual_frequency": "Real time",
            "category": "Service Delivery",
            "about": "<p>Train service delays, cancelations and trackwork, as
reported by Sydney Trains RSS feed.</p>\r\n<p>Includes Sydney suburban services
only (T1-T7 plus South West Rail Link).</p>",
            "category_aspect": 1,
            "name": "Train Line Incidents",
            "refresh_rate": 100,
            "url": "train_service_interrupt",
            "source_url_text": "Sydney Trains RSS feeds",
            "display": {
                "tiles": [
                    {
                        "statistics": [
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "Delays on",
                                "footer": false,
                                "trend": false,
                                "precision": 0,
                                "label": "delays",
                                "traffic_light_scale": {
                                    "scale": "std 3 code",
                                    "codes": [
                                        {
                                            "colour": "green",
                                            "value": "good"
                                        },
                                        {
                                            "colour": "amber",
                                            "value": "poor"
                                        },
                                        {
                                            "colour": "red",
                                            "value": "bad"
                                        }
                                    ]
                                },
                                "type": "numeric",
                                "unit": {
                                    "suffix": " lines"
                                }
                            },
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "Cancellations",
                                "footer": false,
                                "trend": false,
                                "precision": 0,
                                "label": "cancellations",
                                "traffic_light_scale": {
                                    "scale": "std 3 code",
                                    "codes": [
                                        {
                                            "colour": "green",
                "value": "good"
            },
            {
                "colour": "amber",
                "value": "poor"
            },
            {
                "colour": "red",
                "value": "bad"
            }
        ]
    },
    "type": "numeric",
    "unit": {
        "suffix": " lines"
    }
},
{
    "icon_library": null,
    "display_name": true,
    "name": "Trackwork on",
    "footer": false,
    "trend": false,
    "precision": 0,
    "label": "current_trackwork",
    "traffic_light_scale": {
        "scale": "std 3 code",
        "codes": [
            {
                "colour": "green",
                "value": "good"
            },
            {
                "colour": "amber",
                "value": "poor"
            },
            {
                "colour": "red",
                "value": "bad"
            }
        ]
    },
    "type": "numeric",
    "unit": {
        "suffix": " lines"
    }
},
{
    "icon_library": null,
    "display_name": true,
    "name": "Overnight work on",
    "footer": false,
    "trend": false,
    "precision": 0,
    "label": "sched_overnight_trackwork",
    "traffic_light_scale": {
        "scale": "std 3 code",
        "codes": [
            {
                "colour": "green",
                "value": "good"
            },
            {
                "colour": "amber",
                                        "value": "poor"
                                    },
                                    {
                                        "colour": "red",
                                        "value": "bad"
                                    }
                                ]
                            },
                            "type": "numeric",
                            "unit": {
                                "suffix": " lines"
                            }
                        }
                    ],
                    "list_label_width": 60,
                    "expansion": false,
                    "aspect": 1,
                    "type": "priority_list",
                    "columns": 1
                },
                {
                    "statistics": [
                        {
                            "display_name": true,
                            "name": "",
                            "numbered_list": false,
                            "hyperlinkable": true,
                            "footer": false,
                            "label": "interruptions",
                            "traffic_light_scale": {
                                "scale": "std 3 code",
                                "codes": [
                                    {
                                        "colour": "green",
                                        "value": "good"
                                    },
                                    {
                                        "colour": "amber",
                                        "value": "poor"
                                    },
                                    {
                                        "colour": "red",
                                        "value": "bad"
                                    }
                                ]
                            },
                            "type": "string_list"
                        }
                    ],
                    "type": "single_list_stat",
                    "aspect": 1,
                    "expansion": true,
                    "list_label_width": 100
                }
            ],
            "deexpansion_hint": "Show less",
            "expansion_hint": "Show more"
        }
    },
    {
        "subtitle": null,
        "subcategory": "Warnings",
        "source_url": "http://www.rfs.nsw.gov.au/fire-information/fdr-and-
tobans",
            "actual_frequency": "Daily",
            "category": "Public Safety",
            "about": "<p>Data taken from NSW Rural Fire Services.</p>",
            "category_aspect": 1,
            "name": "Fire Danger Ratings",
            "refresh_rate": 1000,
            "url": "fire",
            "source_url_text": "NSW Rural Fire Service fire danger ratings and
fire bans",
            "display": {
                "tiles": [
                    {
                        "statistics": [
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "",
                                "numbered_list": false,
                                "hyperlinkable": false,
                                "trend": false,
                                "footer": false,
                                "label": "rating_list_main",
                                "traffic_light_scale": {
                                    "scale": "fire_danger_ratings",
                                    "codes": [
                                        {
                                            "colour": "white",
                                            "value": "none"
                                        },
                                        {
                                            "colour": "green",
                                            "value": "low_moderate"
                                        },
                                        {
                                            "colour": "blue",
                                            "value": "high"
                                        },
                                        {
                                            "colour": "yellow",
                                            "value": "very_high"
                                        },
                                        {
                                            "colour": "orange",
                                            "value": "severe"
                                        },
                                        {
                                            "colour": "red",
                                            "value": "extreme"
                                        },
                                        {
                                            "colour": "red-black",
                                            "value": "catastrophic"
                                        }
                                    ]
                                },
                                "rotates": true,
                                "type": "string_kv_list"
                            }
                        ],
                        "type": "single_list_stat",
                        "aspect": 0,
                        "expansion": false,
                        "list_label_width": 50
                    },
                {
                    "statistics": [
                        {
                            "icon_library": null,
                            "display_name": true,
                            "name": "",
                            "numbered_list": false,
                            "hyperlinkable": false,
                            "trend": false,
                            "footer": false,
                            "label": "rating_list_expansion",
                            "traffic_light_scale": {
                                "scale": "fire_danger_ratings",
                                "codes": [
                                    {
                                        "colour": "white",
                                        "value": "none"
                                    },
                                    {
                                        "colour": "green",
                                        "value": "low_moderate"
                                    },
                                    {
                                        "colour": "blue",
                                        "value": "high"
                                    },
                                    {
                                        "colour": "yellow",
                                        "value": "very_high"
                                    },
                                    {
                                        "colour": "orange",
                                        "value": "severe"
                                    },
                                    {
                                        "colour": "red",
                                        "value": "extreme"
                                    },
                                    {
                                        "colour": "red-black",
                                        "value": "catastrophic"
                                    }
                                ]
                            },
                            "type": "string_kv_list"
                        }
                    ],
                    "type": "single_list_stat",
                    "aspect": 1,
                    "expansion": true,
                    "list_label_width": 50
                }
            ],
            "deexpansion_hint": null,
            "expansion_hint": "Show more"
        }
    },
    {
        "subtitle": null,
        "subcategory": "Market",
        "source_url": "http://www.asx.com.au/",
        "actual_frequency": "Real time",
        "category": "Economic",
        "about": "<p>20 minute old data scraped from Yahoo! Finance.</p>",
"category_aspect": 1,
"name": "ASX",
"refresh_rate": 120,
"url": "asx",
"source_url_text": "ASX web site",
"display": {
    "tiles": [
        {
            "statistics": [
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "",
                    "footer": false,
                    "trend": true,
                    "precision": 2,
                    "label": "index",
                    "type": "numeric",
                    "unit": {
                        "underfix": "ASX-200"
                    }
                },
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "Today",
                    "footer": false,
                    "trend": false,
                    "precision": 2,
                    "label": "movement",
                    "type": "numeric",
                    "unit": {
                        "suffix": "%",
                        "signed": true
                    }
                },
                {
                    "icon_library": null,
                    "display_name": false,
                    "name": "Stock Prices",
                    "hyperlinkable": false,
                    "trend": true,
                    "footer": true,
                    "precision": 2,
                    "label": "stock_prices",
                    "rotates": true,
                    "type": "numeric",
                    "unit": {}
                }
            ],
            "type": "single_main_stat",
            "aspect": 1,
            "expansion": false
        },
        {
            "statistics": [
                {
                    "icon_library": null,
                    "display_name": true,
                    "name": "Today's Min",
                    "footer": false,
                    "trend": false,
                    "precision": 2,
                    "label": "today_min",
                                "type": "numeric",
                                "unit": {}
                            },
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "52wk Min",
                                "footer": false,
                                "trend": false,
                                "precision": 2,
                                "label": "yr_min",
                                "type": "numeric",
                                "unit": {}
                            },
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "52wk Max",
                                "footer": false,
                                "trend": false,
                                "precision": 2,
                                "label": "yr_max",
                                "type": "numeric",
                                "unit": {}
                            },
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "Today's max",
                                "footer": false,
                                "trend": false,
                                "precision": 2,
                                "label": "today_max",
                                "type": "numeric",
                                "unit": {}
                            }
                        ],
                        "list_label_width": 50,
                        "expansion": true,
                        "aspect": 1,
                        "type": "priority_list",
                        "columns": 1
                    }
                ],
                "deexpansion_hint": "Show less",
                "expansion_hint": "Show more"
            }
        },
        {
            "subtitle": "Sydney Catchments",
            "subcategory": "Water",
            "source_url": "http://www.sca.nsw.gov.au/water/dam-levels",
            "actual_frequency": "Daily",
            "category": "Environment and Planning",
            "about": "<p>Data taken from the Sydney Catchment
Authority</p>\r\n<p>Key dams are the four largest dams serving Greater
Sydney.</p>",
            "category_aspect": 1,
            "name": "Dam Levels",
            "refresh_rate": 7200,
            "url": "dam",
            "source_url_text": "Sydney Catchment Authority dam levels",
            "display": {
                "tiles": [
{
    "grid": {
        "corner_header": "",
        "show_column_headers": false,
        "rows": [
            {
                "header": "SCA Levels",
                "statistics": [
                    {
                        "icon_library": null,
                        "footer": false,
                        "trend": true,
                        "precision": 1,
                        "label": "all_dams_avg",
                        "traffic_light_scale": {
                            "scale": "std 3 code",
                            "codes": [
                                {
                                    "colour": "green",
                                    "value": "good"
                                },
                                {
                                    "colour": "amber",
                                    "value": "poor"
                                },
                                {
                                    "colour": "red",
                                    "value": "bad"
                                }
                            ]
                        },
                        "type": "numeric",
                        "unit": {
                            "suffix": "%"
                        }
                    }
                ]
            },
            {
                "header": "Last week",
                "statistics": [
                    {
                        "icon_library": null,
                        "footer": false,
                        "trend": false,
                        "precision": 1,
                        "label": "all_dams_last_week",
                        "traffic_light_scale": {
                            "scale": "std 3 code",
                            "codes": [
                                {
                                    "colour": "green",
                                    "value": "good"
                                },
                                {
                                    "colour": "amber",
                                    "value": "poor"
                                },
                                {
                                    "colour": "red",
                                    "value": "bad"
                                }
                            ]
                        },
                    "type": "numeric",
                    "unit": {
                        "suffix": "%"
                    }
                }
            ]
        }
    ],
    "columns": [
        {
            "header": "Dam Levels"
        }
    ],
    "show_row_headers": true
},
"type": "grid",
"aspect": 1,
"expansion": false
```
```
},
{
    "statistics": [
        {
            "icon_library": null,
            "display_name": true,
            "name": "Warragamba",
            "footer": false,
            "trend": true,
            "precision": 1,
            "label": "warragamba",
            "traffic_light_scale": {
                "scale": "std 3 code",
                "codes": [
                    {
                        "colour": "green",
                        "value": "good"
                    },
                    {
                        "colour": "amber",
                        "value": "poor"
                    },
                    {
                        "colour": "red",
                        "value": "bad"
                    }
                ]
            },
            "type": "numeric",
            "unit": {
                "suffix": "%"
            }
        },
        {
            "icon_library": null,
            "display_name": true,
            "name": "Avon",
            "footer": false,
            "trend": true,
            "precision": 1,
            "label": "avon",
            "traffic_light_scale": {
                "scale": "std 3 code",
                "codes": [
                    {
                        "colour": "green",
                    "value": "good"
                },
                {
                    "colour": "amber",
                    "value": "poor"
                },
                {
                    "colour": "red",
                    "value": "bad"
                }
            ]
        },
        "type": "numeric",
        "unit": {
            "suffix": "%"
        }
    },
    {
        "icon_library": null,
        "display_name": true,
        "name": "Cataract",
        "footer": false,
        "trend": true,
        "precision": 1,
        "label": "cataract",
        "traffic_light_scale": {
            "scale": "std 3 code",
            "codes": [
                {
                    "colour": "green",
                    "value": "good"
                },
                {
                    "colour": "amber",
                    "value": "poor"
                },
                {
                    "colour": "red",
                    "value": "bad"
                }
            ]
        },
        "type": "numeric",
        "unit": {
            "suffix": "%"
        }
    },
    {
        "icon_library": null,
        "display_name": true,
        "name": "Cordeaux",
        "footer": false,
        "trend": true,
        "precision": 1,
        "label": "cordeaux",
        "traffic_light_scale": {
            "scale": "std 3 code",
            "codes": [
                {
                    "colour": "green",
                    "value": "good"
                },
                {
                    "colour": "amber",
                                        "value": "poor"
                                    },
                                    {
                                        "colour": "red",
                                        "value": "bad"
                                    }
                                ]
                            },
                            "type": "numeric",
                            "unit": {
                                "suffix": "%"
                            }
                        }
                    ],
                    "list_label_width": 50,
                    "expansion": true,
                    "aspect": 1,
                    "type": "priority_list",
                    "columns": 1
                }
            ],
            "deexpansion_hint": "Show less",
            "expansion_hint": "Show more"
        }
    },
    {
        "subtitle": "Water quality",
        "subcategory": "Environment",
        "source_url": "http://www.environment.nsw.gov.au/beach/",
        "actual_frequency": "Daily",
        "category": "Environment and Planning",
        "about": "<p>Beach quality data taken from the NSW Environment
Beachwatch RSS feeds.</p>",
        "category_aspect": 1,
        "name": "Beaches",
        "refresh_rate": 2000,
        "url": "beaches",
        "source_url_text": "Environment NSW Beachwatch",
        "display": {
            "tiles": [
                {
                    "statistics": [
                        {
                            "icon_library": null,
                            "display_name": true,
                            "name": "",
                            "footer": false,
                            "trend": true,
                            "label": "all_ocean_beaches",
                            "traffic_light_scale": {
                                "scale": "std 3 code",
                                "codes": [
                                    {
                                        "colour": "green",
                                        "value": "good"
                                    },
                                    {
                                        "colour": "amber",
                                        "value": "poor"
                                    },
                                    {
                                        "colour": "red",
                                        "value": "bad"
                                    }
                ]
            },
            "type": "string"
        },
        {
            "icon_library": null,
            "display_name": true,
            "name": "YTD",
            "footer": false,
            "trend": false,
            "label": "all_ocean_ytd",
            "traffic_light_scale": {
                "scale": "std 3 code",
                "codes": [
                    {
                        "colour": "green",
                        "value": "good"
                    },
                    {
                        "colour": "amber",
                        "value": "poor"
                    },
                    {
                        "colour": "red",
                        "value": "bad"
                    }
                ]
            },
            "type": "string"
        },
        {
            "icon_library": null,
            "display_name": false,
            "name": "Highlight Beach",
            "hyperlinkable": false,
            "trend": false,
            "footer": true,
            "label": "highlight_beach",
            "traffic_light_scale": {
                "scale": "std 3 code",
                "codes": [
                    {
                        "colour": "green",
                        "value": "good"
                    },
                    {
                        "colour": "amber",
                        "value": "poor"
                    },
                    {
                        "colour": "red",
                        "value": "bad"
                    }
                ]
            },
            "rotates": true,
            "type": "string"
        }
    ],
    "type": "single_main_stat",
    "aspect": 1,
    "expansion": false
},
{
```

```
"statistics": [
    {
        "icon_library": null,
        "display_name": true,
        "name": "Sydney Ocean Beaches",
        "footer": false,
        "trend": false,
        "label": "region_SYDOC",
        "traffic_light_scale": {
            "scale": "std 3 code",
            "codes": [
                {
                    "colour": "green",
                    "value": "good"
                },
                {
                    "colour": "amber",
                    "value": "poor"
                },
                {
                    "colour": "red",
                    "value": "bad"
                }
            ]
        },
        "type": "string"
    },
    {
        "icon_library": null,
        "display_name": true,
        "name": "Sydney Harbour Beaches",
        "footer": false,
        "trend": false,
        "label": "region_SYDHB",
        "traffic_light_scale": {
            "scale": "std 3 code",
            "codes": [
                {
                    "colour": "green",
                    "value": "good"
                },
                {
                    "colour": "amber",
                    "value": "poor"
                },
                {
                    "colour": "red",
                    "value": "bad"
                }
            ]
        },
        "type": "string"
    },
    {
        "icon_library": null,
        "display_name": true,
        "name": "Southern Harbour Beaches",
        "footer": false,
        "trend": false,
        "label": "region_BOTNY",
        "traffic_light_scale": {
            "scale": "std 3 code",
            "codes": [
                {
                        "colour": "green",
                        "value": "good"
                    },
                    {
                        "colour": "amber",
                        "value": "poor"
                    },
                    {
                        "colour": "red",
                        "value": "bad"
                    }
                ]
            },
            "type": "string"
        },
        {
            "icon_library": null,
            "display_name": true,
            "name": "Pittwater Beaches",
            "footer": false,
            "trend": false,
            "label": "region_PIWAT",
            "traffic_light_scale": {
                "scale": "std 3 code",
                "codes": [
                    {
                        "colour": "green",
                        "value": "good"
                    },
                    {
                        "colour": "amber",
                        "value": "poor"
                    },
                    {
                        "colour": "red",
                        "value": "bad"
                    }
                ]
            },
            "type": "string"
        },
        {
            "icon_library": null,
            "display_name": true,
            "name": "Central Coast Beaches",
            "footer": false,
            "trend": false,
            "label": "region_CTRCT",
            "traffic_light_scale": {
                "scale": "std 3 code",
                "codes": [
                    {
                        "colour": "green",
                        "value": "good"
                    },
                    {
                        "colour": "amber",
                        "value": "poor"
                    },
                    {
                        "colour": "red",
                        "value": "bad"
                    }
                ]
                },
                "type": "string"
            },
            {
                "icon_library": null,
                "display_name": true,
                "name": "Illawarra Beaches",
                "footer": false,
                "trend": false,
                "label": "region_ILLAW",
                "traffic_light_scale": {
                    "scale": "std 3 code",
                    "codes": [
                        {
                            "colour": "green",
                            "value": "good"
                        },
                        {
                            "colour": "amber",
                            "value": "poor"
                        },
                        {
                            "colour": "red",
                            "value": "bad"
                        }
                    ]
                },
                "type": "string"
            },
            {
                "icon_library": null,
                "display_name": true,
                "name": "Hunter Beaches",
                "footer": false,
                "trend": false,
                "label": "region_HUNTR",
                "traffic_light_scale": {
                    "scale": "std 3 code",
                    "codes": [
                        {
                            "colour": "green",
                            "value": "good"
                        },
                        {
                            "colour": "amber",
                            "value": "poor"
                        },
                        {
                            "colour": "red",
                            "value": "bad"
                        }
                    ]
                },
                "type": "string"
            }
        ],
        "list_label_width": 60,
        "expansion": true,
        "aspect": 1,
        "type": "priority_list",
        "columns": 1
    }
],
"deexpansion_hint": "Show less",
                "expansion_hint": "Show more"
            }
        },
        {
            "subtitle": "Agency tweets outbound",
            "subcategory": "Twitter",
            "source_url": "http://twitter.com/",
            "actual_frequency": "Real time",
            "category": "Social Media",
            "about": "<p>Live Tweets from selected NSW agency Twitter
streams:</p>\r\n<ul>\r\n<li>Business NSW</li>\r\n<li>NSW
Planning</li>\r\n<li>NSW Board of Studies</li>\r\n<li>Service
NSW</li>\r\n<li>Urban Growth NSW</li>\r\n<li>NSW Police</li>\r\n<li>Data
NSW</li>\r\n<li>Teach NSW</li>\r\n<li> Visit NSW</li>\r\n</ul>",
            "category_aspect": 1,
            "name": "Social Media",
            "refresh_rate": 15,
            "url": "tweets",
            "source_url_text": "Twitter",
            "display": {
                "tiles": [
                    {
                        "statistics": [
                            {
                                "icon_library": null,
                                "display_name": true,
                                "name": "",
                                "numbered_list": false,
                                "hyperlinkable": false,
                                "trend": false,
                                "footer": false,
                                "label": "tweets",
                                "type": "string_kv_list"
                            }
                        ],
                        "type": "newsfeed",
                        "aspect": 1,
                        "expansion": false
                    },
                    {
                        "type": "list_overflow",
                        "aspect": 1,
                        "expansion": true
                    }
                ],
                "deexpansion_hint": "Show less",
                "expansion_hint": "Show more"
            }
        }
    ],
    "type": "Migrated View",
    "children": []
}
```

COMMENTS: Widget list will vary by view. Even where a particular widget is always shown, it may be displayed differently depending on the view.

POSSIBLE FUTURE DIRECTIONS: The API could benefit from a clearer separation between metadata describing the meaning and format of the widget data of interest to all consumers of the API; and presentation directives that may be more finely targetted to a particular front-end implementation and design.

5 Get Map Layers

URL: *api-root*/map_layers

AUTHENTICATION: May be required, depending on system configuration and choice of theme.

GET args:

Argument	Description
view	The label of the current (map) view.
hierarchical	(Optional, defaults to empty string) If zero or empty string, return layer menu as a flat list of Map Layer Definition objects. Otherwise return a hierarchical list of menus of Map Layer Definition objects, as described below.

REQUEST example: *apt-root*/map_layers?view=environment&hierarchical=Y

RESPONSE description: A Map Definition object

5.1 Map Definitions

A Map Definition object contains the following keys:

Key	Type	Description
menu	See below.	A menu of layers displayable on the map.
window	Geo-Window Definition Object	The bounding rectangle of the map. Initial display window. Features may also be truncated so that only features inside or intersecting the window are returned.

If a hierarchical response is requested, the menu key may be either a list of Map Layer Menu Objects, or a list of Map Layer Definition Objects. Otherwise, the menu key is always a list of Map Layer Definition Objects.

For a hierarchical response, the menus may currently be at most 2 levels deep (i.e. menus, submenus, layer items).

5.2 Geo Window Definitions

A Geo-Window Definition object contains the following keys:

Key	Type	Description
north	float	The northern limit of the window (GDA-94 degrees latitude).
south	float	The southern limit of the window (GDA-94 degrees latitude).
east	float	The eastern limit of the window (GDA-94 degrees longitude).
west	float	The western limit of the window (GDA-94 degrees longitude).

5.3 Map Layer Menus

A Map Layer Menu Object contains the following keys:

Key	Type	Description
menu_label	string	The label for the menu item
contents	list	May be either another list of Map Layer Menu Objects (i.e. sub-menus) or of Map Layer Definition Objects.

5.4 Map Layer Definitions

A Map Layer Definition Object contains the following keys:

Key	Type	Description
category	string	The category of the widget.
subcategory	string	The subcategory of the widget.
label	string	A short “slug” label for the layer, used to identify the layer in API calls.
name	string	A name for the layer, suitable for display to users.
geom_type	string	The geometry type of the features in the layer. May be: “point”, “line”, “polygon”, “multi-point”, “multi-line”, “multi-polygon”, “predefined” or “external”. “Predefined” means that the first feature property will represent a lookup to a predefined geometry (i.e. as defined by csv-geo-au for Terria compatibility)
properties	List of Feature Property Definition objects	(For internal geom_types only) Definitions of the data properties that may be available for features in this layer.
external_url	string	(For “external” geom_type only) The external URL of the dataset
external_type	string	The type of the external dataset (e.g. “geojson”, “csv”, “czml”, etc)

5.4.1 *Feature Property Definitions*

A Feature Property Definition Object contains the following keys:

Key	Type	Description
label	string	A short “slug” label for the property, used to identify the property in API requests and responses.
name	string	A name for the property, suitable for display to users.
type	string	One of: “string”, “numeric”, “date”, “time”, “datetime”.
class	string	One of: “predefined”: The pre-defined geometry field “data”: The data property field “other”: All other properties
precision	integer	(For properties of “numeric” type only) The number of decimal places.

6 Get Widget Data

URL: *api-root*/widgets/<widget_label>

AUTHENTICATION: May be required, depending on system configuration and choice of theme.

GET args:

Argument	Description
view	The name of the current view. Widgets with the same name/label may differ between views.

REQUEST example: *apt-root*/widgets/air_pollution?view=environment

RESPONSE description: A Widget Data object, which contains the following keys:

Key	Type	Description
widget_last_updated	datetime string (ISO 8601)	When the data was last updated on the backend. e.g. “2015-03-30T22:55:36+0000”
actual_frequency	string	The “actual frequency” to be displayed in the bottom right hand corner of the widget – overrides the value supplied in the Widget Definition.

The Widget Data object also contains an additional key for each statistic defined in the display definition of the widget. The key is the label of the statistic, and the value is either a Statistic Data Object or a list of Statistic Data Objects (if the statistic is a display list statistic or “rotates” is true in the Statistic Definition).

The fields of a Statistic Data Object are:

Key	When	Type	Description
value	Always	string *or* integer *or* float depending on the statistic type.	The data value of the statistic or list item.
traffic_light	If the statistic has a traffic light scale.	string	The traffic light code value for the statistic or list item
trend	If the statistic has “trend”=true	integer	The trend direction for the statistic or list item. Allowed values are 0 (steady), 1 (trending upwards) and -1 (trending downwards).
name	For all “kv_list” type statistics, and statistics where “name_as_label” is false.	string	A user-readable name for the statistic or list item.
icon	If the statistic has an icon library	string	The icon value for the statistic or list item.
url	If the statistic has “hyperlinkable” = true	string	An external URL associated with the list item.
date	For event_list statistics only	date string (ISO 8601)	The date for the list item. e.g. (“2015-03-30”)
datetime_level	For hierarchical_event_list statistics only.	string	The level (granularity) of the associated datetime. One of: “second”, “minute”, “hour”, “day”, “month”, “quarter”, “year”
datetime	For hierarchical_event_list statistics only.	string	The format of datetime varies according to the level. For “hour”, “minute” and “second” datetimes, the format is an ISO 8601 datetime string: “yyyy-mm-ddThh:mm:ss” e.g. “2015-03-30T15:52:55” The minute and second will be truncated to zero as required. For “day” and “month” datetimes, the format is as for the date field described above. e.g. “2015-03-30”. For “month” datetimes the day will be truncated to “01”. For “quarter” datetimes, the format is “yyyyQq”. e.g. “2015Q2”. For “year” datetimes, the format is “yyyy”. e.g. “2015”.

RESPONSE example:

```
{
    "widget_last_updated": "2015-05-10T23:21:14+0000",
    "actual_frequency": "Real time",
    "statistics": {
        "am_pm": {
            "value": "am"
        },
        "average_speed": {
            "trend": 1,
            "value": 94
        },
        "M4": {
            "value": 81
        },
        "M7": {
            "value": 97
        },
        "M1": {
            "value": 100
        },
        "M2": {
            "value": 83
        }
    }
}
```

7 Get Widget Graph Data

URL: *api-root*/widgets/<widget_label>/graph

AUTHENTICATION: May be required, depending on system configuration and choice of theme.

GET args:

Argument	Description
view	The name of the current view. Widgets with the same name/label may differ between views.

REQUEST example: *apt-root*/widgets/air_pollution/graph?view=environment

RESPONSE description: An object which contains one key per graph defined for the widget. The key is the label for the graph, the value is a Graph Data Description Object containing the data for that graph.

A Graph Data Description Object contains the following fields (which may vary depending on the graph type):

Key	When	Type	Description
vertical_axis_scale	Line graphs.	Axis Range Object	The range of values for the vertical axis.
vertical_axis_2_scale	Line graphs with a secondary vertical axis definedAxis Range Object	Axis Range Object	The range of values for the secondary vertical axis.
numeric_axis_scale	Histograms and Bar charts	Axis Range Object	The range of values for the numeric axis.
numeric_axis_2_scale	Histograms and Bar charts with a secondary numeric axis defined	Axis Range Object	The range of values for the secondary numeric axis.
horizontal_axis_scale	Line graphs	Axis Range Object	The range of values for the horizontal axis.
Data	Always	Graph Data Object	The current data to be displayed on the graph.

An Axis Range Object contains the following fields (N.B. The type for the key will reflect the type defined for the relevant axis):

Key	Description
min	The minimum value for this axis in the current data. Note that the relevant "always_show_zero" flag for this axis may also constrain the axis scale.
max	The maximum value for this axis in the current data.

The contents of a Graph Data Object depends on the graph type:

Graph Type	Contents of Graph Data Object
line	One key per line defined for the graph. The key is the url of the line, the value is a list of datapoint pairs. The first element in each datapoint pair is the horizontal axis value, the second the vertical axis value.
histogram	One key per cluster defined for the graph. The key is the url of the cluster, the value is an object containing a key for each bar defined for the graph, with the key being the bar url and the value the numeric value for that bar and cluster.
bar	
pie	One key per pie defined for the graph. The key is the url of the pie, the value is an object containing a key for each sector defined for the graph, with the key being the sector url and the value the numeric value for that pie and sector. NB the numeric values for the sectors of a given pie are not guaranteed to be normalised (i.e. the sum of the sector values for a given pie are not guaranteed to sum to any given value. You must calculate the total yourself.)

The values for the horizontal axis in a line graph are returned as follows:

Horizontal Axis Type	Value type/format
numeric	A JSON number
date	A string in ISO format: yyyy-mm-dd (e.g. “2015-04-25”)
time	A string in 24 hour ISO format: hh:mm:ss (e.g. “13:46:59”)
datetime	A string in ISO date-time format: yyyy-mm-ddThh:mm:ss (e.g. “2015-04-25T13:46:59”)

RESPONSE example (line graph with TIME type horizontal axis):

```
{
      “service_times”: {
            “vertical_axis_scale”: {
                  “min”: 0,
                  “max”: 9.1,
            },
            “vertical_axis_2_scale”: {
                  “min”: 0,
                  “max”: 6.2,
            },
            “horizontal_axis_scale”: {
                  “min”: “07:00”,
                  “max”: “12:30”
            },
            “data”: {
                  “wait”: [
                        [“07:00”, 0],
                        [“07:15”, 3.1],
                        [“07:30”, 7.2],
                        …,
                        [“12:30”, 8.8]
                  ],
                  “service”: [
                        [“07:00”, 0],
                        [“07:15”, 3.5],
                        [“07:30”, 5.2],
                        …,
                        [“12:30”, 5.9]
                  ]
            }
      }
}
```

8 Get Widget Raw Data

URL: api-root/widgets/<widget_label>/rawdata/<raw_data_set_label>

AUTHENTICATION: May be required, depending on system configuration and choice of theme.

GET args:

Argument	Description
view	The name of the current view. Widgets with the same name/label may differ between views.

RESPONSE: A CSV file, in the format specified by the corresponding raw data set definition.

REQUEST example: *apt-root*/widgets/population/rawdata/raw_population?view=demographics

9 Get Widget Map Data

URL: *api-root*/widgets/<widget_label>/map/<map_label>/<map_layer_label>/

AUTHENTICATION: May be required, depending on system configuration and choice of theme.

GET args:

Argument	Description
view	The name of the current view. Widgets with the same name/label may differ between views.
format	Default: json json: Geo-json csv: csv-geo-au compliant CSV (for point type datasets only) html: Geo-json, pretty printed for browser display.
headings	Default is “label” if format=json|html and “name” if format=csv. label: Use property label to label properties. name: Use property name to label properties.

REQUEST example:

> *apt-root*/widgets/air_pollution/map/air_quality/?view=environment&format=json

RESPONSE example:

json format:

> Geo-json feature list. Feature properties as defined for the dataset.

Csv format:

> For “point” data, the first two columns will be headed “Lat” and “Lon” and will contain the latitude and longitude of the point feature as decimal degrees. For “predefined” data, the first column will be a csv-geo-au compatible “canned” region. Subsequent columns will be the feature properties defined for the dataset, in the order they appeared in the definition.

10 Get Map Data

URL: *api-root*/map/<map_layer_label>

AUTHENTICATION: May be required, depending on system configuration and choice of theme.

GET args:

Argument	Description
view	The name of the current view. Maps with the same name/label may differ between views.
format	Default: json json: Geo-json csv: csv-geo-au compliant CSV (for point type datasets only) html: Geo-json, pretty printed for browser display.
headings	Default is “label” if format=json|html and “name” if format=csv. label: Use property label to label properties. name: Use property name to label properties.

REQUEST example: *apt-root*/map/my_map_layer/?view=environment&format=csv

RESPONSE example:

json format:

> Geo-json feature list. Feature properties as defined for the dataset.

Csv format:

> First two columns will be headed “Lat” and “Lon” and will contain the latitude and longitude of the point feature as decimal degrees. For “predefined” data, the first column will be a csv-geo-au compatible “canned” region. Subsequent columns will be the feature properties defined for the dataset, in the order they appeared in the definition.

11 Terria/National Map initialisation json

URL: *api-root*/terria_init/<view_label>/<shown_urls>/init.json

api-root/terria_init/<view_label>/init.json

<shown_urls> is a slash-separated list of Map Layer labels that should be shown initially. Any label provided that does not correspond to the label of a map layer that would be included in the selected view is ignored.

AUTHENTICATION: May be required, depending on system configuration and choice of theme.

REQUEST examples: *apt-root*/terria_init/environment/init.json

apt-root/terria_init/environment/air_quality/rainfall/init.json

RESPONSE: A TerriaJS JSON initialisation file for configuring a TerriaJS/NationalMap instance with the relevant map layers.