



**Weave**

Business Integration Framework

# **Weave System Administrator Guide Excerpt Feb 2022**

1. Frequently Asked Questions	3
1.1 General FAQ	3
1.1.1 Accessibility	3
1.1.2 Benefits of Weave	3
1.1.3 Does Weave compete with systems like ArcGIS Server?	4
1.1.4 How do I access the Admin UI	4
1.1.5 Level of Training (System, and application administrators)	5
1.1.6 Long Term Support	5
1.1.7 Out Of Memory	5
1.1.8 Release Notes	6
1.1.8.1 Release Notes 2.5.21	6
1.1.8.2 Release Notes 2.5.22	11
1.1.8.3 Release Notes 2.5.23	11
1.1.8.4 Release Notes 2.5.24	14
1.1.8.5 Release Notes 2.5.25	18
1.1.8.6 Release Notes 2.5.26	19
1.1.8.7 Release Notes 2.5.28	22
1.1.8.8 Release Notes 2.5.29	26
1.1.8.9 Release Notes 2.5.30 (and 2.6.0)	27
1.1.8.10 Release Notes 2.6.4	28
1.1.8.11 Release Notes 2.6.5	30
1.1.8.12 Release Notes 2.6.6	33
1.1.8.13 Release Notes 2.6.7	35
1.2 BIRT FAQ	37
1.2.1 Can not connect to database via JDBC	37
1.2.2 Can not load list values	38
1.3 DataSource FAQ	39
1.3.1 Cannot load JDBC Driver class	39
1.3.2 Oracle Errors	40
1.3.3 Unable to obtain jdbc connection	40
1.4 SpatialEngine FAQ	41
1.4.1 ArcSDE Errors	41
1.4.2 PropertyDescriptor is null	41
1.5 MapEngine FAQ	41
1.5.1 ArcIMS	41
1.5.1.1 Requested Service is not available	41
1.5.2 Securing Web Mapping Services (WMS)	42
1.6 Support FAQ	42
1.6.1 Support Entitlement	42
1.6.2 Software Updates	42
1.6.2.1 Upgrading Weave	42
1.6.3 Level of support	46
1.6.4 Support Process	46
1.7 Weave 2.5 FAQ	46

# Frequently Asked Questions

## General FAQ

- [Accessibility](#)
- [Benefits of Weave](#)
- [Does Weave compete with systems like ArcGIS Server?](#)
- [How do I access the Admin UI](#)
- [Level of Training \(System, and application administrators\)](#)
- [Long Term Support](#)
- [Out Of Memory](#)
- [Release Notes](#)

### Accessibility

Currently Weave is not able to comply to a WCAG level of accessibility as it is a *mapping application* and some level of vision is required to view the site. If Weave is configured to provide a text only interface (Grid Based) then compliance can be achieved.

Weave also provides comprehensive tooltips on many parts of the interface allowing users to get or find functions easily. The tooltips are configurable and allow a site to change the content of the tooltips to suit the needs of that site. Weave can be configured for keyboard use for the map, which adds the ability to navigate around the map using the arrow keys and the plus and minus keys to zoom in and out.

### Benefits of Weave

#### Increased return on investment in data

Through its short implementation cycle, Weave provides online access to spatial and other database content to all authorised users. As a result, the information can be quickly made available to a wide range of staff without significantly increasing the administrative overhead, and with an improvement of the return obtained from the data.

The powerful database integration capabilities provide the foundation for the creation of value added information products, enhancing the use of the data, and the value of that data to the organisation.

#### Increased revenue generation

Rapid visualisation and the cross-linking of data means that existing corporate data stores can be tested for accuracy and compatibility. Each individual database can be validated against others. More reliable data leads to improved quality of service and better client satisfaction, which can result in higher revenue generation.

Spatial data can be made accessible to other stakeholders such as customers, business partners and suppliers, featuring in visually compelling e-commerce applications.

#### Capital cost savings

The utilisation of a development environment well accepted in the IT world (OSGi, J2EE, Ajax, JSON, SQL, XML) allows the IT group to administer the system without the need for specialised knowledge, leading to overall capital cost savings.

The use of free Web browsers as the interface to the integrated data allows organisations to deliver the information to a wider range of staff without additional costs. The cost per seat is low and the system is extensible.

The Weave server supports internal users as well as users or customers on the Internet. Another licence to support such separated activity does not need to be purchased. In addition the Weave server is not licensed per user, and can be used to the maximum activity that the infrastructure can support.

Further savings can be achieved by minimising the number of applications required throughout the organisation. This is typical of places using desktop applications for inquiry and reporting, which Weave replaces with a web browser, saving maintenance dollars.

#### Increased productivity

The easily configurable SQL/XML interface facilitates the rapid fulfillment of "change requests", e.g. change a search; integrate more databases, create new reports, etc. This reduces IT staff workload and enables improved productivity in the broader workforce.

The facility provided to all users for the generation of reports relieves specialist staff of the associated workload. This allows the specialist staff to focus their activities on tasks more aligned with their business objectives such as data maintenance and improvement that are likely to benefit a wider audience.

Users only require a web browser for accessing Weave, while providing significant introductory and analytical GIS capabilities (e.g. buffer, geographic feature selection, reporting), which can significantly increase staff productivity.

Output from Weave can be easily incorporated into company documents. It takes little time to implement a user-specified search, report, or map, thereby significantly improving staff productivity and the effectiveness of presentation material.

Managers are provided with a much improved view of the organisation's corporate data, as Weave makes the key elements of the business visible through a more integrated view of the business information. This results in faster and better informed decision making.

#### Operational cost savings

New users are productive using the default Weave client interface within minutes, as it is highly intuitive. This represents a major cost saving compared with the training required to use traditional desktop applications provided by the decision support systems currently in use.

Staff qualified through a three-day training course can administer Weave. This enables organisations to avoid the outsourcing of this activity if desired. This can lead to operational cost savings through reduced system administration costs.

By encapsulating the linkages between various corporate datasets in a simple, open SQL/XML system, business rules are captured and preserved, holding business knowledge openly within the application itself, not fractured by groups or personalities, and helping to minimise retraining costs in the case of staff loss or transfer.

Does Weave compete with systems like ArcGIS Server?

If you want to develop a pure GIS Viewing or Editing web based application that only talks to your Feature Classes in ArcSDE then Weave does not compete in this space. Weave is not about being another WebADF, Flex and JavaScript API. ESRI for example is about GIS, Weave is about Business Systems Integration where GIS is one of the Integration technologies that we work with. Such systems may be Relational Databases, Property Management Systems, Document Management Systems, Asset Management Systems, and others. GIS provides the spatially enabling path for these systems and most of the time it is the main integration the Weave user will work with.

 This same argument can also be applied to other mapping products, e.g. Erdas Appollo, ArcIMS, MapExtream, etc

The following bullet points summarise the answer to a typical question such as "For an ESRI developer, what does Weave provide on top of the ESRI technology which ESRI is not providing already?".

- Insulate the developer from the lower level api's of the particular system.
- Provide easy access to spatial features in a variety of Spatial Engines.
- Has advanced Security at all levels of the Weave system including restricting access to entities, data, clients, etc
- Connection Pooling across all connections (spatial and database)
- Ability to write plugins that contribute code to the Weave subsystem.
- Work seamlessly with Spatial and Non-Spatial Data.
- Integrate with multiple Map Engines regardless of the underlying Spatial Vendor.
- Create multiple client front ends without having to write any code.
- Write custom code to enhance a client(s) Views and Actions.
- Natural Language Geocoding Engine.
- Built in reporting engine Business Intelligence Reporting Tools (BIRT) that enables the creation of sophisticated multi page Reports that can exported in a variety of formats (pdf, doc, xls, etc)
- Openness - Use of open, non-proprietary APIs and SDKs and support for multiple database and spatial engines
- Integration of Spatial Data - Powerful data integration from multiple spatial data sources such as Oracle Spatial, ArcSDE, Shapefiles, PostGIS, WFS, GML and others
- Integration of Non-Spatial Data - Powerful data integration from multiple JDBC and ODBC non-spatial data sources
- Integration of Map Services - Simultaneously work with many map services such as WMS, ArcIMS, ArcGIS Server, MapXtreme, Image Web Server, MapServer and others
- Insulation - The data complexity is hidden from the end users
- Simple configuration - Minimise/consolidate configuration information
- Centralised - The configuration is controlled by the site administrator from a single location
- Web/Browser-based - Weave has browser-based clients using JavaScript and HTML. No browser plugins are required
- Customisable/Expandable - Weave can be customised and expanded by third parties. Weave has been built with its own open software development kit which is supplied with the product
- Minimal footprint - Minimal requirements to integrate data sources (i.e. no selection tables, no forced schemas)
- Scalable - A multi-tier architecture is supported
- Security - Role based access control to interfaces, data, and reports

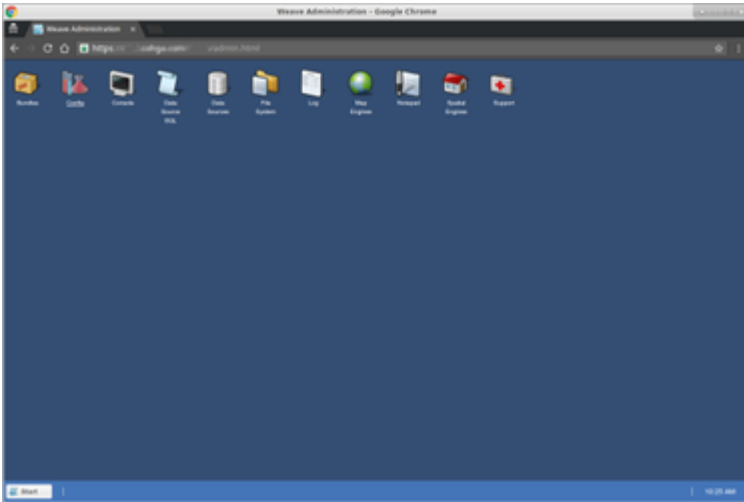
A key Weave strength is the ability to get a site up and running with the integration of multiple Spatial, Map, and Data Engines quickly where as doing that with ESRI products you will be starting from a very basic level. If the site needs to integrate with other systems that are outside the ArcSDE environment regardless if they are spatial or non-spatial then Weave at a high level is a good fit.

How do I access the Admin UI

The administration user interface is available via the URL `/weave/admin.html` on the weave server, so it would be something like:

```
http://weaveserver:8080/weave/admin.html
```

You should secure access to this page following the instructions at [Securing the Admin UI](#)



### Level of Training (System, and application administrators)

Once the installation of Weave and the basic configuration of the system is complete, then Cohga recommends that at least 3 days of administrator training be undertaken by 2 to 4 people to best understand, manage, and Configure the Weave system.

The purpose of the instruction is to impart basic key knowledge of the Weave architecture. In addition, the instruction will enable the Weave administrator(s) to perform basic administration of the system, configure Weave, and monitor and interpret system logs.

The training makes use of the customer's data and systems to allow the administrator(s) to grasp concepts using entities and layers from the perspective of the customer.

The BIRT reporting system that is bundled with Weave is included in the training. This system is substantial in its functionality and within the time provided for Administrator training, only a limited introduction to BIRT is provided in the initial training. The purpose of the instruction in BIRT is to enable the customer to work on reports for their site in the BIRT format and to then be able to extend their current reporting capability.

Cohga also instructs the customer in the use of the online Cohga support site.

**!** More advanced training can be provided to help those who wish to perform Customisation (programming, development) of the system.

### Long Term Support

#### Weave 2.4.10 Gets Long Term Support

In an effort to help sites who have exhaustive processes to upgrade software we're providing version 2.4.10 of Weave as a long term support version.

This means that version 2.4.10 will continue to receive bug fixes even after 2.4.11 is released, thereby allowing for a site to maintain a stable environment whilst still receiving bug fixes from Cohga.

Previously bug fixes and enhancements were combined into the latest release and to receive bug fixes you also needed to receive the latest enhancements, which could result in new bugs being introduced.

Additionally it provides a stable base for custom development, where a site can develop, or have a third party develop for them, custom bundles using 2.4.10 as the base platform and know that the custom bundles will continue to work whilst still receiving bug fixes from Cohga.

#### Out Of Memory

##### **OutOfMemoryError: GC overhead limit exceeded**

You may see the following error in the logs if no more memory is available within the Java Virtual Machine (JVM).

"GC overhead limit exceeded"

This error message can occur in multiple locations throughout the code if the memory limit has been exceeded. If parts of the application require more memory then this message will typically be seen in different contexts which may seem strange. It is recommended that Weave is shutdown and that the memory limit is increased to allow for the extra load on the server.

##### ***java.lang.OutOfMemoryError: requested 32756 bytes for ChunkPool::allocate. Out of swap space?***

This message is essentially saying that the native objects do not have enough memory to use. This is usually because you have allocated too much memory to your heap thereby reducing the amount that is available for native objects. See [this article](#).

The solution is to reduce the amount of heap memory that you have allocated. For example if you have set `-Xmx4096`, then you should consider reducing this to `-Xmx2048m`.

**!** Remember that if you are using a 32bit JVM you cannot allocate more than `-Xmx2048m` for linux (and less than that for windows). Using a 64 bit JVM can resolve this problem.

## Release Notes

- [Release Notes 2.6.7](#)
- [Release Notes 2.6.6](#)
- [Release Notes 2.6.5](#)
- [Release Notes 2.6.4](#)
- [Release Notes 2.5.30 \(and 2.6.0\)](#)
- [Release Notes 2.5.29](#)
- [Release Notes 2.5.28](#)
- [Release Notes 2.5.26](#)
- [Release Notes 2.5.25](#)
- [Release Notes 2.5.24](#)
- [Release Notes 2.5.23](#)
- [Release Notes 2.5.22](#)
- [Release Notes 2.5.21](#)

### Release Notes 2.5.21

**Weave 2.5.21** includes a number of bug fixes and enhancements to existing tools. A full list of changes can be found at the Weave [What's New](#) page.

To help you decide if you should upgrade, the top 5 enhancements are listed here. If these functions are useful to your Weave users, it's time to upgrade!

### *Spatial Mappings*



#### **Spatial Mapper Update**

There has been a breaking change made to the spatial mapper configuration if you specify the spatial engine at the mapping level and not at the top level of the spatial mapper.

Previously a single spatial mapper would allow you to add mappings that pointed to different spatial engines, this is no longer supported.

Additionally, if you specify the spatial engine setting for each mapping, and do not set it once at the top level in the spatial mapper, you will have to change this so that there is a single spatial engine set at the top level of the spatial mapper rather than having the spatial engine set for each mapping. Both formats were previously supported, but now only setting the spatial engine at the top level is supported.

If you have multiple spatial engines you will need to create separate spatial mappers for each spatial engine.

For example the following spatial mapper

#### **Invalid format spatial mapper**

```
<mapper:mapper id="mapper.main">
  <mapping>
    <entity>suburbs</entity>
    <spatialEngine>spatialengine.main</spatialEngine>
    <table>SUBURBS</table>
    <key>SUB_NAME</key>
  </mapping>
  <mapping>
    <entity>wards</entity>
    <spatialEngine>spatialengine.main</spatialEngine>
    <table>WARDS</table>
    <key>WARD_NAME</key>
  </mapping>
</mapper:mapper>
```

would need to change to

**Valid format spatial mapper**

```

<mapper:mapper id="mapper.main">
  <spatialEngine>spatialengine.main</spatialEngine>
  <mapping>
    <entity>suburbs</entity>
    <table>SUBURBS</table>
    <key>SUB_NAME</key>
  </mapping>
  <mapping>
    <entity>wards</entity>
    <table>WARDS</table>
    <key>WARD_NAME</key>
  </mapping>
</mapper:mapper>

```

Also, if you have more than one spatial engine in a spatial mapper

**Invalid format spatial mapper**

```

<mapper:mapper id="mapper.main">
  <mapping>
    <entity>suburbs</entity>
    <spatialEngine>spatialengine.arcsde</spatialEngine>
    <table>SUBURBS</table>
    <key>SUB_NAME</key>
  </mapping>
  <mapping>
    <entity>wards</entity>
    <spatialEngine>spatialengine.sqlserver</spatialEngine>
    <table>WARDS</table>
    <key>WARD_NAME</key>
  </mapping>
</mapper:mapper>

```

it will need to be split into two spatial mappers

### Valid format spatial mapper

```
<mapper:mapper id="mapper.arcsde">
  <spatialEngine>spatialengine.arcsde</spatialEngine>
  <mapping>
    <entity>suburbs</entity>
    <table>SUBURBS</table>
    <key>SUB_NAME</key>
  </mapping>
</mapper:mapper>

<mapper:mapper id="mapper.sqlserver">
  <spatialEngine>spatialengine.sqlserver</spatialEngine>
  <mapping>
    <entity>wards</entity>
    <table>WARDS</table>
    <key>WARD_NAME</key>
  </mapping>
</mapper:mapper>
```

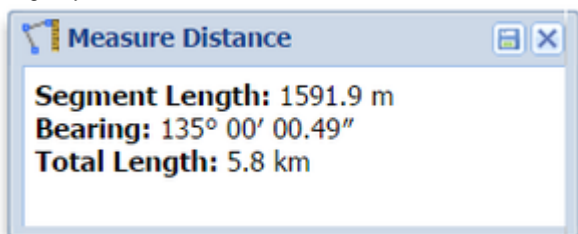
### 👍 Measure Tools

The *Measure Polyline* and *Measure Polygon* tools have improved in both form and function:

- A new button has been added to the Measure windows allowing the measured item to be added to the Redline layer. Once saved, the measured item becomes a standard Redline object.
- Number of digits reported in the Measure windows can be changed by setting `digits` value, default is 3. For example, using this:

```
<item action="weave.measurePolyline" windowTitle="Measure
Distance" digits="1" />
```

will give you this:



- The initial Measure window placement has been changed so it's not in the middle of the screen.
- You can decide which of the three measurement outputs (segment, bearing and total) are reported in the window by hiding the outputs that you don't want. For example, using this:



```
<item action="weave.measurePolygon" windowTitle="Measure Area"
hideBearing="true" windowHeight="75"/>
```

will give you this:



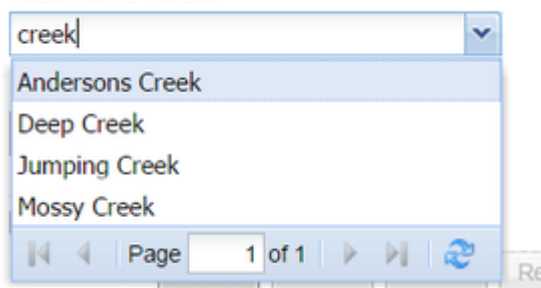
### 👍 List Box Filtering

It's now possible to use more sophisticated filtering in your list boxes in a Search. Previously a list box would match on the start of the text entered by the Weave user. But now you can set the word match to be at the start, the middle or the end of the text. By setting `matchType` to `middle` or `end` you can filter whether the text typed by the user is in the middle or at the end of the value, the default is `start`. For example, using this:

```
<search:parameter id="StreetNameMid" promptText="Road Name Middle"
controlType="listbox" dataSet="streetnames" column="cid.prop.
road_name" valuecolumn="value" labelcolumn="label" matchType="middle"
/>
```

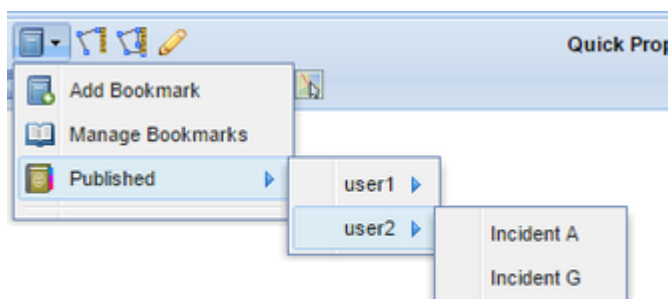
will give you this:

Road Name Middle:



### 👍 Bookmark Menu

The Published Bookmarks have been better organised as they are now sorted firstly by the user name and secondly by the Bookmark name. This will improve Bookmark management at sites that have many published Bookmarks.



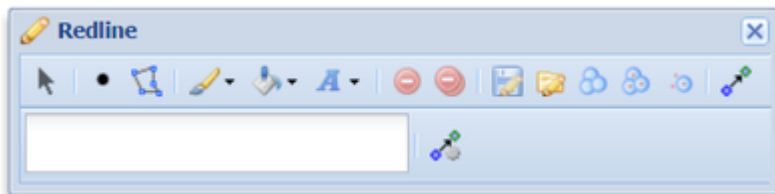
## 👍 Redline Toolbar

It's now possible to configure which tools and settings menus appear in the Redline toolbar by adding `geometryTool` and/or `settingTool` values to the `redline` action.

For example using this:

```
<item action="weave.redline" multilineLabel="true">
  <geometryTool>point</geometryTool>
  <geometryTool>polygon</geometryTool>
  <settingTool>stroke</settingTool>
  <settingTool>fill</settingTool>
  <settingTool>font</settingTool>
  <tool>save</tool>
  <tool>load</tool>
  <tool>buffer</tool>
  <tool>bufferAll</tool>
  <tool>bufferSelected</tool>
</item>
```

will give you this:



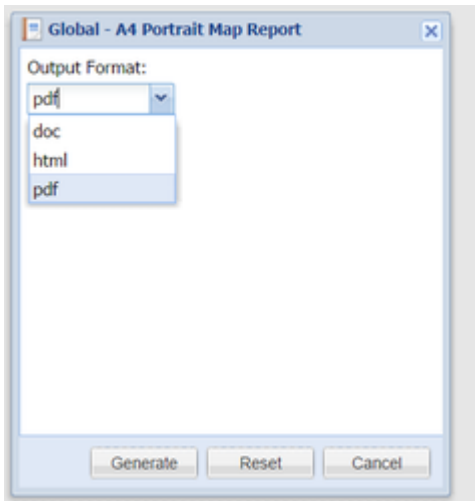
## 👍 BIRT Reports

It's now much easier to specify which report formats are available for each report. By setting a `format` value in a BIRT report you can restrict the format types.

For example using this:

```
<birt:report id="a4_port_mapreport" report="reports/map_report_port.
rptdesign" label="A4 Portrait Map Report" format="pdf,doc,html" />
```

will give you this:



### Release Notes 2.5.22

**Weave 2.5.22** includes a number of bug fixes and enhancements. This release is predominately focused on enhancements to existing tools rather than the addition of new tools and functions. A full list of changes can be found at the Weave [What's New](#) page.

👉 Main enhancements are:

- **BIRT Reports**
  - The option of setting a coordinate system for a BIRT report has been added. This allows your Weave client to be using one coordinate system while a BIRT report creates a map in another. This is useful if you are using a coordinate system in your Weave client that is not suitable for accurate distance measurements but want to include a scale and scale bar on your BIRT report map.
  - And correspondingly, Tiled Map Engines that are using OpenStreetMap, Google, GeoWebCache can be re-projected on the server for display in a BIRT report map. This is useful when generating a map in a BIRT report in a different projection than that being used in the Weave client.
- **Measure Tools**
  - The Weave 2.5.21 release included a new button on the Measure windows that allows the measured item to be added as a Redline object.
  - A number of fixes have been applied to improve the functioning of this new capability.
- **WMS Map Engine**
  - Scale dependencies are now being honoured when accessing ArcGIS Server map services via WMS.

### 🟢 Upgrading

If you need assistance with running a Weave upgrade, refer to this **How-to** page: [How to Run a Weave Software Upgrade](#).

### Release Notes 2.5.23

**Weave 2.5.23** includes a number of bug fixes and enhancements. This release is predominately focused on enhancements to existing tools rather than the addition of new tools and functions. These enhancements have increased the flexibility and usability of some of Weave's standard tools. A full list of changes can be found at the Weave [What's New](#) page.

Main enhancements in **Weave 2.5.23** are:

### 👉 Spatial Identify

The Spatial Identify tool can be set to return result for all entities or only the entities associated with visible layers using the `useVisibleEntities` tag.

For example, using this in your client XML file:

```
<item action="com.cohga.html.client.map.spatialIdentify">
  <useVisibleEntities>true</useVisibleEntities>
</item>
```

means that the ToC controls the layers that will be interrogated when using the Spatial Identify tool (i.e. the entities associated with visible layers will be used).

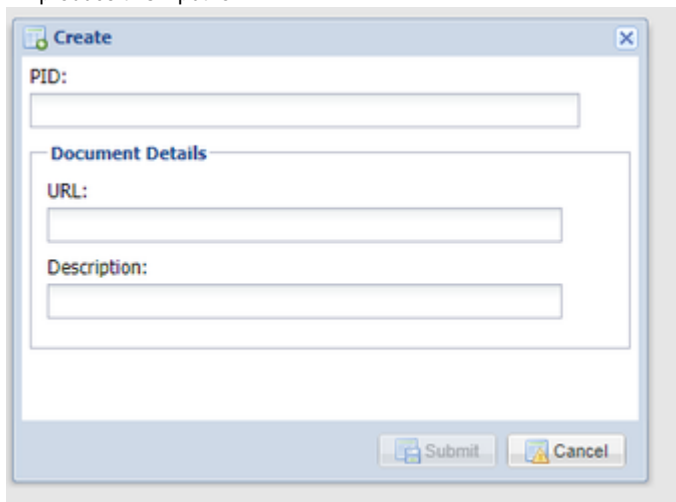
Conversely, setting the `useVisibleEntities` tag to `false` means that the Spatial Identify tool will return results for any entity (independent of whether the layer is visible or not).

## Editing

- Spatial Editing
  - The `createNew` action will turn on the layer associated with the entity (if one is set) when the button is pressed.
- Non-spatial Editing
  - Support has been added for field groups so client input forms, built via `<parameter>` config items, can now group fields. This improves presentation of the input form. For example the following editor configuration:

```
<editor:table id="nse.property_document">
  <label>Property_document</label>
  <datasource>gis</datasource>
  <table>docs.property_document</table>
  <parameter id="pid" label="PID" column="pid"
dataType="integer" />
  <parameter id="id" label="Id" column="id" dataType="
integer" autogenerated="true" hidden="true"/>
  <parameter id="details" label="Document Details">
    <parameter id="url" label="URL" column="url"
/>
    <parameter id="description" label="
Description" column="description" allowBlank="false"/>
  </parameter>
</editor:table>
```

will produce this input form:



The screenshot shows a 'Create' dialog box with a light blue header and a close button (X) in the top right corner. The form contains the following elements:

- A label 'PID:' followed by a single-line text input field.
- A section header 'Document Details' in blue text.
- Under 'Document Details', a label 'URL:' followed by a single-line text input field.
- Below the URL field, a label 'Description:' followed by a single-line text input field.
- At the bottom of the dialog, there are two buttons: 'Submit' (with a blue icon) and 'Cancel' (with a yellow icon).

## Buffer Tool

- Buffer style enhancement
  - The buffer style can be set in the map view by added a <buffer> section in the <highlight> tags.
- Map highlight enhancement
  - It's now possible to specify that highlights and buffers are automatically removed from the map after a fixed delay, rather than waiting until the new map redraw using the <autoClear> and <autoClearDelay> tags.

Both of these enhancement are shown in the extract of the client XML file shown below:

```
<view id="com.cohga.html.client.map.mapView">
    ....
    <highlight>
        <marker>
            ....
        </marker>
        <vector>
            ....
        </vector>
        <buffer>
            <strokeWidth>3</strokeWidth>
            <strokeColor>red</strokeColor>
            <strokeOpacity>0.75</strokeOpacity>
            <fillColor>yellow</fillColor>
            <fillOpacity>0.125</fillOpacity>
        </buffer>
        <autoClear>true</autoClear>
        <autoClearDelay>2000</autoClearDelay>
    </highlight>
    ...
</view>
```

- Generation parameters
  - Various buffer related tools can now be configured to specify buffer generation parameters (include CRS to generate buffer in)

## Reports

- Report panel enhancement
  - Groups will expand and collapse based on the Active Entity to show all the reports available for that entity.
- Improved BIRT legends
  - Legends have been neatened in BIRT reports that are using ArcGIS Server, WMS and Weave Map Engines. The improvements include better line wrapping for long text and the removal of redundant information.

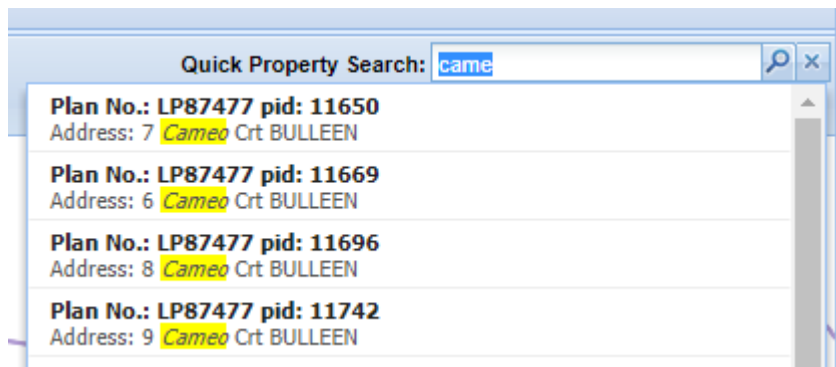
## Custom Stylesheets

- Support for custom stylesheets
  - It is now possible to create a CSS file that allows for simple overriding of CSS styles used in Weave.
  - For example a CSS file can be created to override the highlighting used in a Quick Search. The file needs be called . . . \weave\platform\workspace\styles.css for it to be added to the client at startup. An example of its content is shown below:

**styles.css**

```
.index-item .index-h1 {
  background: #ffff00;
}
```

And it will produce this result in your Weave client:



- Note that for this to work the `highlight` tag needs to be set to "true" for the index combo in the client XML file (`weave.indexcombo`).

### ✓ Upgrading

If you need assistance with running a Weave upgrade, refer to this **How-to** page: [How to Run a Weave Software Upgrade](#).

## Release Notes 2.5.24

**Weave 2.5.24** includes a number of bug fixes and enhancements.

This release is predominately focused on fixing bugs with some of the more recent tools while expanding on the functions provided by these tools. These enhancements have increased the flexibility and usability of some of Weave's standard tools. It also includes a number of security and REST API updates.

A full list of changes can be found at the Weave [What's New](#) page.

*Main enhancements in **Weave 2.5.24** are:*

### 👍 BIRT Reports

#### A. Select Layers for Map in a Report

It is now possible to set up BIRT report that allow users to select layers they want from a list (of predefined layers) to appear on a map in the BIRT report. This means you can add flexibility to your existing reports with a minor change.

To set layers to be drawn in a BIRT report via user parameter, open the *BIRT Report Designer* and:

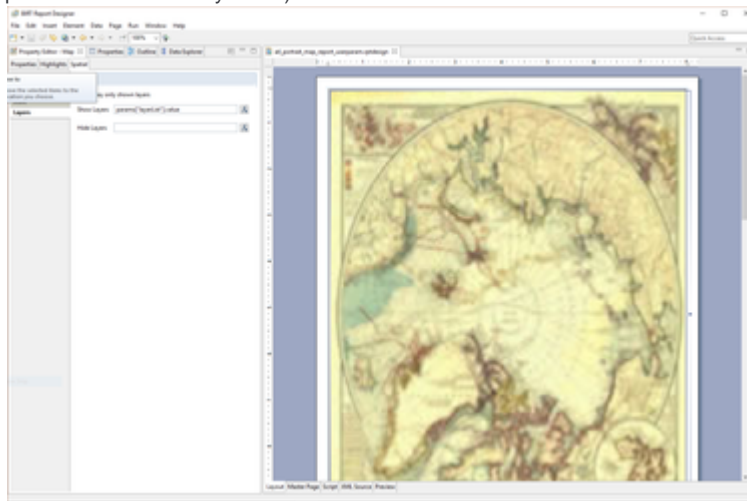
1. Add a new Report Parameter to the map
  - a. Outline > Report Parameters > New Parameter
  - b. Make the new parameter a List Box
  - c. Press the *New* button to create a new list entry.
  - d. Put the layer value in the *Value* column. This is the layer item from your ToC model.  
E.g. If this was in your ToC model:  

```
<entry id='l_mapengine.testing_planning_zones_copy' layer='Planning Zone Labels' label='Planning Zone Codes' checked='true'/>
```

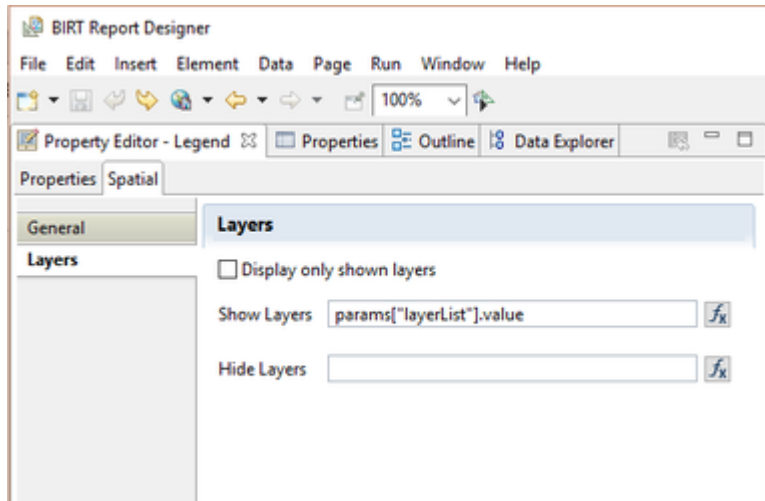
 Then the bold text, **Planning Zone Labels**, is what you'd put in the *Value* column.
  - e. You can list individual layers (so the user only gets to choose one layer) ...
  - f. Or list individual layers, and then check the *Allow Multiple Values* option (so the user can choose more than one layer) ...

- g. Or, use themes/groups (so the user chooses one group name that equates to multiple layers). In this case:
- Set the *Display Text* to be the theme/group name
  - In *Value* column, list the layer names (which is the layer value from the ToC model or the name of the layer coming from the Map Engine). The names should be separated by a comma, no space, and the whole thing must be enclosed in double quotes (see the image below).

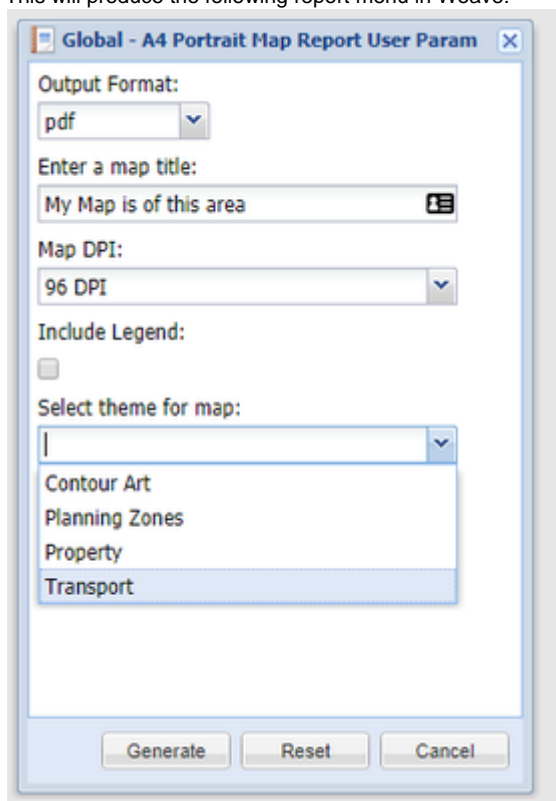
- Add this user parameter to the map element.
  - Click on the map from the report preview
  - Go to Properties Editor - Map > Spatial > Layers
  - For the *Show Layers* item, add the reference to your new parameter, e.g. `params["layerList"].value` (where the parameter is called "layerList").



- If you want these layers to appear in the legend
  - Click on the legend from the report preview
  - Go to Properties Editor – Legend window > Spatial > Layers
  - For the *Show Layers* item, add the reference to your new parameter, e.g. `params["layerList"].value` (where the parameter is called "layerList")



4. This will produce the following report menu in Weave:



## B. Set Coordinate System for Map in Report

Prior to Weave 2.5.24, a CRS could be set for a map by manually adding a "crs" property. This has now been extended so that the CRS can be set by the user via the UI.

To set the CRS for a BIRT report as a user parameter, open the *BIRT Report Designer* and:

1. Add a new Report Parameter to the map, called *proj*:
  - a. Outline > Report Parameters > New Parameter
  - b. Make the new parameter a List Box
  - c. Press the *New* button to create a new list entry.
  - d. Enter the projection information in the *Value* column as "EPSG:xxxx" as shown in image.
  - e. Enter the *Display Text* as what you want to appear to the user on the report window.



f. Add as many CRS entries as you need using steps c to e above.

**Edit Parameter**

Define the properties of the report parameter.

Name:  Display As:

Prompt text:  Help text:

Data type:  Format as:

Display type:  Preview with format:

List Box:  List Limit:

Is Required  Do not echo input  
 Hidden  Allow Duplicate Values

Selection list values

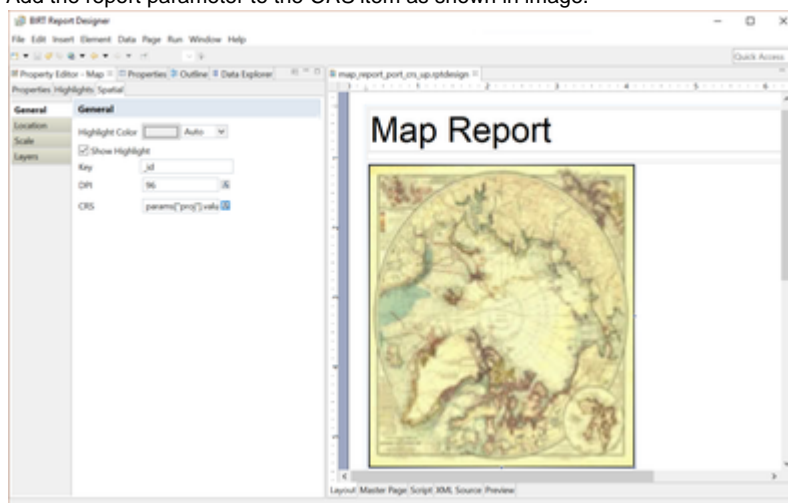
Static  Dynamic  Allow Multiple Values

Default	Value	Display Text	Display Text Key
	EPSG:4283	Geographic GDA94 ...	
	EPSG:28354	MGA Zone 54 (GDA...	
	EPSG:28355	MGA Zone 55 (GDA...	
	EPSG:3111	VicGrid 1994 GDA9...	
	EPSG:32633	UTM Zone 33N - EP...	

Sort

Sort by:  Sort direction:

2. Add this user parameter to the map element:
  - a. Click on the map from the report preview
  - b. Go to Properties Editor – Map window > Spatial > General
  - c. Add the report parameter to the *CRS* item as shown in image.



## Weave Administration

The number of times each map layer is accessed is logged as part of the Weave logging process.

- The number of times a layer is drawn has been added to the *Map Layer Times* panel in the *Server Status* Tool in the Admin Tool. It's under the heading of "Count".

ArcGIS Connections	Map Engine	Layer ?	Avg. Map Time (ms)	Count
Bundle Status	geoserver	mann.buildings_council	1272.8	4
Config Items	geoserver	mann.busroutes	0	0
Equal Distribution	geoserver	mann.busstops	0	0
Hourly Requests	geoserver	mann.contours_01m	0	0
Hourly Response	geoserver	mann.contours_02m	0	0
JDBC Connections	geoserver	mann.contours_05m	0	0
Map Engine Times	geoserver	mann.contours_10m	0	0
Map Layer Times	geoserver	mann.drainage	0	0
Pending Requests	geoserver	mann.easement	0	0
Quantile Distribution	geoserver	mann.easetxt	0	0
Request Details	geoserver	mann.house_numbers	778.47	15
Response Distribution	geoserver	mann.hydro	0	0
Response History	geoserver	mann.lighting_manningham	0	0
User Details	geoserver	mann.main_hyd	0	0
Weekly Requests	geoserver	mann.mainroads	778.47	15
	geoserver	mann.mccbound	778.47	15
	geoserver	mann.mccwards	0	0
	geoserver	mann.property	939.4	5
	geoserver	mann.property_dissolve	939.4	5
	geoserver	mann.reserves	943.63	8
	geoserver	mann.roads	943.63	8

- Or shown on the equivalent of the <http://localhost:8080/weave/server/status?page=map.logging.layer> page.

Map Engine	Layer	Avg. Map Time (ms)	Count
geoserver	mann.buildings_council	1272.79	4
geoserver	mann.busroutes	0	0
geoserver	mann.busstops	0	0
geoserver	mann.contours_01m	0	0
geoserver	mann.contours_02m	0	0
geoserver	mann.contours_05m	0	0
geoserver	mann.contours_10m	0	0
geoserver	mann.drainage	0	0
geoserver	mann.easement	0	0
geoserver	mann.easetxt	0	0
geoserver	mann.house_numbers	778.47	15
geoserver	mann.hydro	0	0
geoserver	mann.lighting_manningham	0	0
geoserver	mann.main_hyd	0	0
geoserver	mann.mainroads	778.47	15
geoserver	mann.mccbound	778.47	15
geoserver	mann.mccwards	0	0
geoserver	mann.property	939.39	5
geoserver	mann.property_dissolve	939.39	5
geoserver	mann.reserves	943.62	8
geoserver	mann.roads	943.62	8
geoserver	mann.schoolsprimary	0	0
geoserver	mann.schoolssecondary	0	0
geoserver	mann.suburb	698	10
geoserver	mann.yvw_sewer_pipe	0	0
geoserver	mann.yvw_water_pipe	0	0


- Turning the layer off/on, without any other change to the map, won't increment the count as this just uses the browser's cached version of the image and does not go to the server to request the map again. If any navigation tools are used to change the map extent and layers are redrawn, then the count will be incremented by one for each change.
- This additional piece of logging data will help Weave Administrators identify layers that are seldom or never used.

**Release Notes 2.5.25**

Weave 2.5.25 includes a number of bug fixes and enhancements.

This release is mostly focused on enhancements and corrections to some of Weave's most recent tools. The enhancements expand on the functions provided by these tools and also increase the flexibility and usability of some of Weave's standard tools.

A full list of changes can be found at the Weave [What's New](#) page.

 Main enhancements are:

- **Anonymous user local storage**

In response to a number of user requests an important change has been made in the way stored items such as Redlines and Bookmarks are persisted for an anonymous user. These items are now stored in the browser for anonymous users, allowing them to save those items across a browser restart (in supported browsers, i.e. those that support JavaScripts `localStorage` API).

- **Editing Extension**

A number of improvements have also been made to the Weave Editing extension. This enables the Weave client to be used for editing operations that would normally have been undertaken in other systems. This is beneficial as editors are also generally Weave users, and they can now edit in an environment in which they are already familiar and comfortable. Allowing users to edit in the Weave client simplifies editing operations for users who could be daunted when faced with the complexity of a Desktop GIS software for what can be a simple editing task (e.g. move a point, update an attribute, etc.).


- **Spatial upload enhancement**

When a layer is added to the ToC via the [Upload Data](#) tool, you can now zoom to this layer using the Context Menu "Zoom to layer" item.

 **Upgrading**

If you need assistance with running a Weave upgrade, refer to this **How-to** page: [How to Run a Weave Software Upgrade](#).

## Release Notes 2.5.26

 2.5.26 had an issue with installation and was immediately replaced by 2.5.27

**Weave 2.5.26** includes a number of bug fixes and enhancements.

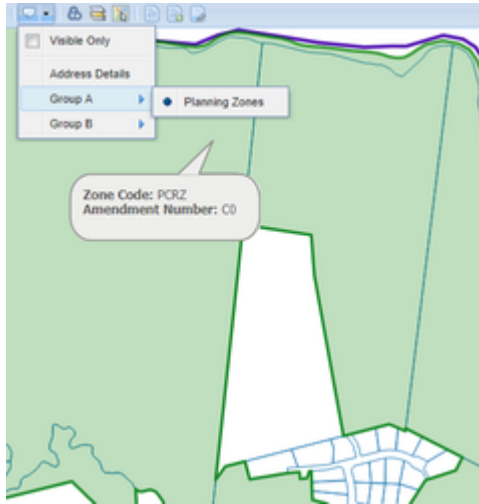
This release is mostly focused on enhancements and corrections to some of Weave's most recent tools. The enhancements expand on the functions provided by these tools and also increase the flexibility and usability of some of Weave's standard tools.

A full list of changes can be found at the Weave [What's New](#) page.

**The main enhancements in 2.5.26 are:**

 Map Tips

- **Showing Geometry**
  - Map Tips will now show the geometry of the related feature in the default highlight style.
  - This is turned on by default and can be turned off by setting `showGeometry` to `false` in the Map Tip tool configuration.
  - If you want to set a different highlight style, use the `highlight` tags as per other parts in the client configuration.
- **Visibility Flags**
  - A new `visibleOnly` flag has been added to the tool and individual map tips, to specify that the tips should only be displayed if the layer is turned on in the ToC. The default value is `false`.
  - There is also `visibleOnlyMenu`, which if set to `true`, adds a new menu item to the Map Tips menu to allow the user to turn the visible only setting on or off. The default is `false`.
- **Groupings**
  - Map Tips can be grouped in the Map Tips menu by setting a `group` attribute for the tip. All tips with the same value for the `group` attribute will be included in the same sub-menu, and tips without a `group` attribute will appear in the main menu.



The XML required to implement the four changes shown above (geometry display, visibility flag, visible-only menu and groupings), are shown in the client XML extract below. Also included is an example of setting a different highlight style for Map Tips.

### Client XML file extract

#### 👍 Table of Contents (ToC)

- Context Menu enhancement for `checkAll` and `uncheckAll`
  - These two items now allow you to set `includeParent` to `true` or `false` to change the behaviour of the option with respect to also turning on or off any parent ToC entries. The default setting for `checkAll` is `true` and for `uncheckAll` it's `false` also as these would be the most common required behaviours.
- Automatically turn on a layer when an entity is selected
  - By setting `turnOnActive` to `true` in the ToC panel Weave will turn on the layer related to the active entity if it's not already turned on.
- Index enhancement
  - An index search can now be set to turn on the layer with the entity (via the ToC model) by setting `doTurnOnLayer` to `true` for the index combo in your client configuration.

#### 👍 Selection Identify

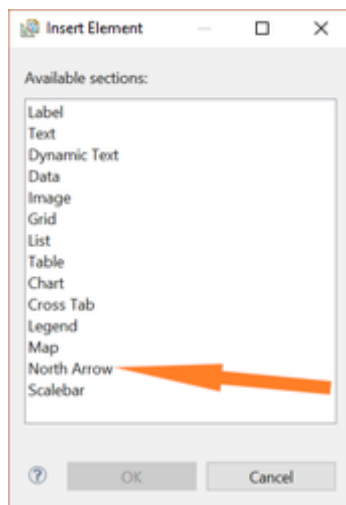
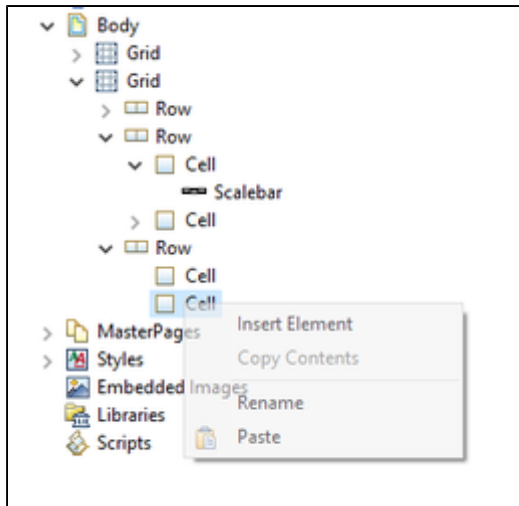
- There is much more flexibility with this tool now as you can:
  - Set it to just search visible layers, by setting `useVisibleEntities` to `true` in the tool configuration.
  - Specify a buffer to use when selecting entities that intersect the selection, by setting `bufferDistance`, `bufferUnits` and `bufferCrs` in the tool configuration.
    - Setting a small negative buffer distance can help report on the correct records when there is a slight spatial overlap between the selected feature and its neighbours as it will stop (or at least reduce) the features being found as overlapping the selected feature.
  - Specify a spatial operation to use when selecting entities that "intersect" the selection, by setting `spatialOperation` to one of `intersect`, `contains`, or `overlaps`.
  - Specify if the active entity should be included in the search, by setting `includeActive` to `true` in the tools configuration.

#### 👍 Spatial Editing

- You can now specify a table directly in the edit configuration that should be edited, rather than relying on what's in the spatial mapping for the entity. This should help with editing entities that are spatially backed by a database view.

#### 👍 BIRT

- BIRT includes a new North Arrow component that can be added to a report. Add it as you would any other component in a BIRT report.



- This is a standard north arrow but means you don't have to load in your own image if you want an north arrow in a report.



#### 👍 Upload File

- KML Files
  - The KML parser is now more forgiving of invalid KML files that are uploaded so you should be able to load up a broader range of KML files.
  - Files that contain label points are supported and will display labels on the Weave Map.

#### 👍 NearMap and Street View Integration

- Reduction of resources used
  - The NearMap and Street View panels will not try and retrieve map tiles when the panel is not visible.
  - This should reduce your NearMap and Street View data usage significantly.

#### 👍 Database Timeouts

- Timeouts added to data definitions
  - Database related data definitions now have a server-side timeout, with a default of 120 seconds, to ensure that long-running queries do not keep running.
  - You can set the timeout, in milliseconds, in the options for the data definition, or globally as a default, or turn it off completely, using the `timeout` tag.
  - Further details and an example can be found on this [wiki](#) page.

## Administration

- Map Engine Refresh
  - It is now possible to refresh a Map Engine using `memd reset` at the OSGi console or via the Map Engine tool of the Admin Tool. This tells Weave to clear any cached information it may have on a back-end map service and is useful when you change the service and need Weave to take note of the update.
- Map Engine Health Checks
  - Health checks have been added for more Map Engine types.
  - These can be viewed in the Server Health tool of the Admin Tool.
- Spatial Metadata
  - The `spmd` command (at the OSGi console or via the Console Tool of the Admin Tool) had been updated so that it outputs the count of records when describing a table.

## Upgrading

If you need assistance with running a Weave upgrade, refer to this **How-to** page: [How to Run a Weave Software Upgrade](#).

## Release Notes 2.5.28

**Weave 2.5.28** includes a number of bug fixes and enhancements.

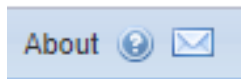
The enhancements increase the flexibility and usability of some of Weave's standard tools. In particular there are a few editing related enhancements which increase the capability of Weave's Editing functions making it a viable alternative for a range of editing scenarios.

A full list of changes can be found at the Weave [What's New](#) page.

The main enhancements in 2.5.28 are:

## Feedback function

- Weave has had a tool that allows feedback to be sent to the site administrator. This has now been made part of core Weave and when added to a client file will display an envelope icon as shown below.

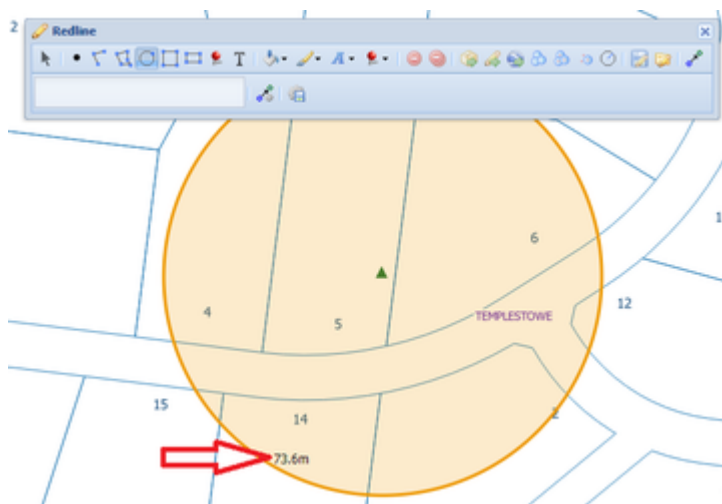


- The tool is documented in the Weave Configuration Reference: <https://cohgawiki.atlassian.net/wiki/x/CYABL>
- For those already using this tool, a checkbox has been added to the feedback form giving the user an opportunity to include the current map in the email.

## Redlining

- Radius Display

When using the Circle Redlining tool to draw a circle, the radius will be dynamically displayed when you drag out a circle (in the same as the way the Circle Select tool).



- Drawing a Circle by Radius

It is possible to create a circular buffer of a specified radius using the `circleByRadius` tool.

For example, the Redline toolbar in your client XML file could look like this:

```
<item action="weave.redline" multilineLabel="true">
  <tool>save</tool>
  <tool>load</tool>
  <tool>entity</tool>
  <tool action="weave.redline.exportMarkers">
    <defaultProj>EPSG:28355</defaultProj>
    <projection id="EPSG:4283" label="Geographic GDA 1994"
  />
    <projection id="EPSG:28355" label="GDA 1994 MGA Zone
55" />
  </tool>
  <tool>exportKml</tool>
  <tool>buffer</tool>
  <tool>bufferAll</tool>
  <tool>bufferSelected</tool>
  <tool>circleByRadius</tool>
</item>
```

## Searching and Editing

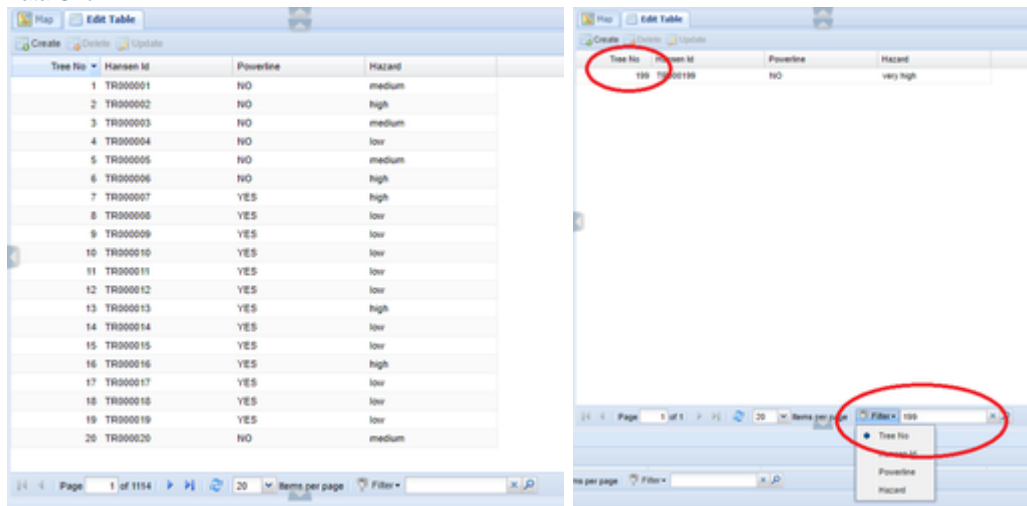
- Time fields now support setting the `minvalue`, `maxvalue` and `increment` tags. This can be used in the Search and Editing functions.
- This has been extended to other field types, where minimum and maximum values and lengths can be set with the following tags: `minvalue`, `maxvalue`, `minlength`, `maxlength`, `increment`. Use the combination of tags that meet your requirements.

### Search example

```
<search:attribute id="sample_point.attr" entity="
sample_point" displayname="Sample Point Search" datasource="gis"
table="mods.sample_pt" key="id">
  <search:parameter id='type' column='type' promptText="Type"
controlType="listbox" dataset="list.poly_type"/>
  <search:parameter id='cost' column='cost' promptText="
Cost" controlType="textbox" datatype='integer' maxlength='5'
minlength='2' minvalue='20' maxvalue='1000' />
  <search:parameter id='checkdate' column='checkdate'
promptText="Check Date" controlType='textbox' datatype='date' />
  <search:parameter id='checktime' column='checktime'
promptText="Check Time" controlType='textbox' datatype='time'
increment='10' />
</search:attribute>
```

## Non-spatial Editing

- Setting a filter on the grid used for non-spatial editing has now been enabled. This works like the *Filter* option that is available on the Data Grid.



### 👍 StreetView

- Map View location
  - The Weave Map View updates to show StreetView location. So when moving locations in the StreetView tab, the Weave Map View will be updated to reflect this new location.
- StreetView Compass
  - There were reported cases when the compass on the StreetView tab was not appearing. To overcome this, the compass is now forced to display so should always appear in the StreetView tab.
- StreetView Controls
  - More controls have been added for the StreetView tab. The Google Street View API allows more control over what's displayed on the view and where it's displayed and these have now all been exposed in the configuration for the panel. These can be set as shown at the bottom of this Weave wiki page <https://cohgawiki.atlassian.net/wiki/x/h4Mx>. The result of these settings is shown in the image below, highlighted by red ovals.



### 👍 Zip 'n' Ship

- You can now specify a list of entities that can be exported. This can be used to restrict the exporting of layers that might be large, or that you have no authority to be supplying.



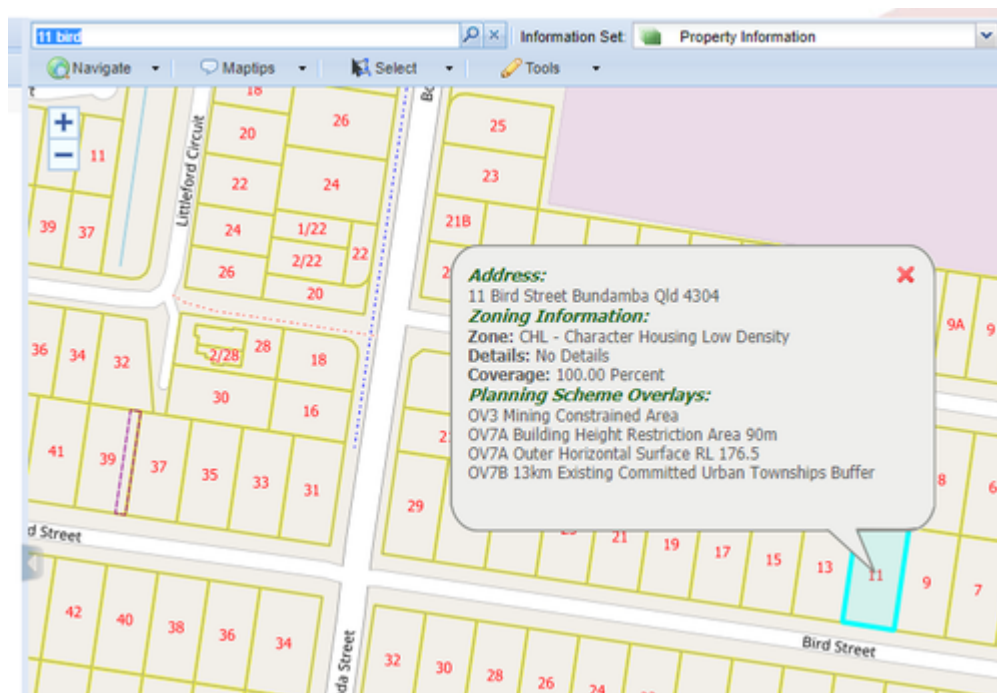
```
<item action="weave.zipnship">
  <entities>
    <entity>property</entity>
    <entity>roads</entity>
  </entities>
</item>
```

- You can also use this to export a group of entities or a theme as shown in the example below.

```
<item action="weave.zipnship">
  <tooltip>
    <title>Zip'n'Ship Greenery</title>
    <text>Export selected Tree or Reserve data</text>
  </tooltip>
  <entities>
    <entity>councilreserve</entity>
    <entity>counciltrees</entity>
  </entities>
</item>
```

## 👍 Map Tips

- Behaviour
  - The behaviour of the Map Tip display has been changed slightly. The tip will now stay on the map if the mouse is in the popup area. This gives a better experience for Map Tip users.
- Map Popup
  - There is a new tool that is a variation on the standard Map Tip tool. The Map Popup tool provides the ability to display a map popup when there is exactly one feature selected on the map. The feature could be selected by any supported selection method (i.e. Quick Search, graphical selection, attribute selection, etc.).
  - It uses data definitions to supply the data, meaning that the data available to the user can come from any provider, not just the attributes attached to the underlying spatial feature.
  - The tip can be displayed using HTML formatting to improve the appearance of the data.
  - This new feature is documented in the Weave [Configuration Reference](#).



## BIRT

- Displaying entries for the current selection can be confusing for some users so there is now an easy way to disable them. At the moment the `remove` tag can be used but this can now be simplified with the `selectionMode` tag. It is similar to `showMode`, will be added with "on" and "off" being the initial options, the default will be "on".

### Client legend config

```
<view id="com.cohga.client.panel.legend" iconCls="">
  <label>Legend</label>
  <location>west</location>
  <remove>planning zones labels, house_numbers</remove>
  <showMode>visible</showMode>
  <extentOnly>true</extentOnly>
  <selectionMode>off</selectionMode>
</view>
```

## Release Notes 2.5.29

**Weave 2.5.29** includes a number of bug fixes and enhancements.

The enhancements increase the flexibility and usability of some of Weave's standard tools. The main enhancements are related to Bookmarks, how they are managed and used. These enhancements increase the flexibility and use of Bookmarks through the Weave client.

A full list of changes can be found at the Weave [What's New](#) page.

The main enhancements in 2.5.29 are:

## Bookmarks

The new functions available for Bookmarks are described below.

- Bookmark Manager (this isn't new but has been expanded)
  - This is a separate Published Bookmark management tool (`weave.bookmark.manage`) which can be provided to specific users (controlled via ACLs) allowing them to open and delete shared bookmarks and open, edit and delete published bookmarks.
  - This tool also provides information about when a bookmark was created, when it was last executed and how many times it's been executed.
  - This tool provides access to *all* published and shared bookmarks, not just those associated with the current user (as the other Bookmark tools do).
- The Bookmark tool, `weave.bookmark`, and the Published Bookmark tool, `weave.publishedbookmark`, have been updated
  - To allow for administrator defined groups to be used to create sub-menus rather than just using the user id of the person that created the bookmark.
- The Bookmark tool, `weave.bookmark`, has been updated to provide:
  - A direct Publish and Share option
    - Rather than having to create a private bookmark then publish or share that a published or shared bookmark can be created directly.
  - Additional management options
    - The management window can also manage shared and published bookmarks and show the information about when a bookmark was created, when it was last executed last executed and how many times it's been executed.
- OSGi bookmark commands
  - New commands are available at the OSGi console, `bm`, to perform administrative actions on bookmarks, including:
    - listing bookmarks that reference a particular ToC entry (based on the ToC model entry id)
    - list bookmarks
    - dump bookmarks to provide text output on the entire contents of the bookmark
    - list users
    - remove users bookmarks

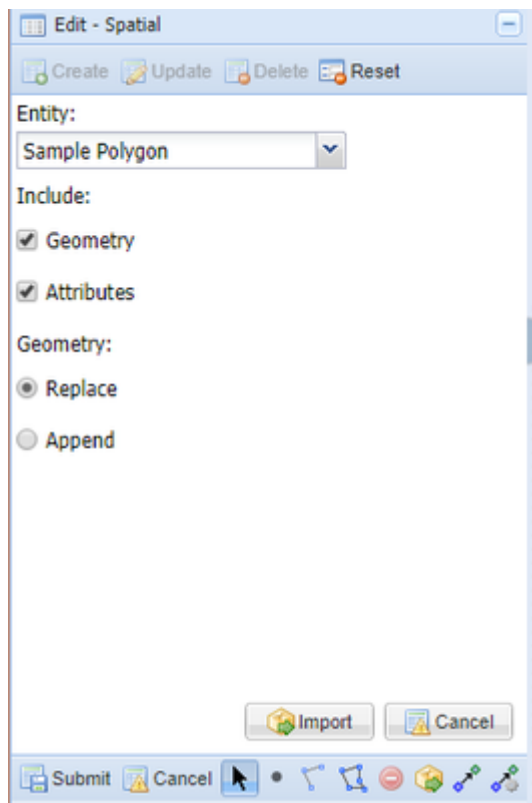
- Shared Bookmarks
  - This is a new concept for the Weave client. A user can "share" a Bookmark using the *Share Bookmark* tool.
  - This creates a URL which can be copied or sent as an email.
  - When used, this URL will open Weave, and zoom to, display layer, display Redlines, set the Active Layer and/or include a selection as specified when the Bookmark was created. It will apply ACLs if they have been set up.
- Active Layers
  - It is now possible to include selection in a Bookmark.
  - The ids of the features selected will be included so if the data changes after a Bookmark has been created, the Bookmark may refer to different features if the ids (as set by the <key> tag for the entity) change.

### **i** More information

Read more about setting up Bookmarks in the Weave [Configuration Reference](#).

### **👍** Spatial Editing

- Previously the Import tool in the Spatial Editing panel only imported geometry, it has now been extended to offer the option of importing the attributes.
- The ability copy geometry from anything remains as is.
- The additional functionality will allow geometry and attributes to be copied from the currently selected entity if and only if it is the same type as the entity being edited and there is only one entity selected.



### **👍** Report Designer

- Previously the BIRT report designer was included as part of the main Weave server installer, and an optional extra, it has now been moved to into its own separate installer which can be downloaded from the same location as the other components.

### **✓** Upgrading

If you need assistance with running a Weave upgrade, refer to this **How-to** page: [How to Run a Weave Software Upgrade](#).

**Weave 2.5.30** (and **2.6.0**) includes a number of bug fixes and enhancements. The enhancements increase the flexibility and usability of some of Weave's standard tools. The main enhancements are listed below.

A full list of changes can be found at the Weave [What's New](#) page.

The main enhancements in 2.5.30/2.6.0 are:

### Data Grid

#### Filtering Data

There have been significant improvements on how the `Filter` in a Data Grid works. The current operation and examples of how the tool works are provided in the [Weave User Guide](#).

#### Export

When using the `Export` button the current filter is honoured so only the filtered results will be exported.

#### Copy Cell Value

It's now possible to *copy* the value of a cell in the Data Grid to *paste* and use outside of Weave. Double click on the cell (for a single value) or click and drag and then right mouse click and select "Copy". The value has now been written to the clipboard.

To enable this function set the `textSelect` to `true` in your Grid. See [this page](#) for more information.

### Spatial Data Definition

Spatial Data Definitions now support filters so you can `<filter>` tag in a `spatialdataconnection` or `spatialintersectiondataconnection`.

```
<data:datadefinition id="busstops_spatial">
  <spatialdataconnection entity="busstops">
    <filter><![CDATA[ACTIVE = 'true']]></filter>
    <parameter column="STOPID"/>
    <parameter column="HOUSE" dataset="vl_list_test" valuecolumn="
value" labelcolumn="label"/>
    <parameter column="FID"/>
    <parameter column="ADDRESS"/>
  </spatialdataconnection>
</data:datadefinition>
```


### Upload Tool

When uploading a MapInfo Interchange Format (MIF) file, the projection is read from the file so does not need to be set by the user through the Upload UI.

### Bookmark Manager Tool

The *Bookmark Manager Tool* in the *Weave Admin Tool* now:

- Includes a `User` button so you can filter on User ID quickly find a user without having to sort and scroll through the list.
- Includes a `Refresh` button.
- Reports dates in an Australian format (yyyy-MM-dd).

 If you are looking to upgrade from 2.5.x to the new 2.6.x more details are in the [Installation Guide](#).

## Release Notes 2.6.4

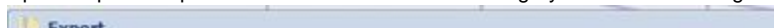
**Weave 2.6.4** includes a number of bug fixes and enhancements. The enhancements increase the flexibility and usability of some of Weave's standard tools. The main enhancements are listed below.

A full list of changes can be found at the Weave [What's New](#) page.

The main enhancements in 2.6.4 are:

### Zip 'n' Ship Envelope

Zip 'n' Ship Envelope now allows the addition of a category. This is useful for grouping layers to be exported.





An example of how to config the `category` tag is found in the [ZipNShip Envelope Configuration Reference](#) page.

### 👍 Selection Identify

You can now exclude entities from the Selection Identify tool. This is done by adding an `exclude` property to the *Selection Identify* tools (you can also have an `include` property).

The `include` value will list entities that are allowed to be enabled for the tool, and the `exclude` list will list entities that should not be allowed to enable the tool.

Only one of `include` or `exclude` should be set depending upon if its easier to exclude entities (you have a handful you don't want it enabled for) or include them (you only have a few that you want it enabled for).

An example of how these properties can be used is given in the Configuration Reference page - [Client Actions Selection Identify](#).

### 👍 Weave Help

The Weave Help bundles are now included as part of a new Weave installation.

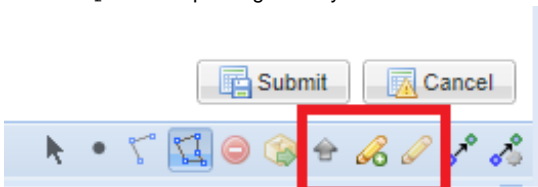
The bundles are not added during an update, only installed as part of a clean installation, and only if the *Online Help* extension is selected under the list of *Extensions* in the installation wizard. However the core help bundle, not the content, would be updated if an older version was installed prior to an update.

The updating of this help content is described here [How-to Update Weave Help Content](#).

### 👍 Spatial Editing

Spatial data can now be loaded from a file or by entering text coordinates. This broadens the scope of the Editing extension immensely. The three new tools (`Upload`, `Create` & `Update`) will appear on the Spatial Editing toolbar by default. The new tools:

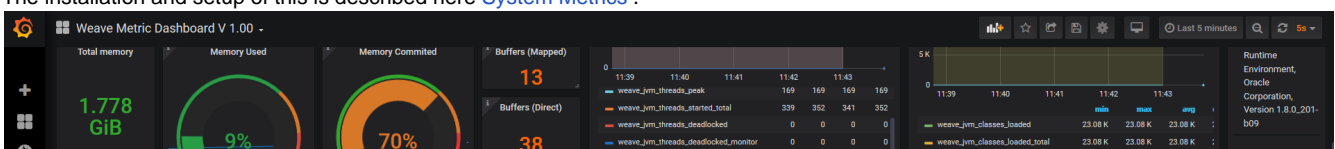
- `Upload` - Upload geometry from a file
- `Create` - Create geometry from text coordinates
- `Update` - Update geometry from text coordinates

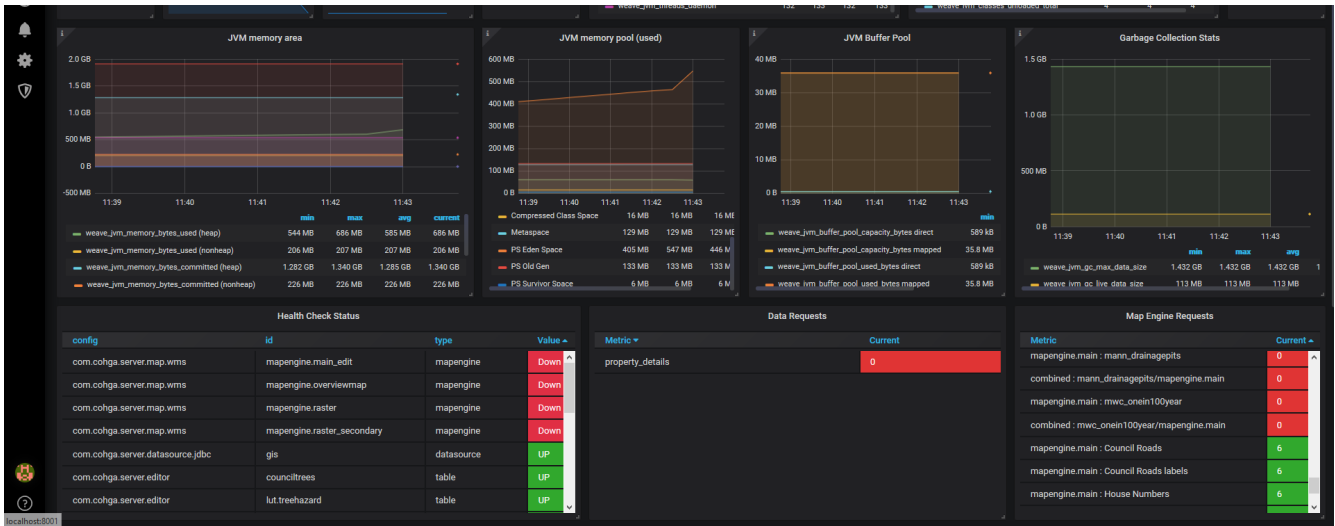


### 👍 System Metrics

Weave now supports System Metrics. These are counters, gauges and histograms that can be used to monitor the performance of the Weave server for your environment.

The installation and setup of this is described here [System Metrics](#).





## Release Notes 2.6.5

Weave 2.6.5 includes a number of bug fixes and enhancements.

The enhancements increase the flexibility and usability of some of Weave's standard tools. In particular there are a few editing related enhancements that increase the capability of Weave's Editing functions making it a viable alternative for a range of editing scenarios.

This release also completes the work required to make Weave GDA2020 compliant.

A full list of changes can be found on the Weave [What's New](#) page.

**⚠️ You will not be able to use the runtime installed with Weave to start the updater because this release contains an update to that underlying Java Runtime.**

If you need to update Weave on a server that does not have another Java Runtime available use the weave-updater ISO file which contains Java Runtime along with the updater.

### The main enhancements in 2.6.5 are:

#### 👍 Non-Spatial Editing - Editing Options

- You can now specify if an edit config is creatable, updatable and/or deletable at the edit level. This allows the editing options to be tailored to your unique editing requirements.
- If you want all your edits to have the same options, set them at the panel level like this:

```
<!-- Define panel that allows updating records only -->
>
    <view id="editor.panel.simplegrid">
      <label>Edit - Non-spatial</label>
      <location>west</location>
      <enableCreate>>false</enableCreate>
      <enableDelete>>false</enableDelete>
    </view>
```

- But if you want your edits to have different editing options then don't set them at the edit level, set them at the edit level like this:

```
<editor:table id="nse.plans_document">
  <label>Plans Document</label>
  <creatable>>true</creatable>
  <deletable>>false</deletable>
  <updatable>>true</updatable>
  <datasource>gis</datasource>
  <table>docs.plans_document</table>
```

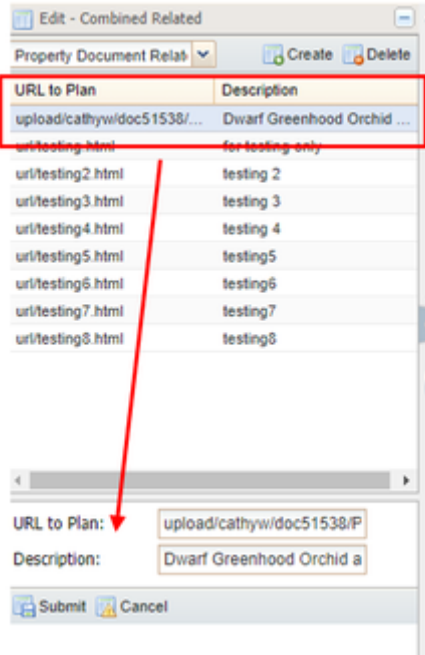
```

        <key>objectid</key>
        <parameter id="url" label="URL" column="url" />
        <parameter id="planno" label="Plan Number"
column="planno" />
    </editor:table>
    
```

- For more details on editing refer to the [Non-spatial Editing](#) page.

**Non-Spatial Editing - Combined Editor View**

- The Combined Editor View builds on the functionality provided in the Simple Edit View. The Combined Editor View displays all related records while allowing one record to be edited.
- For more details on editing refer to the [Non-spatial Editing](#) page.



**Table of Contents - Favourites**

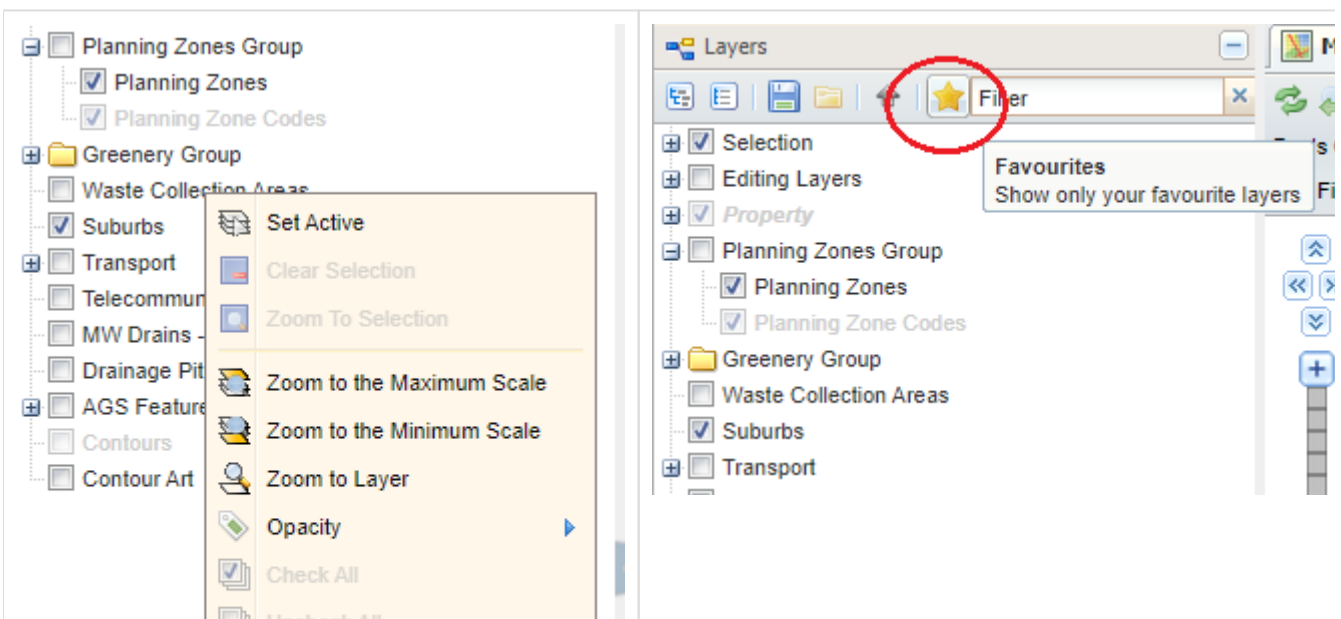
- Users can now set layers as favourites so they can easily find layers they regularly use.
- Layers are marked as favourites in the ToC View Context Menu by adding this to the client config:
 

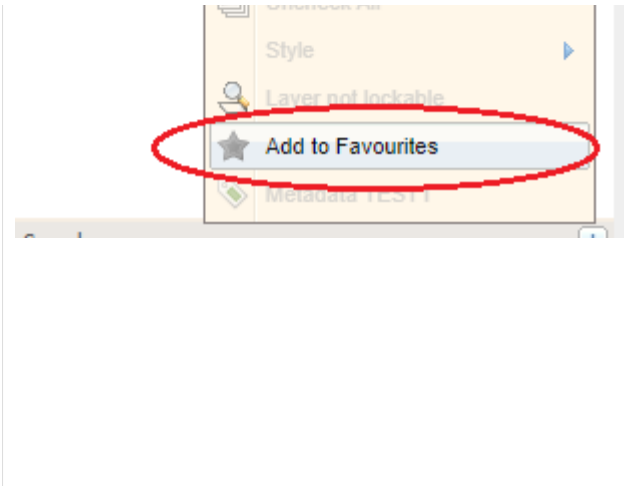
```

            <item action="weave.toc.favourite" />
            
```
- Favourite layers are listed in the *Favourite* toolbar item with this client config:
 

```

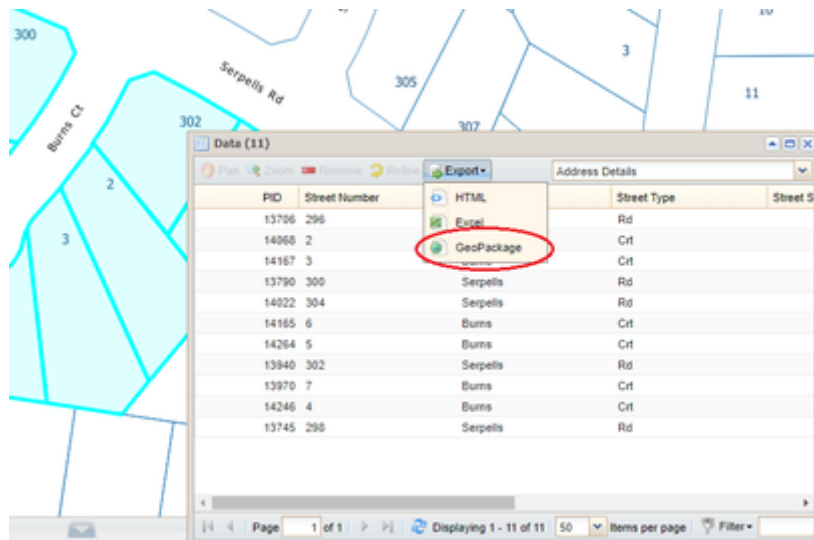
            <item action="weave.toc.favourites" />
            
```
- A full example of this in a client file is on the [Client Views ToC](#) page.





### Data Grid Export

- The export options allowed in a Data Grid now include *GeoPackage*. So your Data Grid can export to CSV, HTML, Excel and GeoPackage.



- A sample of this in a client configuration is given below. For more details on exporting refer to the [Client Actions Grid export](#) page.

```

<item action="com.cohga.html.client.actions.gridWindowAction">
  <textSelect>true</textSelect>
  <pageSize>50</pageSize>
  <toolbar>
    <item action="com.cohga.html.client.grid.
panAction" />
    <item action="com.cohga.html.client.grid.
zoomAction" />
    <item action="weave.grid.remove" />
    <item action="weave.grid.refine" />
    <item action="com.cohga.client.actions.
menuAction">
      <text>Export</text>
      <iconCls>icon-database_go</iconCls>
      <item action="com.cohga.html.client.grid.
exportAction">
        <text>HTML</text>
        <format>html</format>
        <iconCls>icon-page_white_code<

```



```

/iconCls>
                                </item>
                                <item action="com.cohga.html.client.grid.
exportAction">
                                    <text>Excel</text>
                                    <format>excel</format>
                                    <iconCls>icon-page_white_excel<
/iconCls>
                                </item>
                                <item action="com.cohga.html.client.grid.
exportAction">
                                    <text>GeoPackage</text>
                                    <format>gpkg</format>
                                    <iconCls>icon-page_white_world<
/iconCls>
                                </item>
                                </item>
                                </toolbar>
</item>

```

### Setting a Projection for Redlines


- Redlines created before Weave version 2.6.5 (and 2.6.4.8) did not record the projection of the stored coordinates. This would cause a problem if the client map projection was changed or if a bookmark/redline created in a client with one map projection was opened in a client with a different map projection.
- The 2.6.5 Weave release includes two changes. It:
  1. Ensures that the CRS for any new redlines is stored with the redline, and
  2. It provides an OSGi command to add the projection information to existing redlines.
- After you have run this upgrade, we **strongly** recommend you:
  - Set a coordinate system for redlines that are missing a CRS value using:
 

```
redline setcrs <crs>
```

where <crs> is the CRS to set, e.g. EPSG:3857  
 e.g. `redline setcrs EPSG:3857`  
 This command will update the bookmarks and redlines to store what CRS was used when they were created. *This will not alter the stored coordinates.*
  - If you need to alter the coordinates for redlines, use the following command to change the stored CRS:
 

```
redline transform <from> <to>
```

e.g. `redline transform EPSG:28355 EPSG:7855`  
 Only redlines that have a CRS matching the <from> CRS will be converted to the <to> CRS. *This command will alter the stored coordinates.*
- It is recommended that you backup your redlines and bookmarks with the `storage` and `ustorage` commands before using the `transform` or `setcrs` commands.
- Note, it is **not** mandatory to transform your redlines into the projection of the Weave client. If these two projections are different then the coordinates of the redlines will be transformed each time the users load the client.

 The `redline setcrs <crs>` command will set the CRS value for all redlines to the CRS provided in the command, but it won't update any redlines that already have a CRS value set. However, you should note that it assumes that *all* the redlines that it will update were created using the CRS provided in the command.

If you only use a single client, or all of your clients use the same CRS for the map view, then this will not be a problem and the command will just update all the stored redlines to indicate the CRS that the client/clients used.

If, however, you use multiple clients and they have different CRS values for the map view, then it could be that you have bookmarks stored with coordinates that were generated with the different projections (if the bookmarks and/or redline saving is enabled for the client) and you won't know which CRS was used to create which redlines. We do not believe this to be a common occurrence but if you do fall into the category please contact Cohga Support for help.

### Release Notes 2.6.6

**Weave 2.6.6** includes a number of bug fixes and enhancements.

The enhancements increase the flexibility and usability of some of Weave's standard tools. An important enhancement is the *Layer Filtering* which provides an exciting new capability for map display.

This release also fixes a minor bug related to Weave being fully GDA2020 compliant. Check the [What's New](#) page for full details of this fix.



```

<item action="weave.redline" multilineLabel="true">
  <tool>entity</tool>
  <tool>exportKml</tool>
  <tool>buffer</tool>
  <tool>bufferAll</tool>
  <tool>bufferSelected</tool>
  <tool>circleByRadius</tool>
  ...
  <tool>spatialSelect</tool>
</item>

```

## Release Notes 2.6.7

**Weave 2.6.7** includes a number of bug fixes and enhancements. The enhancements increase the flexibility and usability of some of Weave's standard tools.

The most important change in this release is the use of a new logging library ([Logback](#)). This change ensures Weave is using a modern, secure and versatile logging framework. [Logback](#) is a logging framework for Java applications, created as a successor to the [log4j](#) project.

A full list of changes in this release can be found on the Weave [What's New](#) page.

## The main enhancements in 2.6.7 are:

### 👍 Weave Logging

- The logging in Weave has always been done using a third party library. Prior to Weave 2.6.7 [Log4j 1.2](#) was used, and from 2.6.7 Weave is now using [Logback](#).
- In essence, the logging works as it has in previous versions of Weave with the following changes:
  - Logging configuration is through the `logging.xml` rather than `logging.properties` file.
  - The ordering of items in the `logging.xml` file is important so the ordering in the `logging.xml` file provided in the Weave upgrade should not be altered.
  - The `logging.xml` file contains comments in every section to assist the Weave Administrator with changing settings.
  - When Weave has been installed without any modifications to the `logging.xml` file, Weave logs information to the `weave.log` file within the `$WEAVE_HOME/logs/` directory when started as a service/daemon.
  - You can now also use the `dump logs` OSGi command, or the Admin Tool Log Tool, to see additional log output if Weave was not started as a service/daemon.
- For more details on logging refer to the [Administration Guide Logging](#) page.

### 👍 Single Image Slider

- Previously offered as a custom bundle, the Single Image Slider is now part of core Weave.
- This slider offers an alternative to having basemap-type layers in the Table of Contents and allows transitioning visibility between vector and raster layers.



- An example of configuring the Single Image Slider is given on the [Client Components](#) page.

### 👍 Splash Screen

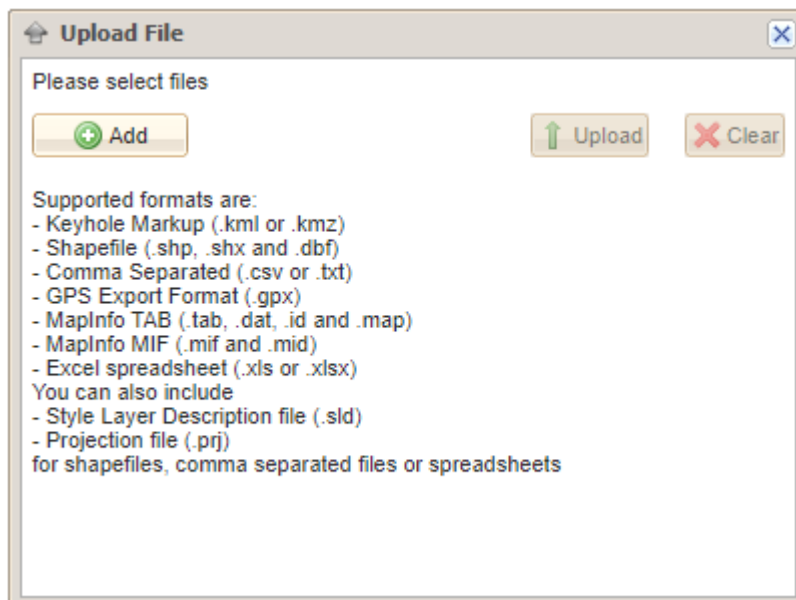
- The ability to display a message to the user as they enter Weave, and have them interact with that message, has been enhanced. This enhancement provides full splash screen functionality.
- The new splash screen has the combined function of what was previously available as the licence and start-up tip. A splash screen provides a way to display information to the user at start-up similar in appearance to the licence agreement but provides additional options to customise the experience for the user.



- A full description of how to configure a splash screen can be found in [Client](#) section in the Weave Configuration Reference.

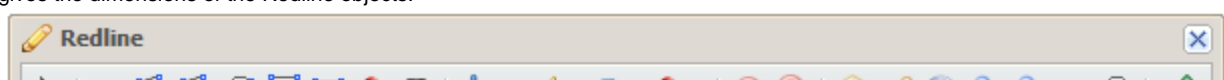
#### Data Upload - Spreadsheet

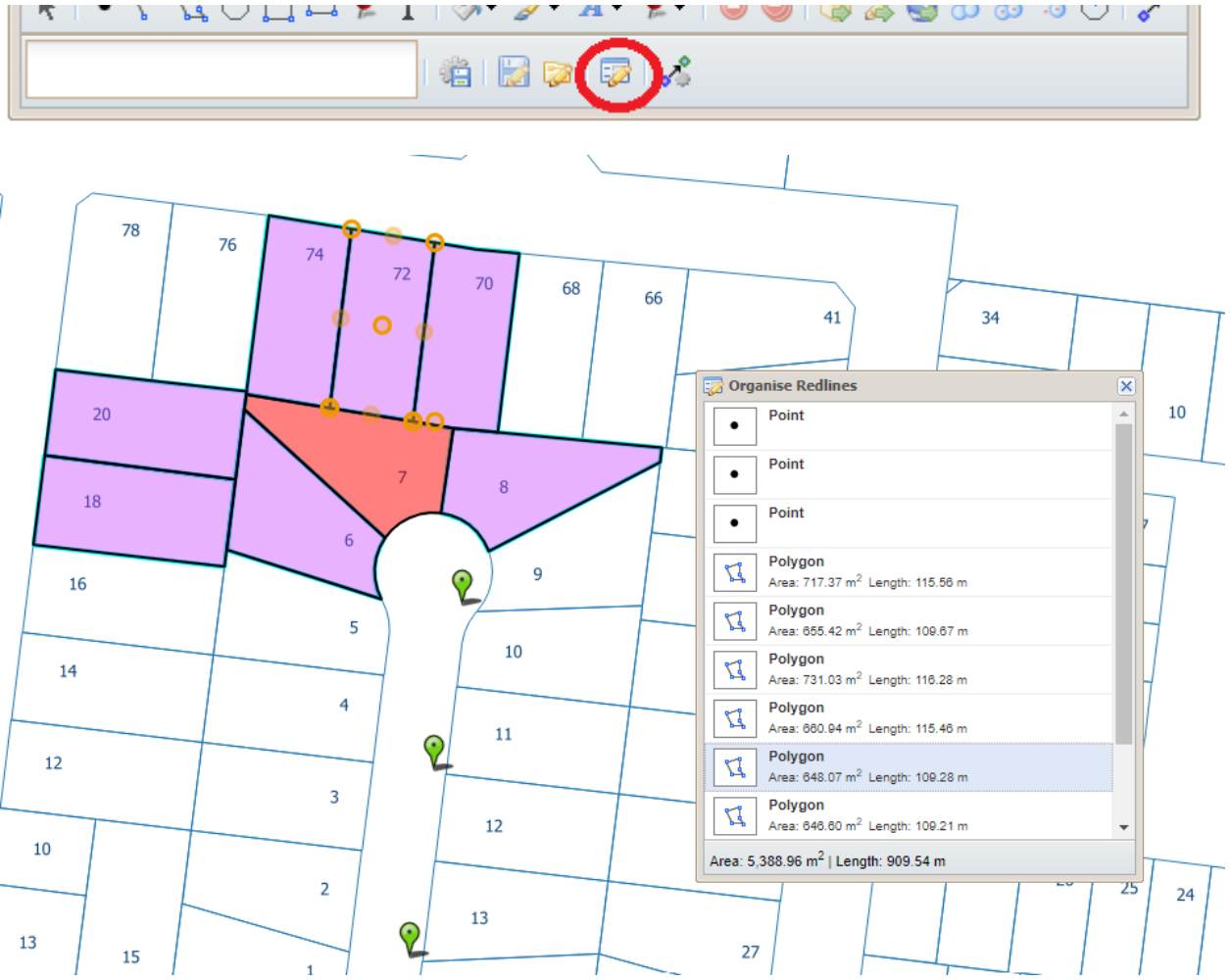
- The *Upload* tool now allows displaying of coordinate values from a spreadsheet.
- This adds another format to the wide range of files that can be uploaded for map display.



#### Redline Tool - Organise Redlines

- A new tool has been added to the Redline tools, the *Organise* tool. It appears on the Redline toolbar by default.
- This tool allows you to organise Redline objects by reordering them to ensure overlapping objects are visible. The window also gives the dimensions of the Redline objects.





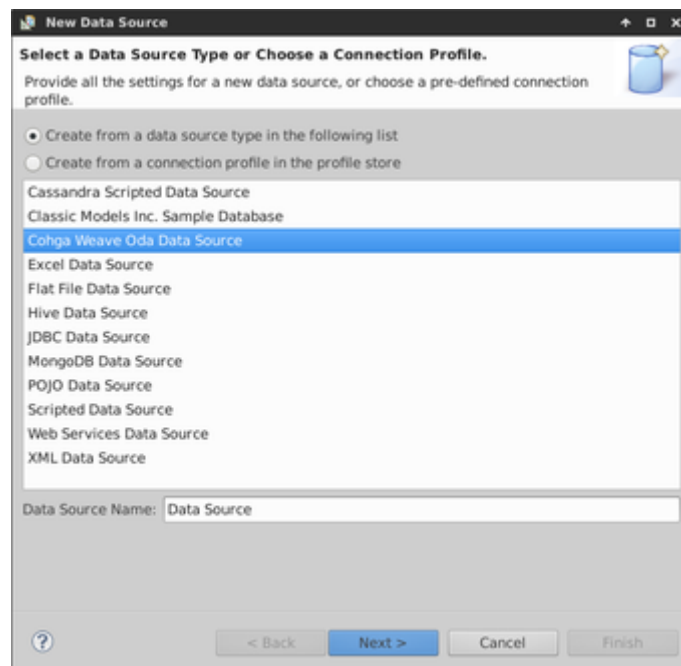
- For more details on the Organise tool refer to the [Weave User Guide Organise tool](#) page.

## BIRT FAQ

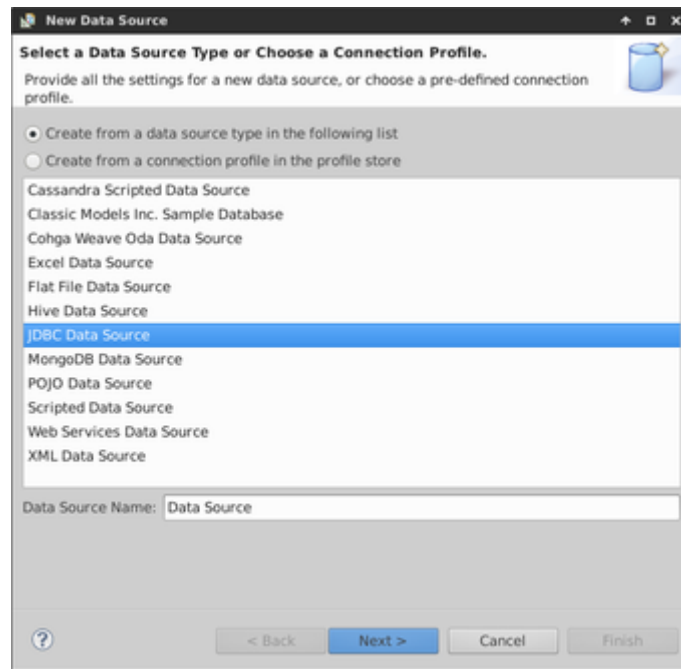
- [Can not connect to database via JDBC](#)
- [Can not load list values](#)

Can not connect to database via JDBC

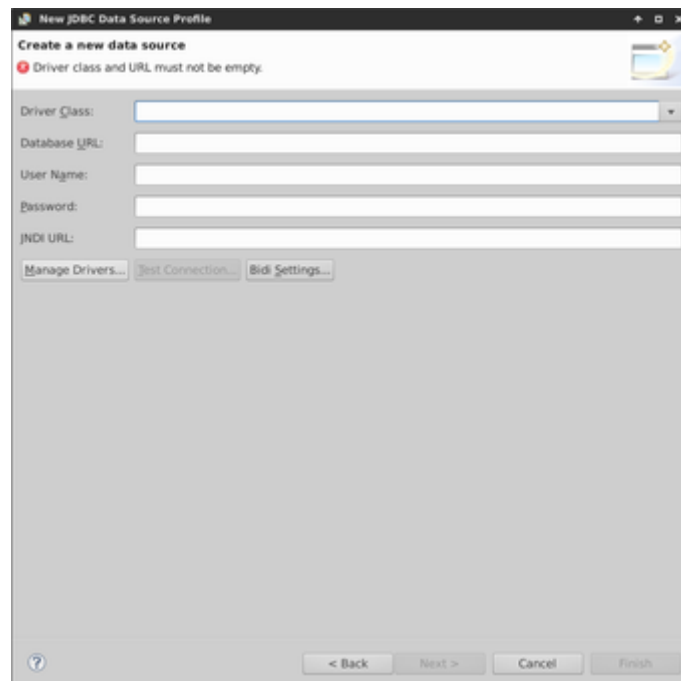
Normally when you're generating data in BIRT, you use a *Cohga Weave Oda Data Source* and have BIRT connect to Weave to generate the required data.



However, there may be situations when you require BIRT to directly connect to the database to retrieve the required data. This is done using a *JDBC Data Source*.



This requires installation of the appropriate JDBC driver to allow BIRT to connect to the database. Such drivers are installed via the *Manage Drivers...* button when adding the new JDBC Data Source.



This will copy the JDBC driver to the correct directory within the BIRT Designer and make it available *while you're designing the report*. What it *won't* do is make the JDBC driver available for the Weave server when *generating the report*.

If you're going to directly connect to a database via JDBC from a BIRT report you also need to manually copy the JDBC driver to the `drivers` directory in the BIRT ODA Driver plugin. Note that this is different from copying the JDBC driver to the `...\platform\workspace\jdbc\` directory, which is where Weave will load the drivers from when it connects to the database (as opposed to BIRT connecting to the database).

The directory name for the BIRT ODA Driver plugin is `...\platform\plugins\org.eclipse.birt.report.data.oda.jdbc_X.Y.Z.vTIMESTAMP\drivers\` (where `X.Y.Z.vTIMESTAMP` will be different depending upon the version of BIRT). If you copy the required JDBC .jar file to that directory BIRT will then be able to load the driver when it needs to connect to the database, it could do this to load values for a list parameter in the report, or to directly generate the data for the report, for example.

Can not load list values

If you've created a report list or combo box parameter and obtain the values dynamically for the parameter via a BIRT JDBC Data Source, as opposed to having the values listed directly in the report or obtaining the values from a different types of BIRT Data Source (for example a Cohga Weave Oda Data Source) but when you try and generate the report within Weave the values for the list/combo box do not display, it might be because the JDBC driver for the database is not available.



There are three locations where JDBC drivers need to be installed to be used by Weave. The first one is in the `...platform\workspace\jdbc\` directory. This is where Weave accesses drivers that it uses when connecting to databases to generate data, perform searches, etc. This is the location where you'd generally copy JDBC drivers to if you want to generate data within Weave from a particular database.

The second and third locations are linked and both related to BIRT. BIRT can also fetch data directly from a database via JDBC rather than always obtaining the data via Weave. In this situation, the appropriate JDBC driver must be made available to the BIRT Report Designer, so it can be used when creating report designs, *and* it must also be made available to the BIRT ODA Driver plugin (it's this step that would be missing if Weave does not display the list values). These two locations are the same, but the former is relative to the BIRT Report Designer installation directory, and the latter relative to the Weave installation directory.

The directory name is `...plugins\org.eclipse.birt.report.data.oda.jdbc_X.Y.Z.vTIMESTAMP\drivers\` (where X.Y.Z.vTIMESTAMP will be different depending upon the version of BIRT).

You need to copy the JDBC driver to the BIRT Report Designer drivers directory, so it can be used when using the designer to create a report. However you also need to copy it to the Weave drivers directory so it can be used by BIRT when generating the report (which includes generating the list of values for the parameters, since you can also directly connect to a database from BIRT to generate the actual report data, not just the parameter values).

## DataSource FAQ

- [Cannot load JDBC Driver class](#)
- [Oracle Errors](#)
- [Unable to obtain jdbc connection](#)

Cannot load JDBC Driver class

## Error

```

Caused by: org.eclipse.birt.report.data.oda.jdbc.JDBCException:
Cannot load JDBC Driver class: oracle.jdbc.OracleDriver
    at org.eclipse.birt.report.data.oda.jdbc.JDBC DriverManager.
findDriver(JDBC DriverManager.java:623)
    at org.eclipse.birt.report.data.oda.jdbc.JDBC DriverManager.
loadAndRegisterDriver(JDBC DriverManager.java:731)
    at org.eclipse.birt.report.data.oda.jdbc.JDBC DriverManager.
doConnect(JDBC DriverManager.java:192)
    at org.eclipse.birt.report.data.oda.jdbc.JDBC DriverManager.
getConnection(JDBC DriverManager.java:158)
    at com.cohga.server.datasource.jdbc.internal.
PooledJdbcDataSource$ConnectionFactory.makeObject
(PooledJdbcDataSource.java:162)

```

## Solution

Make sure that the JDBC driver has been installed in the `... \platform\workspace\jdbc\` directory.

In the case above it is looking for class `oracle.jdbc.OracleDriver` which is contained within the `ojdbc.jar` file, if you're trying to connect to a different database, for example SQL Server or Postgres, the required JDBC driver will be different.

The driver should be downloaded from the web site of the database provider or copied from the installation media of the database.

## Oracle Errors

[Oracle Error for 04031](#) You find that a message has been logged with the following pattern

```
java.sql.SQLException: ORA-XXXXX
```

e.g.

```
java.sql.SQLException: ORA-04031
```

## Solution:

The above message indicates that an error has occurred with the Oracle Database. This could be due to configuration error in the config.xml file or it could be an error thrown by Oracle when communicating with the database. There are many good resources on the web which will explain what each of the error code are and actions required to fix them.

In the above example using Google and searching for "ORA-04031" returns the following page. <http://ora-04031.ora-code.com>

Unable to obtain jdbc connection

## Error

```
com.cohga.server.datasource.jdbc.internal.PooledJdbcDataSource "Unable to obtain jdbc connection"
```

## Solution

The above error logged when Weave cannot create a connection to the Database configured. Unfortunately this error message does not go into any detail as to why the connection could not be created however there will typically be an error that will accompany this error which will explain why the error has occurred. This message is typically logged from the underlying Database itself and may mean that the database is down, there is a network connection problem or the connection settings supplied in the configuration are wrong.



A good way to test if a database is currently working using the JDBC API is by using a third party application to verify the connection settings. This is also a good way to isolate the problem from the Weave environment. Some good tools that can be used are

DBVisualizer: <http://www.dbvis.com/>

SQL Developer: <http://www.oracle.com/technetwork/developer-tools/sql-developer/overview/index.html>

The connection pooling infrastructure will handle creating and disposing connections to databases. Weave will ask the pooling code to create a new connection to the database, if the connection could not be created for some reason the error above will be thrown. Before the error is thrown Weave tries to recreate the connection 3 time waiting for 1 second between attempts.

Generally there is a detailed exception as to why the connection could not be created to the database which should be observed before making any decisions regarding what to do with Weave. If Weave is restarted and the database is still not up the error message will be thrown again.

## SpatialEngine FAQ

- [ArcSDE Errors](#)
- [PropertyDescriptor is null](#)

### ArcSDE Errors

[SDE error -XXX][Error desc=**ESRI\_ARCSDE\_MESSAGE**]

e.g.

[SDE error -110][Error desc=THIS CONNECTION IS LOCKED TO A DIFFERENT THREAD.]

Solution:

When ever there is a message from ArcSDE to indicate that a problem has occurred on the ArcSDE server the follow type of error will be seen in the log file. ArcSdeException: [SDE error ARCSDE\_ERROR\_NUMBER][Error desc=ARCSDE\_ERROR\_MESSAGE] Where the error number defined in SDE (see [http://edndoc.esri.com/arcsde/9.1/capi\\_functions/returncodes\\_incl.htm](http://edndoc.esri.com/arcsde/9.1/capi_functions/returncodes_incl.htm) for more details)

For more detailed information regarding error codes and messages from ArcSDE please refer to the following webpage which generally goes into more detail than the ArcSDE return codes go into. You may need to search around the site for the specific error and note that you may not find the specific error code as not all have been documented. Cohga has not relationship with the author of the page.

<http://bingluo.blogspot.com/2007/12/arcsde-error-codes-reason-and-solution.html>

The one to look out for is the dreaded

[SDE error -51][Error desc=DATABASE LEVEL ERROR OCCURRED.] Underlying DBMS error.

The above message indicates that something has gone very wrong on the database that ArcSDE communicates with. It is best that you contact you ArcSDE administrator and investigate the ArcSDE log files to determine why the error occurred. You may have to enable more fine grained logging in ArcSDE to ascertain the exact problem.

PropertyDescriptor is null

Error:

java.lang.NullPointerException: PropertyDescriptor is null

Solution:

The above error indicates that the configured KEY for the spatial layer is not valid. When changing the key for the layer, watch the console to see if the spatial mapper has been registered correctly. If not Weave will indicate what valid fields are available to be used from the datasource.

## MapEngine FAQ

- [ArcIMS](#)
- [Securing Web Mapping Services \(WMS\)](#)

### ArcIMS

- [Requested Service is not available](#)

**Requested Service is not available**

Error:

AXL Response:

```
<?xml version="1.0" encoding="UTF-8"?>
<ARCXML version="1.1">
  <RESPONSE>
```

```
<ERROR>[ERR0134] Requested Service is not available.</ERROR>
</RESPONSE>
</ARCXML> <com.cohga.server.map.arcims.ArcImsConnection> [Thread-30]
```

**Solution:**

The above messages indicates that the connection to ArcIMS was successful however ArcIMS returned a response that the service is currently unavailable. It is best to open the service in ArcIMS Administrator and refresh the service. A good tool to debug ArcIMS services is

<http://arcscripts.esri.com/details.asp%3Fdbid%3D13053>

Securing Web Mapping Services (WMS)

**Q. Is our proprietary WMS source (e.g. aerial photography) in our public Weave site accessible by public users to exploit?**

A. When a map is requested from the browser it uses Weave own internal message format to communicate with the server. This format is loosely based around the REST standard. To make it harder for people to gain access to the maps from outside Weave we have deliberately not used a well-known standard like WMS or ArcGIS Servers REST API. Weave client URLs do not have the layer name actually listed. When translating the request at the server Weave will use layer identifiers to call the appropriate layer. To make things more secure the layer identifiers can also change over time.

## Support FAQ

### Support Entitlement

A Weave site is entitled to receive all updates of software and documentation to Weave whilst under a current Maintenance contract with Cohga. Twelve months of maintenance is provided in the initial purchase price of Weave and is subject to annual renewal.

Cohga provides email, internet, and phone support services to Weave users that have a current maintenance contract.

### Software Updates

Two types of software update may be required:

1. For a Bug Fix:  
For this type of software update, Cohga can send the site administrator a new version of a Weave bundle that would typically be installed on a test server and once proven, be used to update the production environment
2. For a version update:  
In this case the site administrator will be shipped a complete software installation that comprises the latest version of Weave.

### Upgrading Weave

- ✘ Weave 2.4.16 changes the way user settings are stored and removes the need to backup and restore the user settings when performing an upgrade. The information below about 'Backing up user settings' using `ustorage save` and `ustorage load` is now obsolete if you are running version 2.4.16 or greater. The user settings will be maintained between updates automatically, and will not be effected by cleaning out the `configuration` directory.

## Backing up user settings

Anything the user can save (bookmarks, redlines, etc) will be stored under the Weave installation directory, so they should be exported before the current Weave instance is decommissioned or upgraded.

This must be done from the `osgi` command prompt using the `ustorage` command. The `ustorage` command can save the current user settings to an external file using the following format


```
osgi> ustorage save <filename>
```


and reload the content in the new instance with the `ustorage load` command


```
osgi> ustorage load <filename>
```

You should specify a fully qualified file name as the last parameter in both commands to make it easy for you to locate the exported file, otherwise the file will be exported somewhere under the Weave directory (and if you don't specify a file name it will be named as `ustorage.txt`) but don't worry, the command will print out the name of the file that it is using.

This allows you to backup the current user settings and migrate them to the new Weave instance.

 There is a similar `storage` command that does the same thing for system settings, to ensure a complete migration you should also use the `storage save` and `load` commands with another file.

 Note that you need to be using version 1.1.0 of the `com.cohga.server.user.storage.osgi` bundle for the `ustorage` command, and version 1.1.2 of the `com.cohga.server.storage.osgi` bundle for the `storage` command. These are the versions included with Weave 2.4.0, so if you're running 2.4.0 or later then you already have the `storage` and `ustorage` commands available. They can be manually added to earlier 2.3.x versions of Weave, please contact Cohga for updated bundles if you're running a version earlier than 2.4.0.

 To execute these commands the server needs to either be started using `startup.cmd` (on Windows) or `debug.sh` (on Linux) which provides access to the `osgi` prompt directly. Or, for situations where Weave is started as a background service, `telnet` access must be enabled, allowing the `osgi` console to be remotely connected to using a suitable `telnet` client. Information on the enabling `telnet` access to the [Weave console is available here](#).

## Updating to the new Database user storage option

The method used to store bookmarks and redlines in earlier versions was not robust enough under load, so it has been replaced by one that stores this information in a database.

If you are running version 2.4.16 you are already using this update. If you are running an earlier version you can follow these instructions to update without having to upgrade to 2.4.16.

The updated bundle is available at:  
[com.cohga.server.user.storage.db\\_1.4.2.jar](#)

To use this you must:

- export the existing settings using `ustorage save` at the `osgi` prompt
- stop the server
- remove the older `com.cohga.server.user.storage.osgi` bundle from `platform\plugins`
- install the newer `com.cohga.server.user.storage.db` bundle to `platform\plugins`
- replace `com.cohga.server.user.storage.osgi` with `com.cohga.server.user.storage.db` in `platform\configuration\config.ini`
- clean out the `platform\configuration` directory (except `config.ini`)
- start the server
- import the existing settings using `ustorage load` at the `osgi` prompt.

## Upgrading from 2.3 to 2.4

Upgrading Weave from 2.3 to 2.4 currently requires a fresh installation of the new version and the migration of an existing configuration.

### Installation Sources


There are generally two installation sources for Weave, a CD image `.iso` file (that can either be burnt to a physical CD or mounted directly with the appropriate software) and a `.jar` file which is the actual Weave installer.

Starting the software installation is slightly different if you're installing from a CD than if you're installing from the `.jar` file, but the actual installation process is the same.

The reason for the two different installation media is related to the size of the download, with the installer `.jar` file being significantly smaller than the CD image, but the `.jar` file requires a Java runtime to already be installed on the server, whereas the CD image contains the installer `.jar` file along with a (number of) Java runtime(s) to start it.

Both installation sources end up running the same Weave installer (the afore mentioned installer `.jar` file) and once the installer is started, the installation process is the same for both.

If you're going to be installing to a server that already has Java installed (any Java version 1.4 or above is sufficient) then using the installer `.jar` file directly will save you using unnecessary bandwidth. It may be even quicker to download the installer `.jar` file and a Java runtime (and install it directly) than it would be to download the `.iso` image, since the `.iso` image contains four Java runtimes (32 bit and 64 bit for both Windows and Linux). If you are installing to multiple servers or do not want to install unnecessary software (since the `.iso` installation method doesn't require the installation of the Java runtime just to run the installer) then the `.iso` image may be a better option.

 Another advantage that the CD `.iso` image has over the installer `.jar` file and a reason for its larger size, is the fact that the CD image contains both a 32 and a 64 bit Java runtime for running the installer. The reason for this is that currently the version of the Java runtime that is installed with Weave, as opposed to the one on the CD image used to run the Weave installer, will be 32 bit or 64 bit depending upon the Java runtime that was used to start the Weave installer. This means that you can install the 64 bit version of Weave using the CD image if your server only has a 32 bit Java runtime, or no Java runtime, installed. You can only install the same version as the Java runtime you are already using if you use the installer `.jar` file directly. This may be changed in the future.

### Installing from a CD image `.iso` file

Launching the Weave installer from the CD image can be done by running the `install.cmd` or `install64.cmd` batch files on Windows or `install.sh` or `install64.sh` scripts on Linux. These scripts are in the root directory of the CD image. These scripts are used to launch the Weave installer `.jar` file using one of the Java runtimes available on the CD, with the `install` versions installing a 32 bit version of Weave, and the `install64` version installing a 64 bit version of Weave.

### Installing from an installer `.jar` file

To install Weave directly from the installer `.jar` file requires starting the `.jar` file using the Java runtime already installed on the server. If you already have an older version of Weave installed on the server then you can use the JDK that was installed with that existing version.

Depending upon the file association configured for your server, you may be able to start the Weave installer `.jar` file directly (by double clicking on it in Windows Explorer for example) but if this does not work then you should open a command prompt and run the installer directly using the following command.

```
java -jar weave-installer.jar
```

**i** In this text when directly referring to the Weave installer `.jar` file it will be presented as `weave-installer.jar`, but the file name may be different, for example it may actually be `weave-installer-latest.jar` or `weave-installer-2.4.0.jar` or even `weave-installer-2.4.0-20110216.jar`

You may need to fully qualify the `java` command if it isn't already in your `PATH`, for example if you're using an existing Weave installation it may be

```
c:\weave\jdk\bin\java -jar weave-installer.jar
```

or for Linux

```
/opt/weave/jdk/bin/java -jar weave-installer.jar
```

### The installation process

Once started, the installer takes you through a number of steps:

- Welcome screen
- License agreement
- Installation directory
- Software selection
- Port selection
- Telnet access
- Software installation
- Completion screen

The Welcome screen and License agreement should be self explanatory, but the Installation directory can be a little tricky since we are performing an update, and generally when performing an update using the Weave installer, you do not want to install on top of an existing Weave installation, so you need to ensure that you install to a new directory.


**i** As an alternative to installing to a new directory you can rename an existing Weave installation, to get it out of the way, and install to the same location, just make sure you remove any existing Windows service (using the `remove-service.cmd` batch file) before renaming the directory.

**Warning:** Once you have settled on an installation directory, choose which packages to install. Do not just install all of the packages, especially the Additional Components. It may be tempting to just click all of the check boxes but this is almost certainly not what you want. This will install components that you will never need and don't use. It will add additional memory and performance overheads that you do not want and it will install multiple versions of bundles where there should only be one of that type.

**i** During the package selection you can see if Weave is installing the 32 or 64 bit version by expanding the Java Runtime pack. If the included pack references `x86` then it is the 32 bit version, and if it references `x86_64` then it is the 64 bit version.

Since we are performing an update to an existing Weave instance, you should know what additional components you had installed, and the process will be simpler if you first install the new Weave instance using the same components as you had before (It is possible to re-run the installer later and add new components if you want to once you have your existing Weave configuration up and running with the new Weave version that we are installing).

You should the expand additional components and *closely* examine the description for each item before selecting it for installation. This is because some components require other components to also be installed (and currently this isn't done automatically) and other components require you to choose *one* from a number of possible alternatives.

 This is probably a good time to mention custom bundles that you may have installed in your existing Weave instance. These are bundles that are added to Weave manually and may have provided functionality just for your site, or functionality that is not included as part of the core Weave installation. Hopefully these components are now available under the Additional Components section of the installation packages but if they are not then you will need to copy these bundles from the existing installation to the new installation after the new installation process has been completed.

If you choose to install Jetty then you will be given the choice of what port numbers Jetty will listen on and whether you'll be provided with access to the OSGi console via telnet (if you don't install Jetty then you'll need to install your own Web application server, such as Tomcat or WebSphere, and manage this yourself).


**It is important that if you want to continue to run your previous Weave instance, that you choose different port numbers when asked.** If you are replacing an existing instance, or you intend to only run one at a time, then you can use the same port numbers.

#### Migration

Once you have finished the installation you need to migrate your existing Weave configuration to the new instance, and copy over any custom plugins, before you start the new Weave instance.

This process involves renaming the new `workspace` directory and copying over the one from your existing instance. You could just delete the new `workspace` directory, but by renaming it you have the option of comparing the old and new version of some of the supplementary files that are included there.

The next migration step is to use the `ustorage` (and `storage`) commands to import the user settings from the previous instance.

 If you're replacing a newly installed workspace with an old production one you may need to copy the `org.hsqldb.hsqldb_2.2.5.jar` file from the old workspace `jdbc` directory to the new one. Otherwise you'll receive "**java.lang.ClassNotFoundException: org.hsqldb.jdbc.JDBCDriver**" errors.

#### Testing

Once the previous workspace is copied to the new instance it should be ready to test and no changes to any configuration files should be required.

It is advisable to do this initial test by starting Weave from the command line, via `startup.cmd` or `debug.sh`, so that you can see log messages immediately, but aware that if Weave is then installed as a service and run as a different user, then permission problems may occur. This is generally an issue when the instance is first started as a service and then later started via the command line. It has been mentioned in case you would prefer to install it as a service immediately and do the testing with Weave running as a service.

#### Support

The system should now be running as it was before, but sometimes there are going to be issues and Cohga support is the place to go, as well as the trouble shooting section of this wiki.

## Upgrading from 2.4.x to 2.4.y


As of the release of Weave 2.4.2 a smaller `weave-updater.jar` file is available that provides an installer that includes only the changes since 2.4.0 and the current release. As at Weave 2.4.2 the size of this file was 12MB, which is considerably smaller than a new installer `.jar` or `.iso` file (by 2.4.10 it's grown to 55MB, still smaller than the other options).


This updater will be updated as new releases are available to include only the changes required to get a 2.4.x version to the latest version, and will save considerable bandwidth and time when trying to update to the latest Weave version. Running the updater on any previous version of Weave will result in the instance of Weave having exactly the same set of bundles as a full new installation of that latest version.

Running the Weave updater is the same as the Weave installer `.jar` file (see the notes above about **Installing from an installer .jar file**) but will provide a cut down list of items that you can update which depends on what is already installed in the Weave instance that the update is being applied to, and what has changed between the latest updater and Weave 2.4.0. This list is only of components that have changed between 2.4.0 and the latest version that will be included in the updater. Furthermore, only functionality that was previously installed via the Weave installer will be available to update via the updater (to install new functionality that wasn't installed initially the original Weave installer should be used and the updater applied after).

The updater will install the newest versions of any plugins that were updated since the existing instance was installed, but it will currently not remove older versions of plugins (but this is ok because the next startup of the Weave instance will only utilise the latest version of any multiple version plugins to be installed). The updater will also update the `config.ini` as appropriate for components installed as part of the Weave installation process, and include custom bundles that were previously manually added to the `config.ini` file.

The latest updater can be downloaded from <http://cohga.com/weave/weave-updater-latest.jar>, or specific versions from <http://cohga.com/weave/weave-updater-2.4.x.jar>, where x should be replaced with the specific release number (starting from 2)

 There has been an incident where Weave would not start correctly after applying an update to an early version of Weave 2.4, but this was resolved by removing any old versions of plugins and manually cleaning out the `weave\platform\configuration` directory (i.e. deleting everything in that directory except the `config.ini` file) then restarting. The Weave updater will be updated (post 2.4.10) to remove old versions of plugins on an update to try and avoid this happening in the future).

 Removing old versions of plugins involves ensuring that there is only one version of any given plugin, for example if there is a `com.cohga.client.weave.main_2.11.8.jar`, a `com.cohga.client.weave.main_2.14.12.jar` and a `com.cohga.client.weave.main_2.18.14.jar` file

then the `com.cohga.client.weave.main_2.11.8.jar` and `com.cohga.client.weave.main_2.14.12.jar` files should be deleted leaving just `com.cohga.client.weave.main_2.18.14.jar`.

However, if the major version number is different, for example 2 is the major version number in the above example, then the latest versions of both bundles should be kept. So if there was also a `com.cohga.client.weave.main_1.1.4.jar` file then that should be kept along with `com.cohga.client.weave.main_2.18.14.jar`. Although in practice this will rarely of ever actually happen.

## Level of support

Cohga provides email, internet, and phone support during normal business hours on Monday through Friday (AEST).

Current experience shows that once Weave is correctly installed and configured in a production environment then the system is stable with the current support level being adequate.

Cohga is open to discussion and negotiation with users that may have a mission critical system and who may require 24 hour support each day of the year.

## Support Process

Users under maintenance are provided a login account to the Cohga on-line support system at <http://cohga.helpserve.com>

The user may lodge support requests by email to [support@cohga.com](mailto:support@cohga.com), or interactively on-line, as well as monitor the status of their support tickets. The user that logs into the on-line support site may be assigned a role where the user may see all tickets associated with their site, or only those tickets submitted by that user. The setting up of this role is done in consultation between Cohga and the Weave user's site administrator.

On receipt of a support request, the Cohga support system issues an email notification to the person submitting the request along with the appropriate new ticket number. This notification usually occurs within thirty minutes.

The support tracking system issues a notification to the support team and one of the team is assigned responsibility for acting on the support request. The support team responds to tickets usually within two hours during normal business hours on Monday through Friday (AEST).

Support requests may be lodged at any time however under the current support level, action by the Cohga support staff is limited to normal business hours Monday through Friday. This can be changed on negotiation with Cohga.

The person who submitted the support request can track progress of their ticket(s) by logging in to the on-line support system.

The Cohga support staff are from the Weave Design and Development team in Melbourne and Adelaide Australia.

## Weave 2.5 FAQ

### What's new in Weave 2.5

#### Libraries

Primarily the intent of this update is to provide updated versions of all of the libraries that underpin Weave. This includes the following updates:

Library	Old Version	New Version
Jetty	6	8
BIRT	2.6	4.3
Eclipse OSGi	3.6	3.9
GeoTools	2.6	10.4
Ext.js	2.3	3.4
Apache MQ	5.1	5.9
Apache Lucene	2.9	3.6
Apache Commons Pooling	1.3	1.6
Apache Commons File Upload	1.2	1.3
Apache Commons Collections	1.8	1.9
Apache Commons Pool	1.3/1.5	1.6
Apache POI	3.8	3.10

#### Configuration

The following changes must be made to a configuration that's being updated from Weave 2.4 to Weave 2.5

## Messaging

If you're using the vehicle tracking extension then you need to add the following line to the client config for any clients that you want to include the vehicle tracking in.

```
<plugin id="weave.messaging" />
```