HiPortfolio

Enhancement Note

D06262-C

Business Intelligence - Phase 2

Versions	3.13, M.3	
Modules	Business Intelligence (BI)	Admin Dashboard (AD)
	Quiesce (QS)	

This enhancement introduces the Business Intelligence (BI) module to HiPortfolio/3. This module allows you to construct highly personalised presentations of HiPortfolio/3 data through the use of subject-specific visualisation components that can be dragged onto report dashboards.

DST GLOBAL SOLUTIONS

Contents

Glossary

Term	Definition
Visualisation Components	Data aware components for use in the Dashboard Manager. These components are used to create real-time reports that can retrieve data from the HiPortfolio/3 database using ad hoc SQL queries.

Summary

This enhancement introduces the Business Intelligence (BI) module to HiPortfolio/3. This module allows you to construct highly personalised presentations of HiPortfolio/3 data through the use of subject-specific visualisation components that can be dragged onto report dashboards.

This enhancement:

- introduces the Business Intelligence (BI) module to the HiPortfolio Web UI
- changes the existing Admin Dashboard application to be a component of the HiPortfolio Web UI application
- enables the HiPortfolio Web UI to be accessed when the Business Intelligence (BI) module or the Admin Dashboard (AD) module is active

For details of this enhancement, see:

HiPortfolio Web UI

Using the HiPortfolio Web UI

Admin Dashboard

Business Intelligence

HiPortfolio Web UI Administration

HiPortfolio Web UI

This enhancement introduces the HiPortfolio Web UI. It is used to visually display information retrieved from HiPortfolio/3 and consists of the following components:

- Admin Dashboard
- Business Intelligence

For more information see:

Using the HiPortfolio Web UI

Admin Dashboard

Business Intelligence

HiPortfolio Web UI Administration

Using the HiPortfolio Web UI

The following topics explains common HiPortfolio Web UI features and how to find your way around the system.

See the following topics:

HiPortfolio Web UI Login

Main Page

Menu Bar

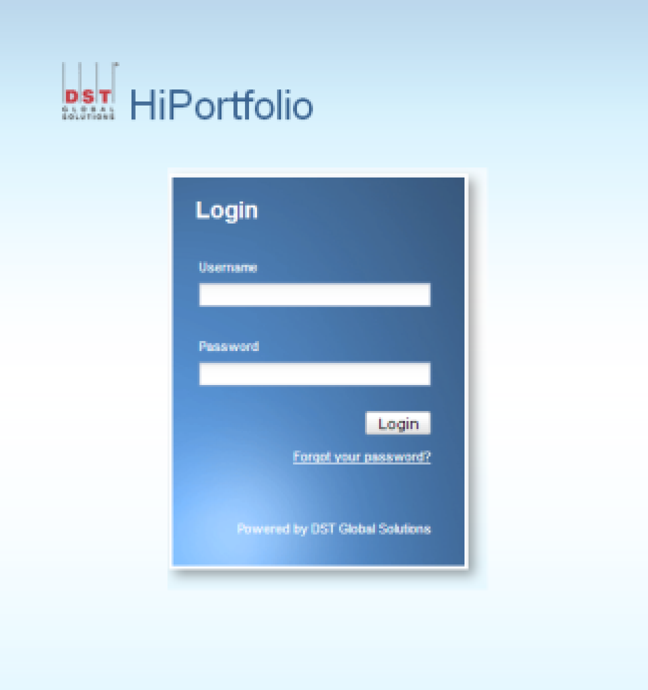
Configuration

HiPortfolio Web UI Login

The HiPortfolio Web UI is accessed by logging in using the **HiPortfolio Web UI Login** screenaccessible from your web browser. You must have a valid HiPortfolio/3 user ID to access the HiPortfolio/3 Web UI.

Note

If you do not know the HiPortfolio Web UI Login URL, contact your system administrator who is responsible for configuring the URL.



Fields

Username

Description	A unique username that identifies you as a HiPortfolio/3 user.
Mandatory	Yes
Type	Alphanumeric
Valid settings	A valid username
Default setting	The same username you use to log into HiPortfolio/3.

Password

Description	The valid password associated with the entered username.
Mandatory	Yes
Valid settings	A valid password. Passwords are case sensitive.

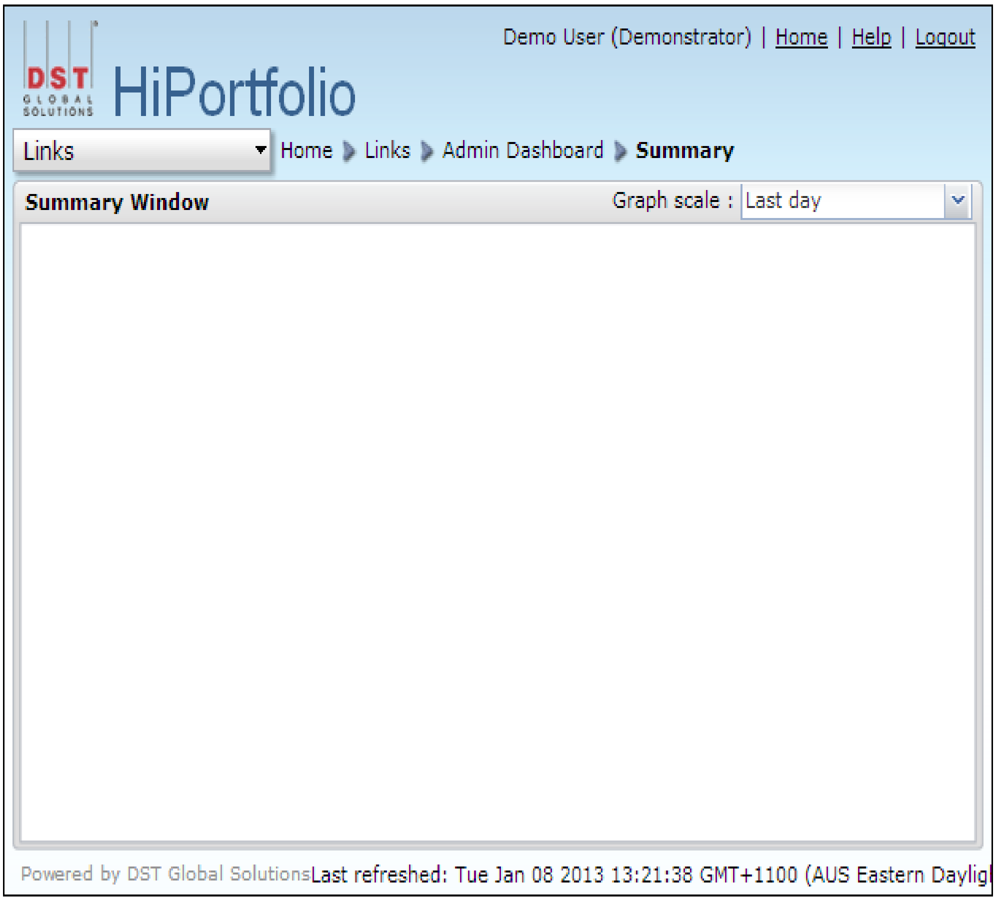
Note

You only have a certain amount of maximum password attempts. This number is configured by your system administrator. HiPortfolio Web UI as well as HiPortfolio/3 will be disabled if this number is exceeded and the following error message will be displayed:

Number of login attempts exceeded. Your account is disabled, please contact your system administrator.

Main Page

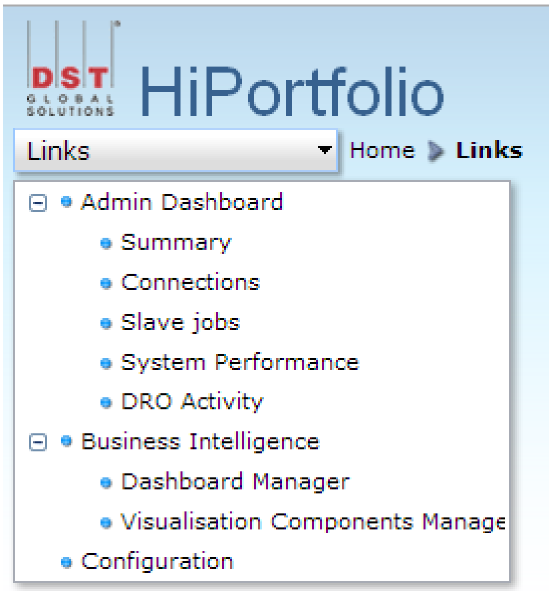
When logging into the HiPortfolio Web UI, the main page opens. Your screen's layout depends on the page defined by your system administrator.


DST GLOBAL SOLUTIONS
HiPortfolio
Demo User (Demonstrator) | Home | Help | Logout
Links
Home ▸ Links ▸ Admin Dashboard ▸ Summary
Summary Window
Graph scale : Last day
Powered by DST Global Solutions

Last refreshed: Tue Jan 08 2013 13:21:38 GMT+1100 (AUS Eastern Daylig

Links Menu

The **Links Menu** is context sensitive and displays the options available, dependent on your user permissions and module statusses. You can access the **Links Menu** by selecting the **Links** option from the main toolbar.



The following options can be accessed from the **Links menu**:

Admin Dashboard

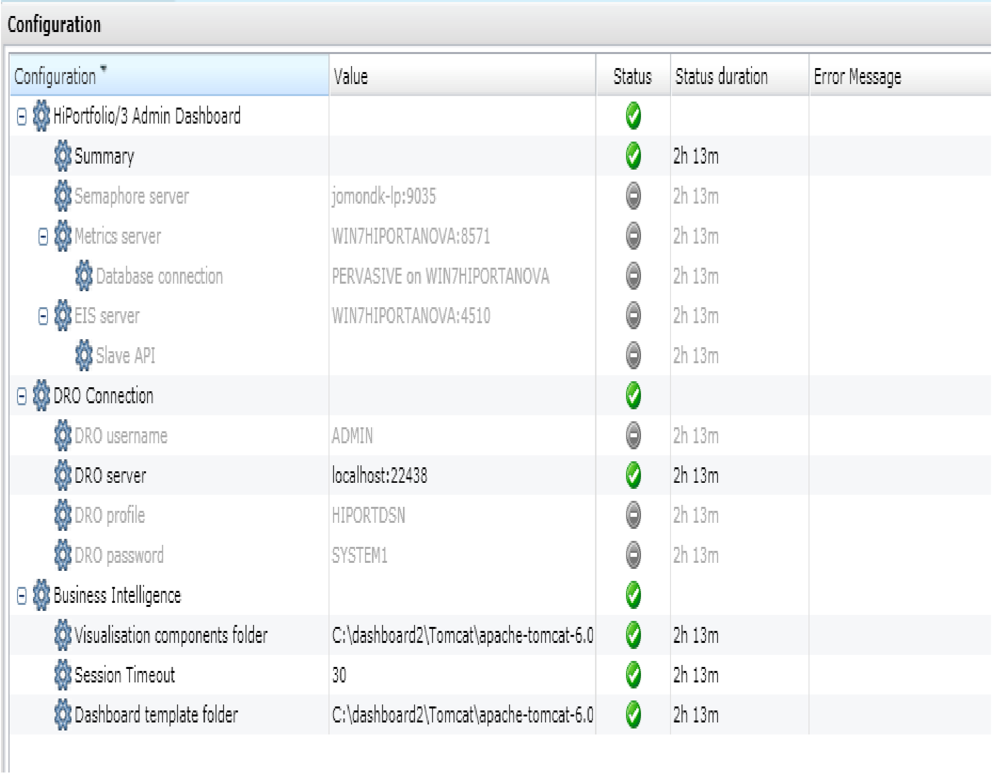
Business Intelligence

Configuration

Configuration

This enhancement changes the **Configuration Window** name to **Configuration** and changes the window elements as follows. It:

- adds four new fields
- changes four existing fields to allow them to be editable
- makes the *Value* column editable
- moves the **Configuration** window from the Admin Dashboard to the HiPortfolio Web UI



Configuration

Configuration	Value	Status	Status duration	Error Message
⊟ HiPortfolio/3 Admin Dashboard		✓		
Summary		✓	2h 13m	
Semaphore server	jomondk-lp:9035	⊖	2h 13m	
⊟ Metrics server	WIN7HIPORTANOVA:8571	⊖	2h 13m	
Database connection	PERVASIVE on WIN7HIPORTANOVA	⊖	2h 13m	
⊟ EIS server	WIN7HIPORTANOVA:4510	⊖	2h 13m	
Slave API		⊖	2h 13m	
⊟ DRO Connection		✓		
DRO username	ADMIN	⊖	2h 13m	
DRO server	localhost:22438	✓	2h 13m	
DRO profile	HIPORTDSN	⊖	2h 13m	
DRO password	SYSTEM1	⊖	2h 13m	
⊟ Business Intelligence		✓		
Visualisation components folder	C:\dashboard2\Tomcat\apache-tomcat-6.0	✓	2h 13m	
Session Timeout	30	✓	2h 13m	
Dashboard template folder	C:\dashboard2\Tomcat\apache-tomcat-6.0	✓	2h 13m	

Note

For details on each of the configuration items, see the *Installation guide* (H3WebUIConfigDeployGuide.pdf).

Columns

Configuration

Description This column displays the name of the configuration item.

Value

Description This column displays any values assigned to the configuration item. These can be modified either by double clicking the value column or within the dashboard.runtime.properties file.

Status

Description This column displays the status of the server being monitored.

Valid values

the data source is running as expected.

the data from the data source may be incomplete or improperly loaded.

the data source has raised an error.

the data source is loading.

the data source is disabled in the configuration file.

Status Duration

Description This column displays the length of time the server has been in its current state.

Error Message

Description This column displays the last error message received for the server since the status update.

New Fields

DRO username

Description	The user name used to connect to the DRO server.
Mandatory	Yes
Type	Alphanumeric

DRO password

Description	The password used to connect to the DRO server
Mandatory	Yes
Type	Alphanumeric

Visualisation components folder

Description	The folder location used to store the visualisation components.
Mandatory	Yes
Type	Alphanumeric

Dashboard template folder

Description	The folder location used to store the created dashboards.
Mandatory	Yes
Type	Alphanumeric

Changed Fields

Semaphore Server

Description	The semaphore server DNS name and port number.
Location	**HiPortfolio Web UI**
Mandatory	Yes
Type	Alphanumeric
Messages	Mandatory field is blank
	Server error - Server must exist
	Server error - Server must be running

Metrics Server

Description	The Metrics Server DNS name and port number.
Location	**HiPortfolio Web UI**
Mandatory	Yes
Type	Alphanumeric
Messages	Mandatory field is blank
	Server error - Server must exist
	Server error - Server must be running

EIS Server

Description	The EIS Server DNS name and port number.
Location	**HiPortfolio Web UI**
Mandatory	Yes
Type	Alphanumeric
Messages	Mandatory field is blank
	Server error - Server must exist
	Server error - Server must be running

DRO Server

Description	The DRO server DNS name and port number.
Location	**HiPortfolio Web UI**
Mandatory	Yes
Type	Alphanumeric
Messages	Mandatory field is blank
	Server error - Server must exist
	Server error - Server must be running

Admin Dashboard

The Admin Dashboard is already available as a standalone application. This enhancement makes the Admin Dashboard a component of the new HiPortfolio Web UI.

To enable the existing Admin Dashboard to become a component of the HiPortfolio Web UI, the following common page changes have been made:

- the tab name **HiPortfolio/3 Admin Dashboard** has been changed to **HiPortfolio**
- the URL *.../hiport-admindashboard*has been changed to *.../hiportfolio*

The Admin Dashboard is no longer available as a standalone application.

Business Intelligence

This enhancement introduces the Business Intelligence (BI) module to HiPortfolio/3. This module allows you to construct highly personalised presentations of HiPortfolio/3 data through the use of subject-specific visualisation components that can be dragged onto report dashboards.

HiPortfolio/3 data can be drawn from any of the hundreds of Data Retrieval Objects (DROs) that are available to users using SQL.

For more details see:

Dashboard Manager

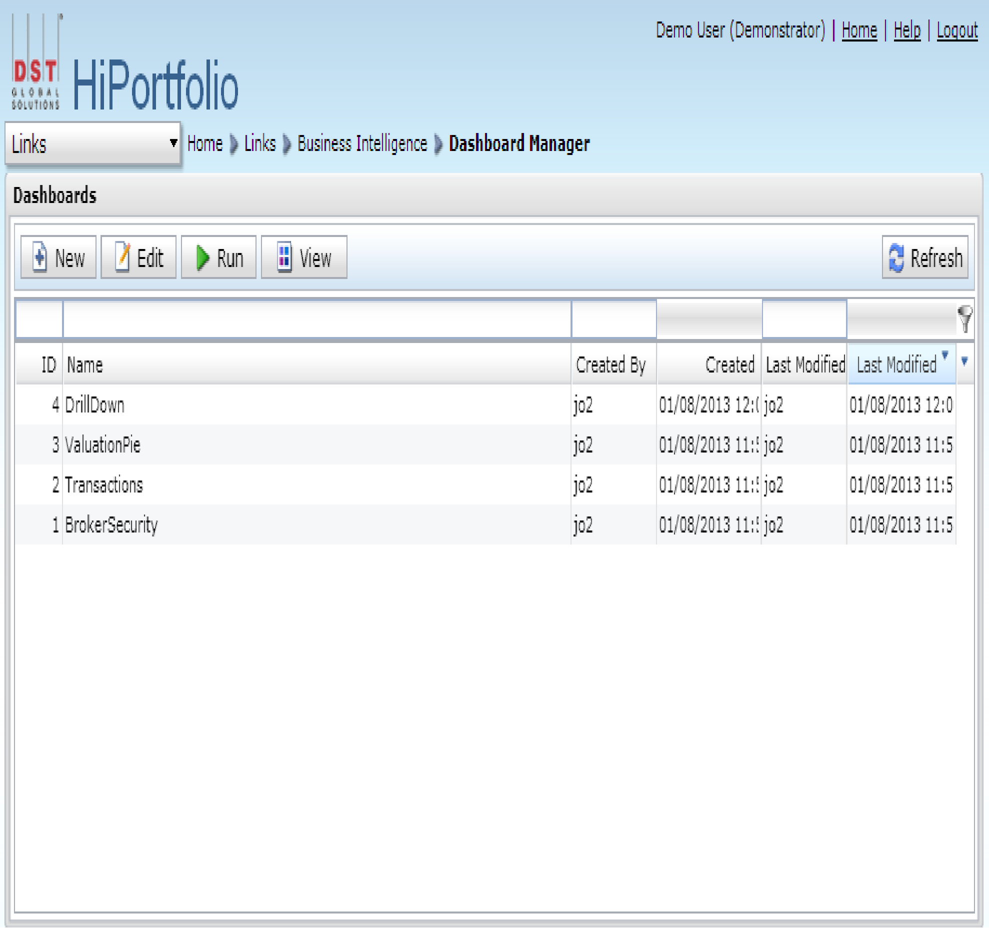
Visualisation Components Manager

Using the Dashboard Editor

Dashboard Manager window

In the **Dashboard Manager** window you can add, edit, run or view a report dashboard.

Home > Links > Business Intelligence > Dashboard Manager



ID	Name	Created By	Created	Last Modified	Last Modified
4	DrillDown	jo2	01/08/2013 12:0	jo2	01/08/2013 12:0
3	ValuationPie	jo2	01/08/2013 11:5	jo2	01/08/2013 11:5
2	Transactions	jo2	01/08/2013 11:5	jo2	01/08/2013 11:5
1	BrokerSecurity	jo2	01/08/2013 11:5	jo2	01/08/2013 11:5

Columns

All columns are alphanumeric.

ID

Description	Dashboard ID number assigned sequencially to each created dashboard.

Name

Description	Dashboard name as defined by the dashboard creator.

Created By

Description	User ID of the dashboard creator.

Created

Description	Date and time when the dashboard was created.

Last Modified By

Description	User ID of the last user to modify the dashboard.

Last Modified

Description	Date and time when the dashboard was last modified.

Buttons

The buttons displayed on the Dashboard Manager are dependent on the permissions granted in the policy.xml. You will only be able to see and access the functions to which you have permission.

You can also access the *View, Run* and *Edit* functions from the right-click menu.

New

Description	Click this button to open the Dashboard Editor and create a new report dashboard.

Edit

Description	Click this button to open the Dashboard Editor and edit the widget without running the dashboard first. This button is only active when you have selected a dashboard to edit.

Run

Description	Click this button to open the Dashboard Editor and run the selected dashboard. This button is only active when you have selected a dashboard to run.

View

Description	Click this button to open the selected dashboard in view mode. This button is only active when you have selected a dashboard to view.

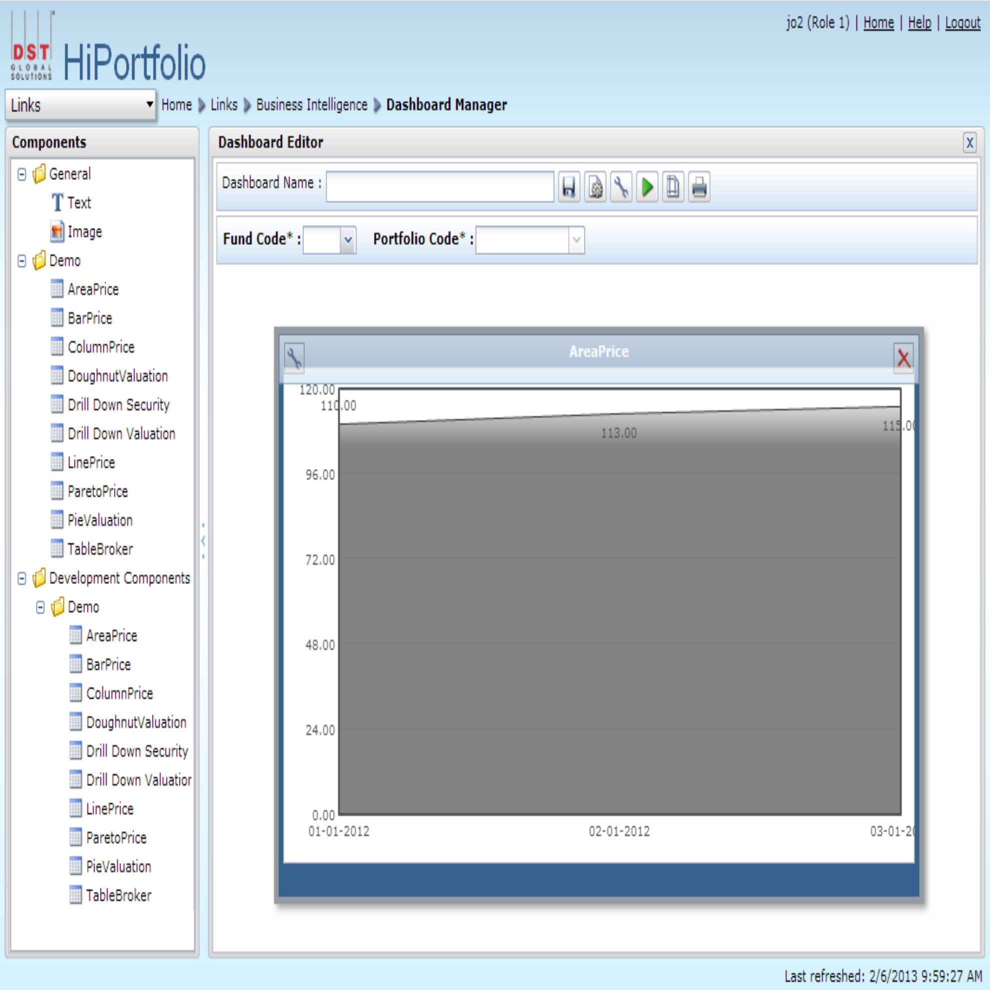
Refresh

Description	Click this button to refresh the list of dashboards available to you.

Dashboard Editor

The Dashboard Editor is used to create and edit dashboards.

Home > Links > Business Intelligence > Dashboard Manager



The Dashboard Editor consists of a Components Toolkit and a Dashboard Editor frame.

Frame

Components

Description	A toolkit of available pre-defined and user-defined components that can be placed on the *Dashboard Editor* frame.
Type	Drag and drop

Fields

Dashboard Name

Description	This field specifies the dashboard name.
Mandatory	Yes
Type	Alphanumeric, 30

Buttons

Save

Description	Click this button to save the dashboard.
Other information	This button cannot be selected until you have: - added or changed the content of the dashboard, or - specified or changed the dashboard name.

Show/hide parameter toolbar

Description	Click this button to either show or hide the parameter toolbar that contains the *Fund Code* and *Portfolio Code* drop-down menus.

Configure all components

Description	Click this button to open the configuration view for all components on the dashboard.

Run all components

Description	Click this button to run all components on the dashboard using the current configuration parameters.

Show/hide page setup

Description	Click this button to open the **Page Setup** window to view a print preview of the dashboard.
Other information	Printing page setup

Open print friendly view

Description	Click this button top opens the **Print Options** window.
Other information	Print Options window

Parameter Toolbar

Fund Code

Description	The code of the fund for which you want to display data in the dashboard components.
Mandatory	No
Type	Drop-down
Default setting	The first fund you have authority to will be displayed by default.
Other information	This field is only displayed if the components within the dashboard display fund specific data.

Portfolio Code

Description	The code of the portfolio for which data is displayed in the dashboard components.
Mandatory	No
Type	Drop-down

Default Setting	The first portfolio for which you have authority will be displayed by default.
Other information	This field is only displayed if the components within the dashboard displays portfolio specific data.

Components Toolkit

The following component types can be dragged onto the *Dashboard Editor* frame from the components toolkit:

- Image Component
- Text Component
- User defined Components

Image Component

The **Image Component** window opens when you drag the image component from the Component Toolkit onto the Dashboard Editor. This allows you to add an image to a dashboard.

Home > Links > Dashboard Manager > New > image

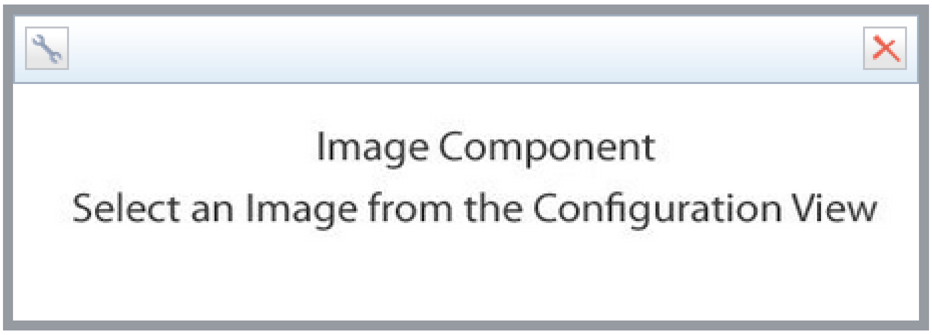


Image Component

Select an Image from the Configuration View

Buttons

Configure component

Description	Click this button to open the Image Component .
Other information	This button becomes available when you hover the mouse over the component.

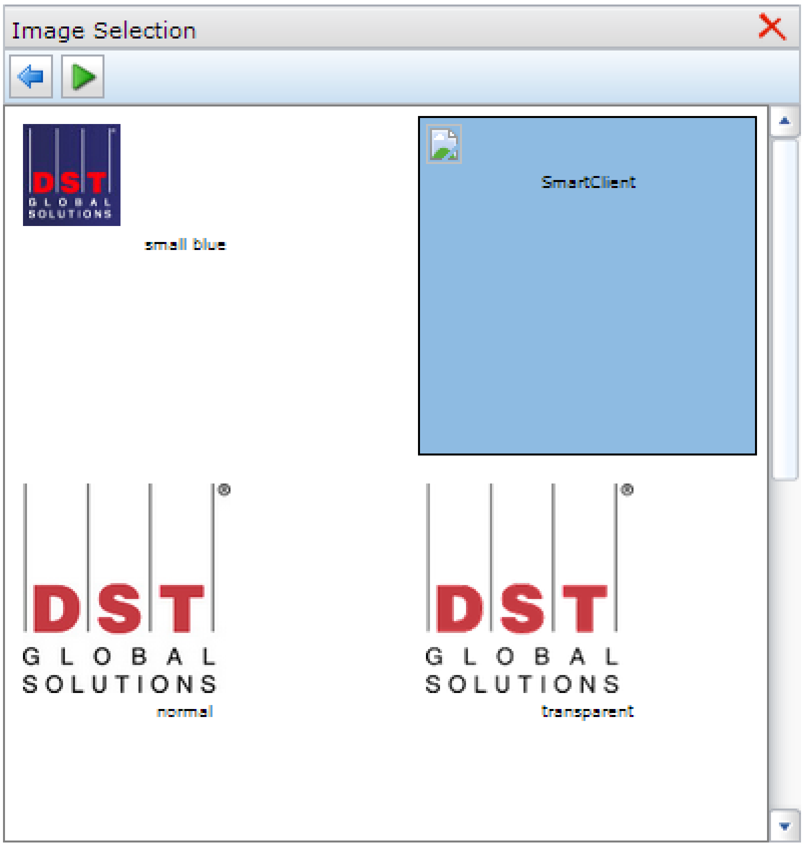
Delete

Description	Click this button to delete the selected image component from the dashboard.

Image selection window

The **Image Selection** window opens when you click the **Configure component** button in the **Image Component** window, allowing you to select an image to display on the Dashboard Editor.

Home > Links > Dashboard Manager > New > image > configuration



Buttons

Back

Description	Returns you to the component view

Run

Description	Runs the image component using the current parameters

Delete

Description	Deletes the component from the dashboard

Text Component

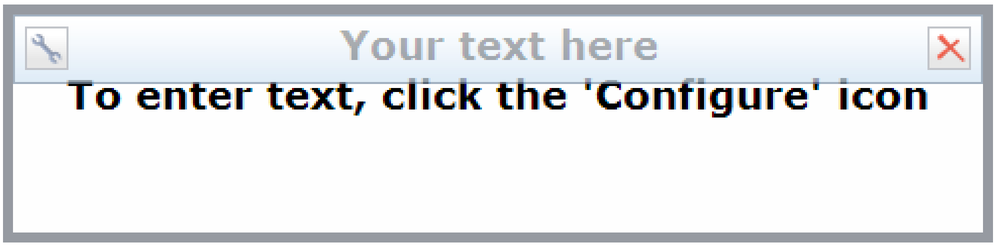
The **Text Component** window opens when you drag the text component from the Component Toolkit onto the *Dashboard Editor* frame.

The configuration of the text component is slightly different from other components in that you can enter any text you need. This allows you to enter further information about the other components on the dashboard or to add a title to the dashboard.

The following text formats are available:

- **Title**
- **Subtitle**
- Text

Home > Links > Dashboard Manager > New > text



Your text here

To enter text, click the 'Configure' icon

Buttons

Configure component

Description	Click this button to open the Text Content window .
Other information	This button becomes available when you hover the mouse over the component.

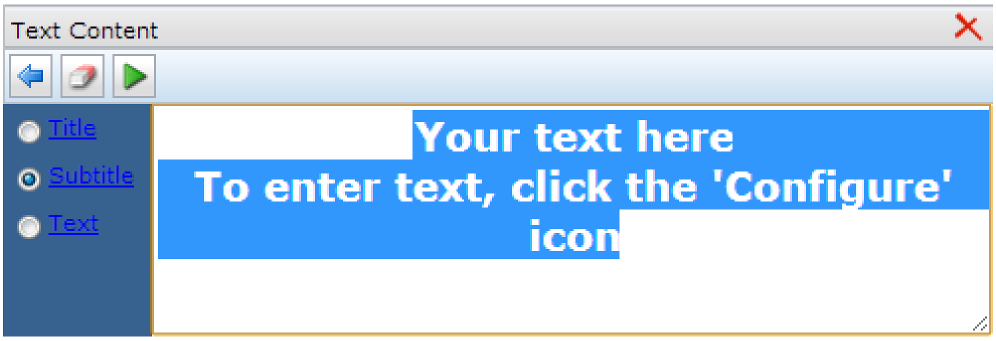
Delete

Description	Click this button to delete the selected text component from the dashboard

Text Content window

The **Text Content** window opens when you click the **Configure Component** button in the **Text Component** window. This window allows you to enter text that will be displayed on the Dashboard Editor.

Home > Links > Dashboard Manager > New > text > configuration



Your text here

To enter text, click the 'Configure' icon

Options

Title

Description	Formats the text to display in the Title format.

Subtitle

Description	Formats the text to display in the Subtitle format.

Text

Description	Formats the text to display in the Text format.

Buttons

Back

Description	Returns you to the component view.

Clear

Description	Clears all the fields in the text component configuration view.

Run

Description	Runs the text component using the current parameters.

Delete

Description	Deletes the component from the dashboard.

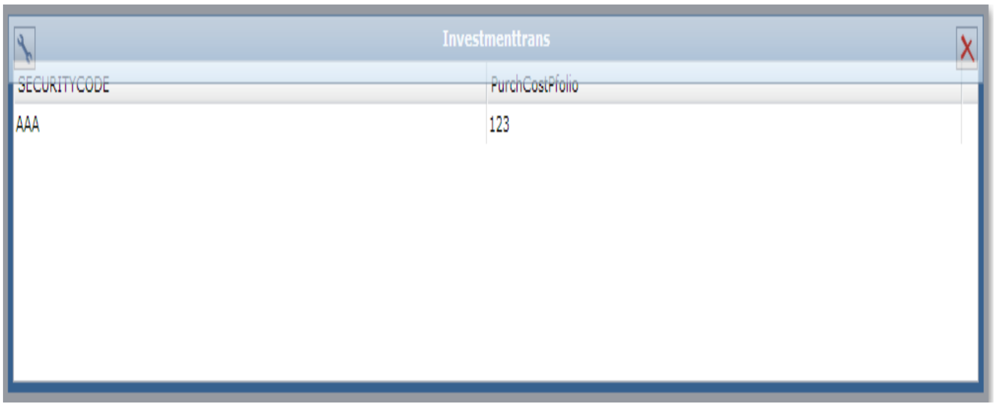
User Defined Components

The **User Defined Components** window opens when you drag an imported component from the Component Toolkit onto the *Dashboard Editor* frame. In the **Component View** window you can view the data in the component and export the data from a component to CSV.

Note

The presentation of the data is defined in the component xsd file.

Home > Links > Dashboard Manager > New > user defined component



SECURITYCODE	PurchCostPfolio
AAA	123

Buttons

Configure component

Description	Click this button to open the Component configuration window.
Other information	This button becomes available when you hover the mouse over the component.

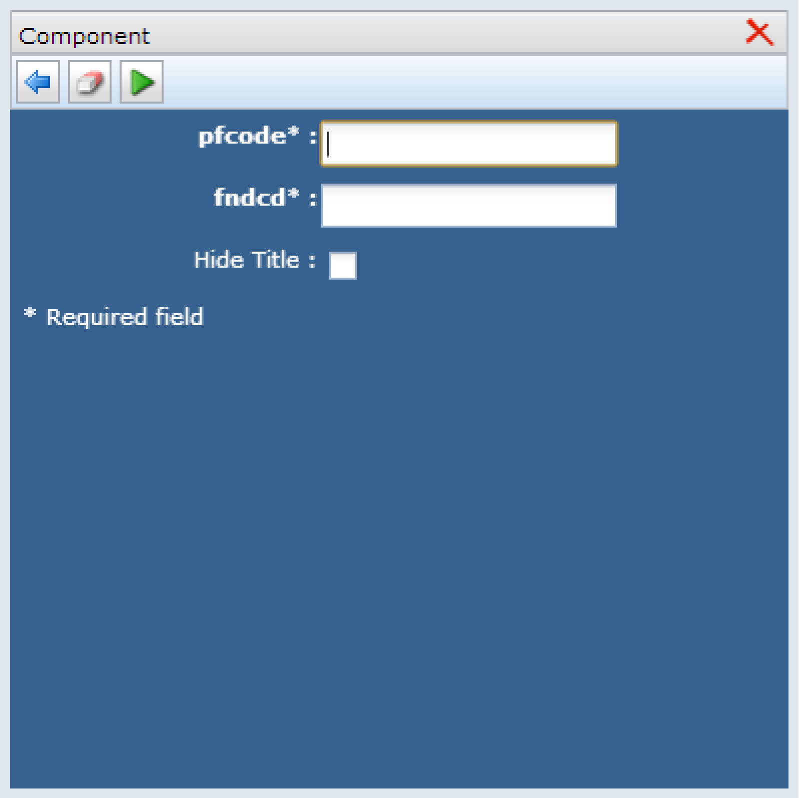
Delete

Description	Click this button to delete the User defined component from the dashboard.

Component configuration window

The **Component configuration** window opens when you click the **Configure component** button. This window is common to all imported components. The **Component configuration** window allows you to set the parameter values that will be used to run the component.

Home > Links > Dashboard Manager > New > text > configuration



pfcode* :

fndcd* :

Hide Title : □

* Required field

Buttons

Back

Description	Returns you to the component view.

Clear

Description	Clears all the fields in the User Defined component configuration view.

Run

Description	Runs the User Defined component using the current parameters.

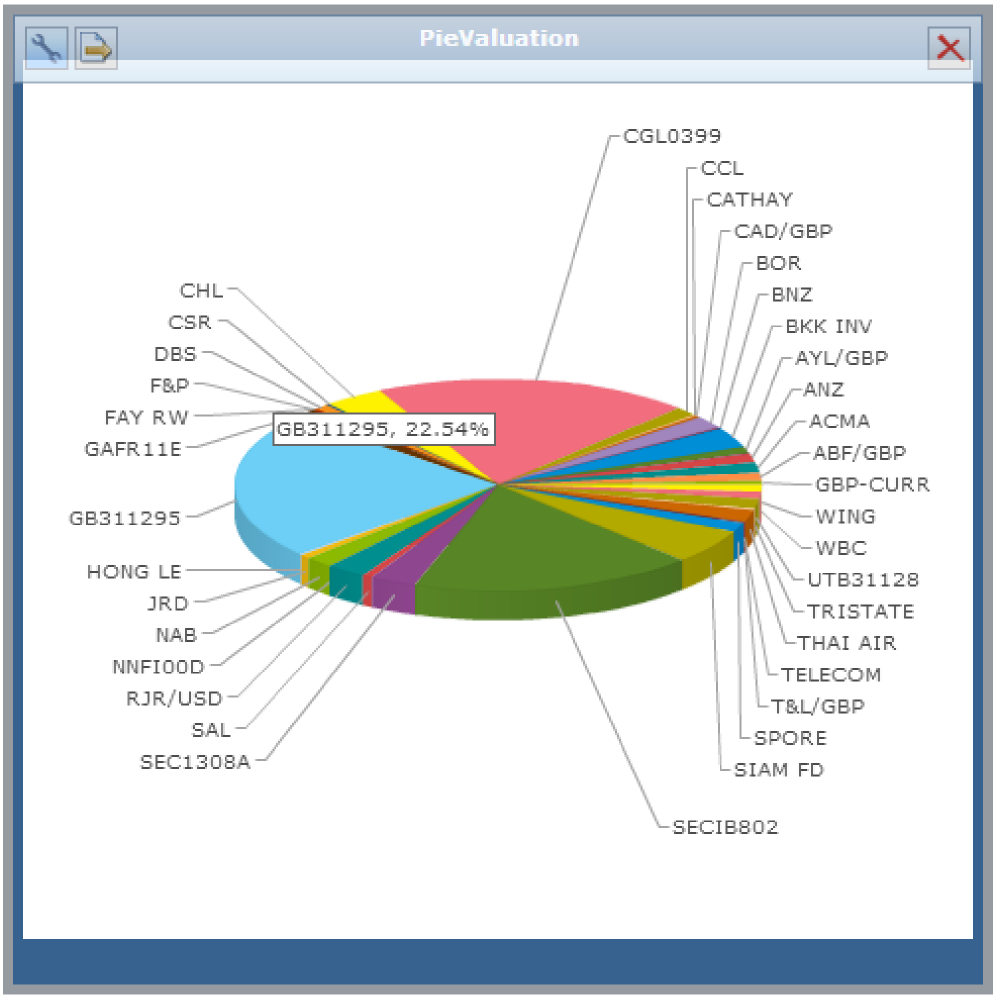
Delete

Description	Deletes the component from the dashboard.

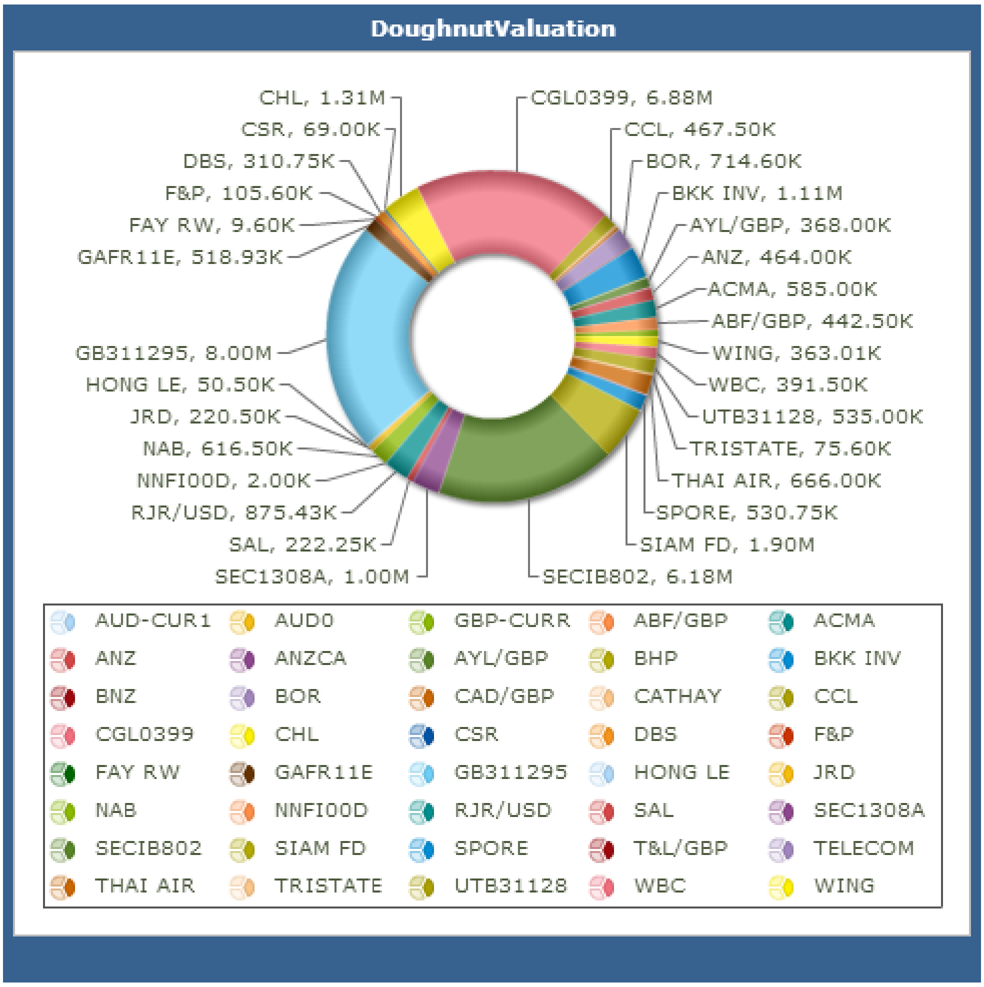
The different component views

The following charts can be used and defined by using the <Chart> element in the visualisation component xml file:

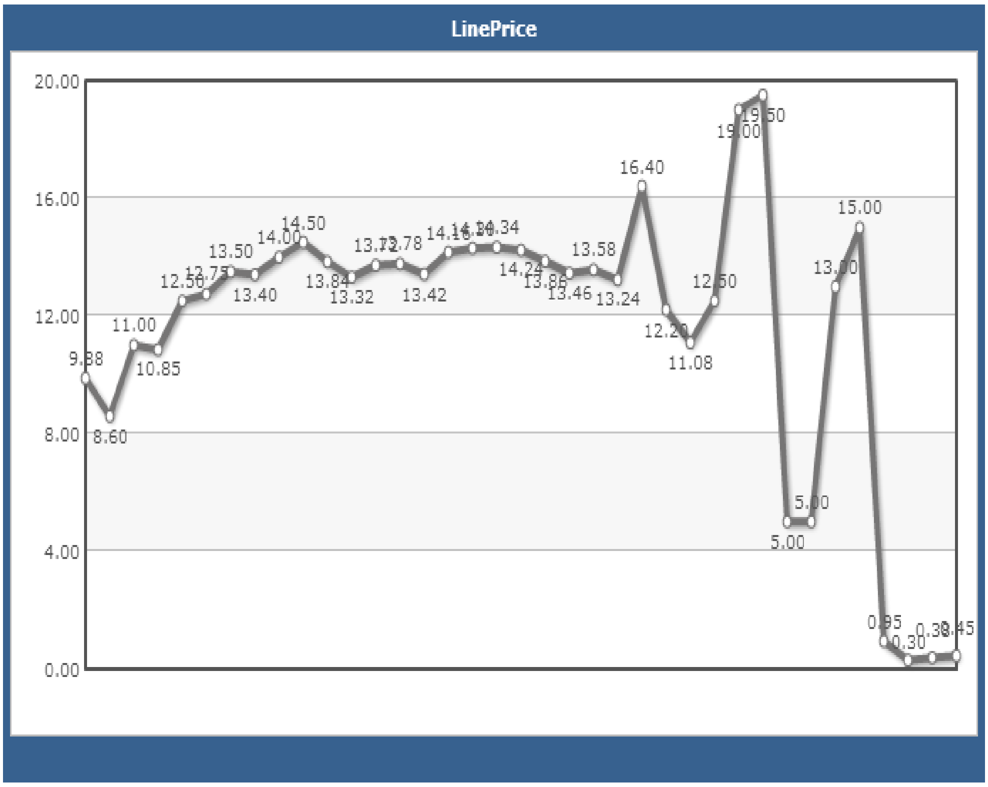
Pie charts <pieChart>



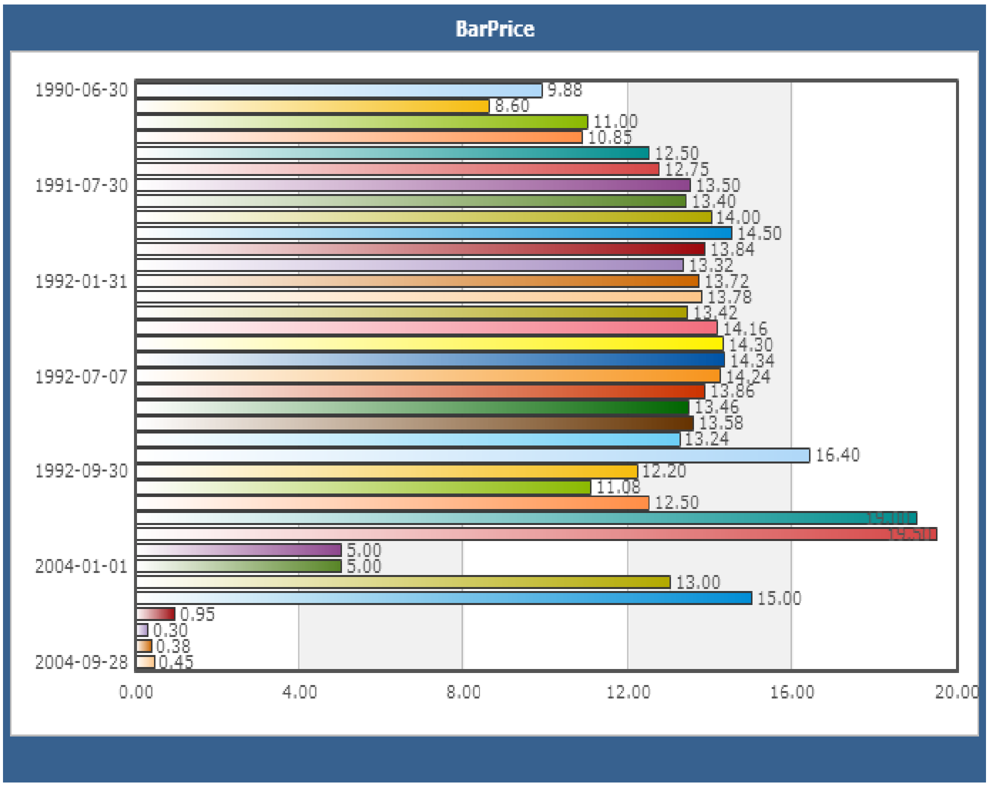
Doughnut charts <doughnutChart>



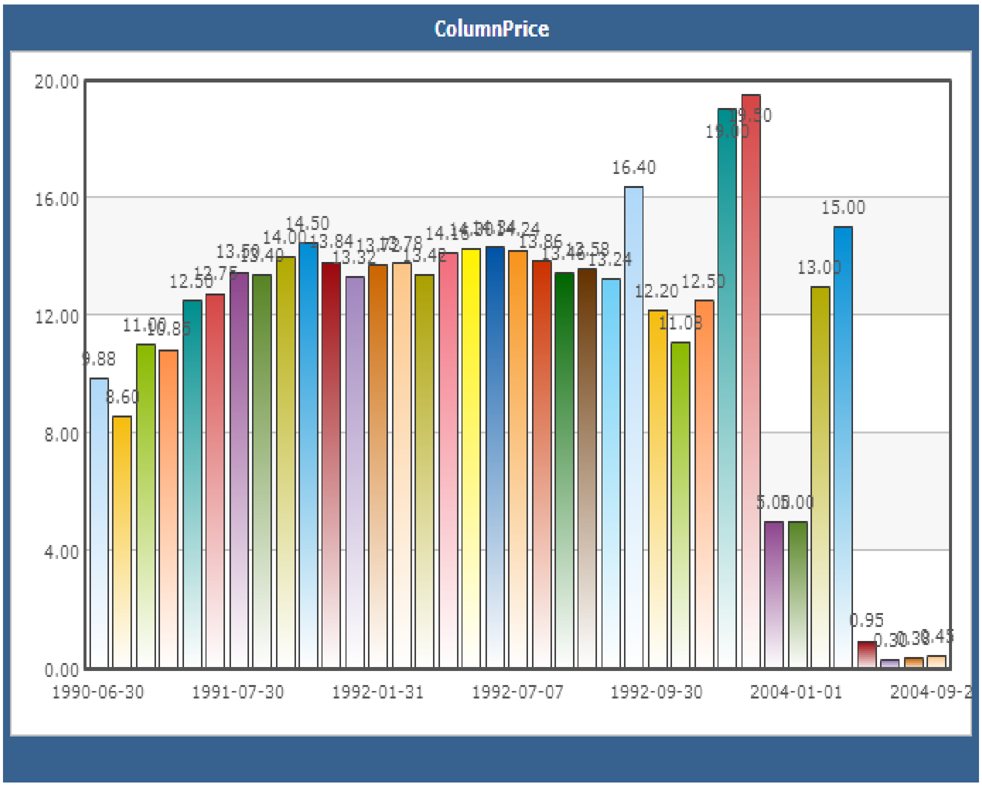
Line charts <lineChart>



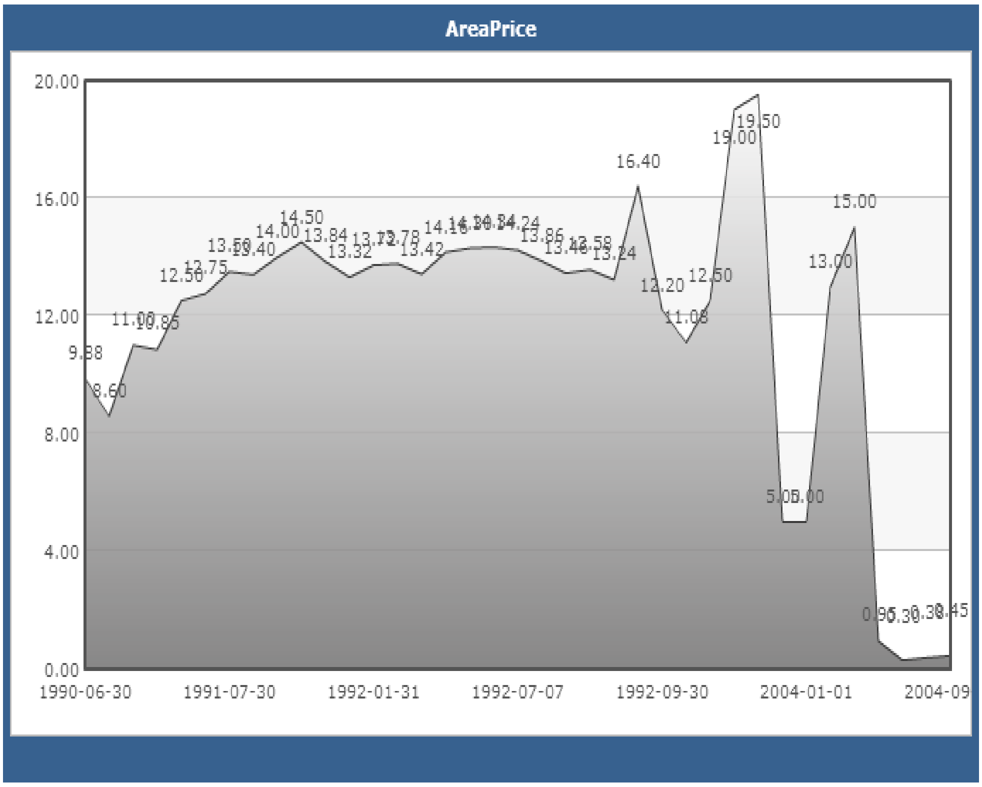
Bar charts <barChart>



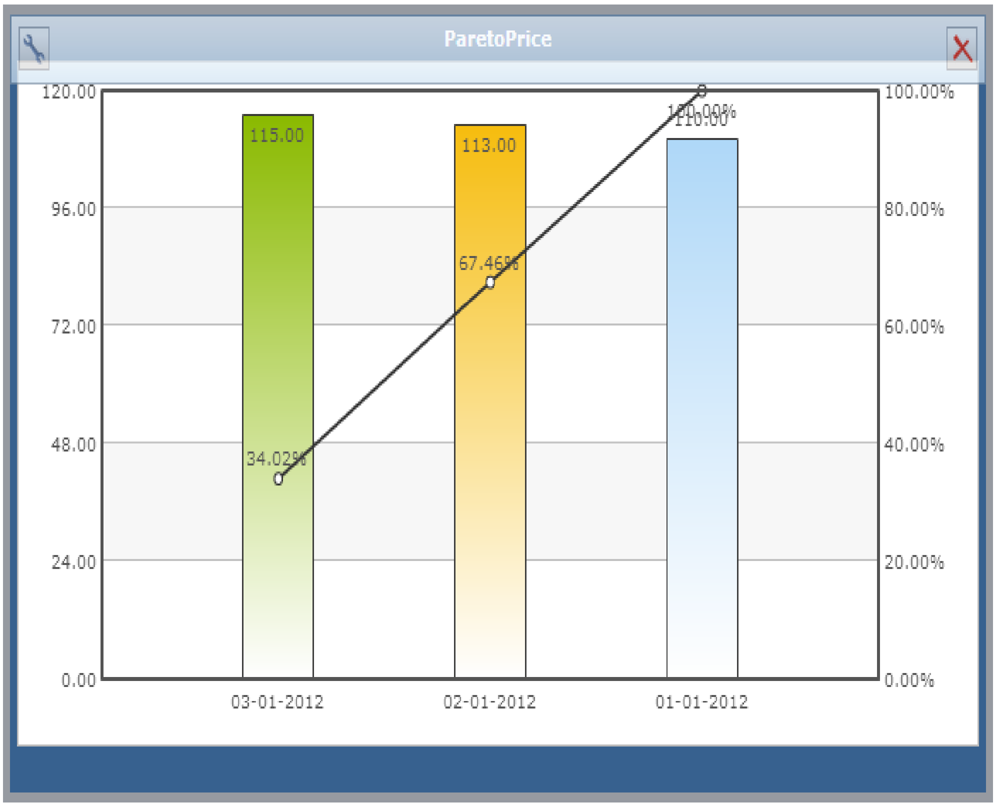
Column charts <columnChart>



Area charts <areaChart>

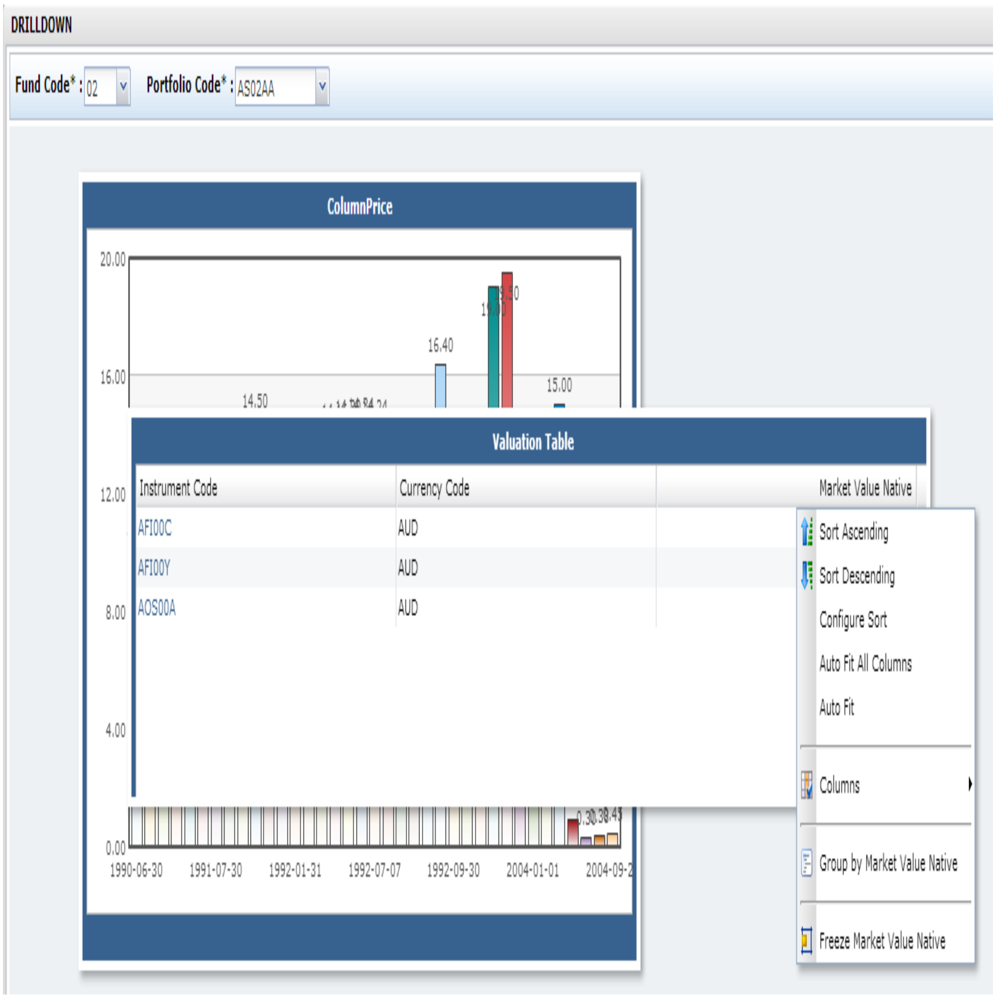


Pareto charts <paretoChart>



Customizing the report view

Throughout the application you can customise your view of Data Grid results. The options available to you are accessed from the heading bar at the top of the grid.



DRILLDOWN

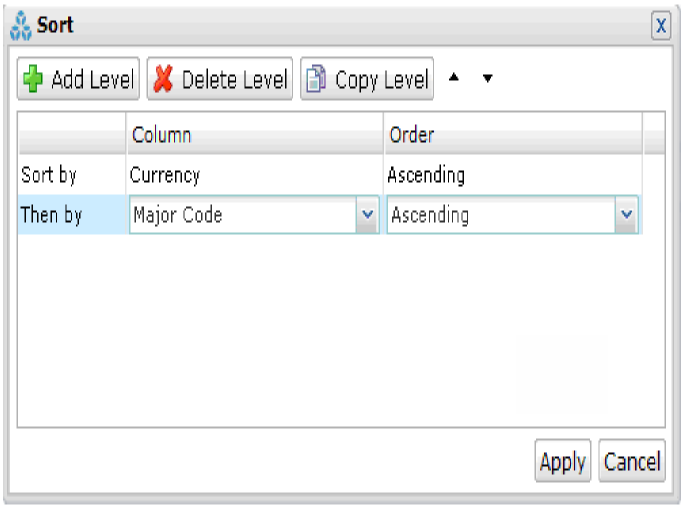
Fund Code* : 02   Portfolio Code* : AS02AA

Valuation Table

Instrument Code	Currency Code	Market Value Native
AFI00C	AUD	
AFI00Y	AUD	
AOS00A	AUD	

When you right-click on the heading bar of an individual column, some or all of the following options are available:

Menu Option	Description
Sort Ascending	Sorts the column in ascending order.
Sort Descending	Sorts the column in descending order.
Configure Sort	Sorts by multiple columns. The result of the sort is indicated by a number in the column title. The following configuration window opens:

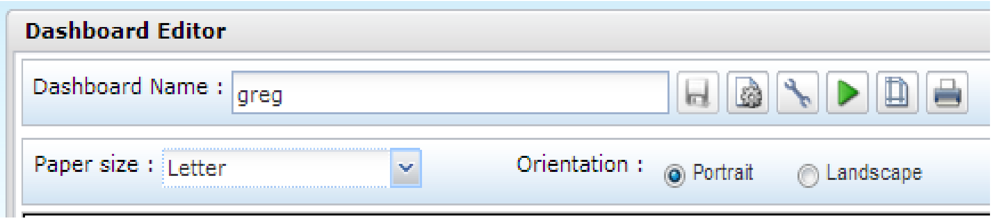
Menu Option	Description
	 Sort Add Level Delete Level Copy Level ▲ ▼ Apply Cancel Column | Order Sort by | Currency | Ascending Then by | Major Code | Ascending
Auto Fit All Columns	Changes the width of all the columns to fit the size of the data.
Auto Fit	Changes the width of individual columns to fit the size of the data.
Columns	Used to select the columns that are to be displayed in the data grid.
Group by <Column Name>	Groups data according to the selected column. When this function is applied, a new row for each unique value is displayed in bold with the corresponding rows grouped.

Note

Standard list grids are the only supported component. Tree view list grids are not supported.

Printing page setup

The **Print page setup** menu bar opens when you click the **Show/hide page setup** button in the Dashboard Editor. In this window, you can view a dashboard print preview.



Dashboard Editor

Dashboard Name : greg

Paper size : Letter

Orientation : ● Portrait ○ Landscape

Fields

Paper Size

Description	In this field you can select an available paper size view.	
Type	Drop-down list	
Valid Settings	**Setting**	**Effect**
	Letter	Letter page size is displayed.
	A3	A3 page size is displayed.
	A4	A4 page size is displayed. This is the default size.

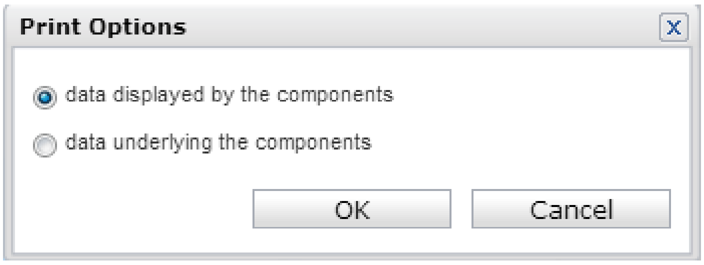
Orientation

Description	Using these option buttons you can select between portrait and landscape orientation.	
Type	Alphanumeric	
Valid settings	**Setting**	**Effect**
	Portrait	Page is displayed with a portrait page orientation.
	Landscape	Page is displayed with a landscape page orientation.

Print Options window

To print components click the **Print** button on the Dashboard Editor. The **Print Options** window opens.

Home > Links > Business Intelligence > Dashboard Manager >Dashboard Editor > Print



Print Options

● data displayed by the components

○ data underlying the components

Buttons

data displayed by the components

Description

Selecting the *data displayed by the components* option will print reports in the configured report view, containing the data retrieved from HiPortfolio/3. This is a graphic representation of the data.

Position the components on the dashboard to fit within the boundaries of the printed page. To aid this, the page setup view has an underlying grid line structure that is activated when you drag a component.

data underlying the components

Description

Selecting the *data underlying the components* option will print the information supplied in the report in a table format, showing the detail of the data retrieved from HiPortfolio/3.

This options enables you to print data retrieved by components regardless of how it is displayed.

Visualisation Components Manager

The **Visualisation Components Manager** screen is used to import and export visualisation components. Once the visualisation components have been set up, they can be placed on dashboards using the Dashboard Manager.

For help on how to create a visualisation component, see How to create a visualisation component.

Home > Links > Business Intelligence > Visualisation Components Manager

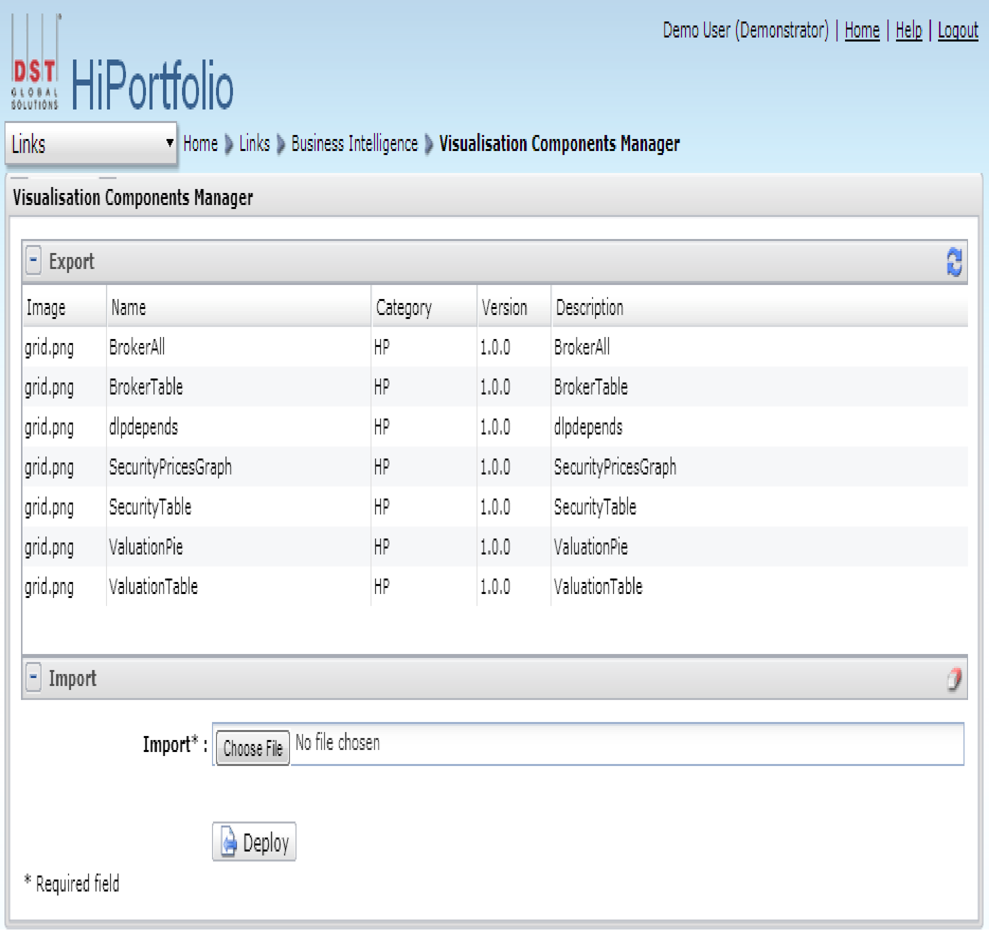

Demo User (Demonstrator) | Home | Help | Logout
DST GLOBAL SOLUTIONS
HiPortfolio
Links
Home ▸ Links ▸ Business Intelligence ▸ Visualisation Components Manager
Visualisation Components Manager
Export
Import
Choose File
Deploy

Image	Name	Category	Version	Description
grid.png	BrokerAll	HP	1.0.0	BrokerAll
grid.png	BrokerTable	HP	1.0.0	BrokerTable
grid.png	dlpdepends	HP	1.0.0	dlpdepends
grid.png	SecurityPricesGraph	HP	1.0.0	SecurityPricesGraph
grid.png	SecurityTable	HP	1.0.0	SecurityTable
grid.png	ValuationPie	HP	1.0.0	ValuationPie
grid.png	ValuationTable	HP	1.0.0	ValuationTable

Import* : No file chosen

* Required field

Columns

Image

Description	The image displayed by the component in the *Components* column of the **Dashboard editor** window.
Location	*Export* collapsible panel, **Visualisation Components Manager** window
Type	Read-only, Alphanumeric
Other information	This field is defined in the visualisation component file, in the `iconImage` attribute of the `componenttag`.

Name

Description	The name of the component.
Location	*Export* collapsible panel, **Visualisation Components Manager** window

Type	Alphanumeric
Other information	This field is specified in the visualisation component file, in the `name` attribute of the `componenttag`.

Category

Description	The category of the component.
Location	*Export* collapsible panel, **Visualisation Components Manager** window
Type	Alphanumeric
Other information	This field is specified in the visualisation component file, in the `category` attribute of the `componenttag`.

Version

Description	The version number of the component.
Location	*Export* collapsible panel, **Visualisation Components Manager** window
Type	Alphanumeric
Other information	This field is specified in the visualisation component file, in the `componentVersion` attribute of the `componenttag`.

Description

Description	The description of the component.
Location	*Export* collapsible panel, **Visualisation Components Manager** window
Type	Alphanumeric
Other information	This field is specified in the visualisation component file, in the `description` attribute of the `componenttag`.

Fields

Import

Description	The file path to the zipped visualisation component that is to be imported.	
Location	*Import* collapsible panel, **Visualisation Components Manager** window	
Mandatory	Yes	
Type	Alphanumeric	
Messages	**Message**	**Cause**
	Mandatory field is blank	This field cannot be blank
	File extension must be .zip format	When any other file type is selected from the file explorer window
	Zip file content not valid for this application	When the file is a zip file and does not contain the defintions.xml, required for visualisation components

Buttons

Export

Description	Click this button to export a component either to edit or use with another application.
Location	*Export* collapsible panel, **Visualisation Components Manager** window
Other information	This button is only available when you hover over a record in the list.

Browse

Description	Click this button to open a **File Explorer** window allowing you to locate and select the file you want to import.
Location	*Import* collapsible panel, **Visualisation Components Manager** window

Import

Description	Click this button to import the selected component in to the HiPortfolio/3 web UI and display the component in the list of components.
Location	*Import* collapsible panel, **Visualisation Components Manager** window

How to use the Dashboard Editor

For help on how to use the Dashboard Editor, see:

Creating a new dashboard

Editing an existing dashboard

Saving a dashboard

Viewing an existing dashboard

Exporting to CSV

Creating a new dashboard

To create a new dashboard, follow these steps:

1. Access the **HiPortfolio Web UI** by logging in using the HiPortfolio Web UI login screen in your web browser.
2. Navigate to the Dashboard Manager by selecting the Dashboard Manager option from the Links menu.
3. Click **New** to open a new dashboard in the Dashboard Editor.
4. Type in a name for the new dashboard in the *Dashboard Name* field.
5. Drag the required components onto the new dashboard from the Components Toolkit.
6. Click **Save** to save the new dashboard.

Editing an existing dashboard

To edit an existing dashboard, follow these steps:

1. Access the **HiPortfolio Web UI** by logging in using the HiPortfolio Web UI login screen in your web browser.
2. Navigate to the Dashboard Manager by selecting the Dashboard Manager option from the Links menu.
3. Select the relevant dashboard from the list of existing dashboards.
4. Click **Edit** to open the selected dashboard in edit mode.
5. Edit the dashboard and click **Save**.

Save a dashboard

The **Save** button becomes available once you have:

- added content to a new dashboard
- specified a dashboard name
- modified an existing dashboard

Note

Saving a dashboard saves the component details such as the component type, size, location and input parameters. It does not save the displayed data retrieved from the database. The next time a dashboard is opened, the latest data is retrieved from the database. Any formatting that you have applied to data grids or charts will also not be saved.

The dashboard changes made are not automatically saved. It is important to remember this if you close the dashboard or navigate away from the HiPortfolio Web UI browser.

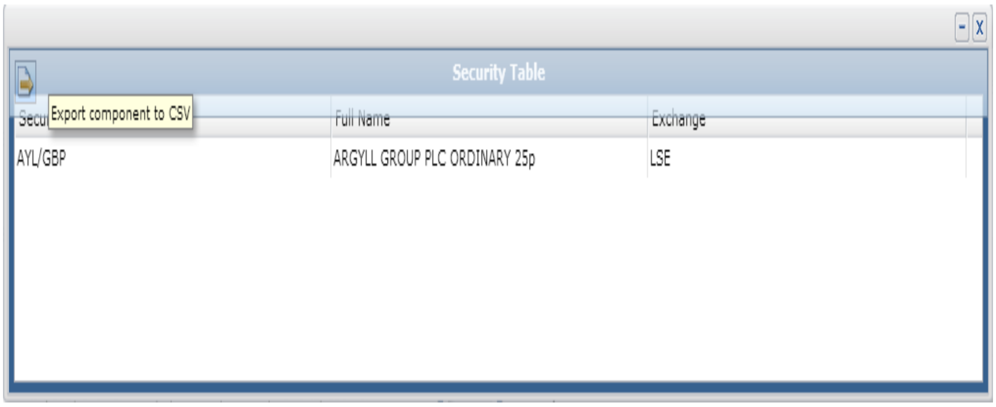
How to view an existing dashboard

To view an existing dashboard, follow these steps:

1. Open the Dashboard Manager.
2. Select the relevant dashboard from the list.
3. Click **View**.
4. The dashboard will open the component in View mode; which is read-only.

Exporting to CSV

Once you have configured your components and applied the criteria to retrieve data, the data can be exported to a CSV file.



Secu	Full Name	Exchange
AYL/GBP	ARGYLL GROUP PLC ORDINARY 25p	LSE

A toolbar is displayed when you place your mouse over the component. Clicking the **Export a Component to CSV** icon opens the standard file download window. When clicked, your browser will prompt you with a dialog to save or open the CSV file. This behaviour may vary depending on the browser you are using.

HiPortfolio Web UI Administration

For information on how to configure and manage the HiPortfolio Web UI see:

Login configuration

Business Intelligence

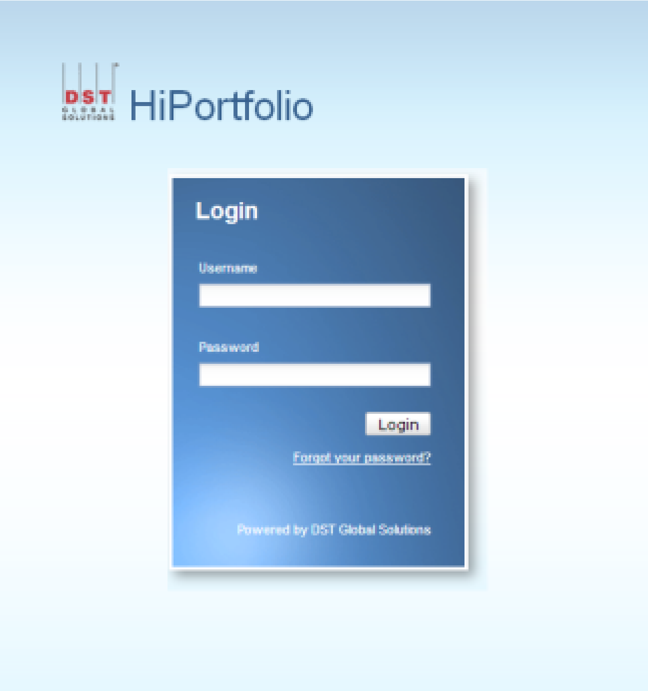
Web UI Permissions

Data Object Changes

Login configuration

The users with access to HiPortfolio/3 will share login capabilities with the HiPortfolio Web UI. This means that any HiPortfolio Web user will first need to exist or be created as a valid user in HiPortfolio/3 before being able to access the HiPortfolio Web UI.

User names are set up in HiPortfolio/3 Maintenance > User maintenance > Maintenance . HiPortfolio Web UI permissions relating to a user need to be set up in the policy.xml.



Note

If the number of maximum password attempts specified in the HiPortfolio/3 function SYUSECON is exceeded, access to both the HiPortfolio Web UI and HiPortfolio/3 is disabled and the following error message is displayed:

Number of login attempts exceeded. Your account is disabled, please contact your system administrator.

Page Disabled error message

The error message '*Page disabled. Ask your administrator for access rights*' is displayed when the following occurs:

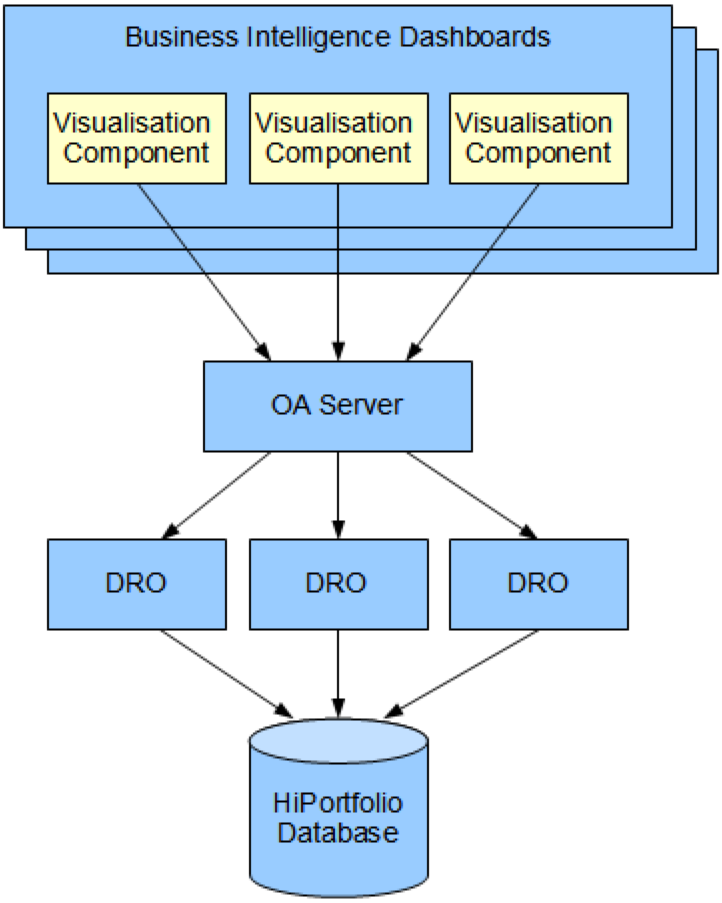
- the Admin Dashboard (AD) module is inactive
- the Business Intelligence (BI) module is inactive
- an invalid default page is contained in the security.xml file for the user

Note

The default page for a user is determined by the *defaultCommand* setting in the security.xml file.

Business Intelligence Architecture

The Business Intelligence component is used to give a visual presentation of HiPortfolio/3 data. The following diagram illustrates the basic architecture of the Business Intelligence component:



See the following topics for additional information on how to configure and manage the Business Intelligence component:

Adding images to the Dashboard Editor

Component Error Messages

Drill-down Functionality

How to create a visualisation component

Component xsd

Definitions.xml File

XML Validation

Adding images to the HiPortfolio Web UI

The **Image Selection** window opens when you click the **Configure component** button in the Image Component window, allowing you to select an image to display on the Dashboard Editor.

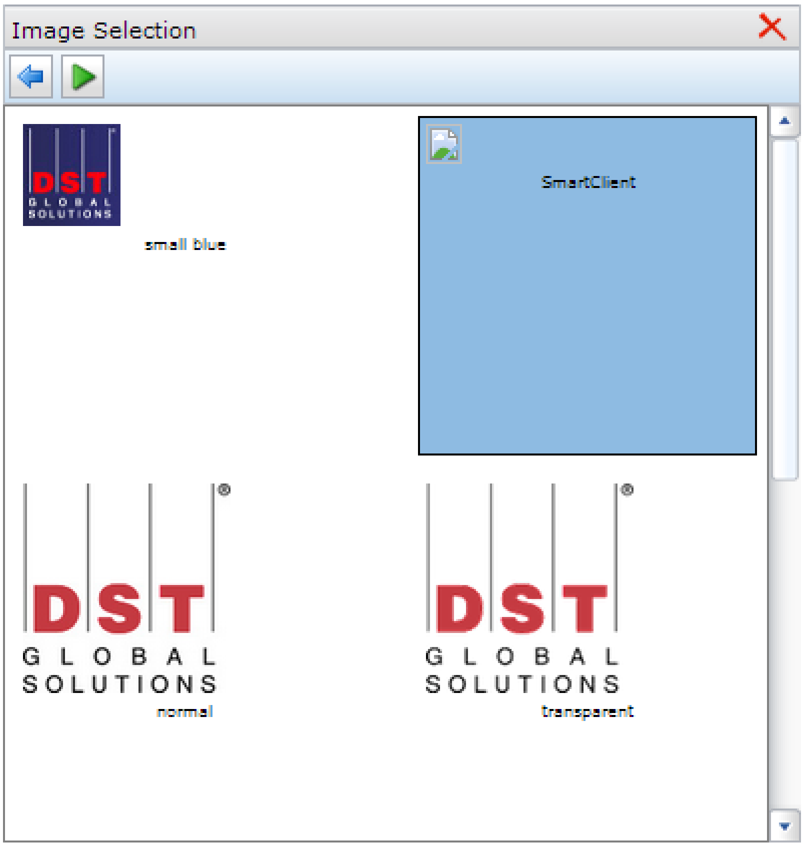
You can add additional images to display by adding them to the images folder on the application server. The following file formats are supported:

- png (recommended)
- jpg
- gif
- bmp (not recommended)

Note

For additional information on how to configure the image folder, see the *Installagion guide* (H3WebUIConfigDeployGuide.pdf).

Home > Links > Dashboard Manager > New > image > configuration



Buttons

Back

Description	Returns you to the component view

Run

Description	Runs the image component using the current parameters

Delete

Description	Deletes the component from the dashboard

Component Error Messages

The following lists provides a description of the possible error messages and how they may be resolved..

Select Fund Code and/or Portfolio Code to view this visualisation component

The component that displays on the dashboard requires that you select a fund code and portfolio code from the drop-down list before it can query the database and render its view. Select a fund code and portfolio code to display the data.

HiPortfolio Runtime Error: Check user name and or password

The user name or password provided when logging into the HiPortfolio/3 Web UI are no longer valid. Contact your system administrator to reset your username and/or password

HiPortfolio Runtime Error: You do not have permission to access Portfolio Code: ‘xxxxxx’

The user ID and password used to log into the HiPortfolio Web UI does not have permission to view the portfolio code specified. To gain access, log into HiPortfolio/3 and add in a permission entry under **Data Objects Data Permissions** (Reports > Data Objects > Data permission maintenance)

HiPortfolio Runtime Error: You do not have permission to access Fund Code: ‘xx’

The user ID and password used to log into the HiPortfolio Web UI does not have permission to view the fund code specified. To gain access, log into HiPortfolio/3 and add in a permission entry under **Data Objects Data Permissions** (Reports > Data Objects > Data permission maintenance)

Drill-down Functionality

After defining the basic components, it is possible to implement drill-down functionality by creating links from one component to another, forming a parent/child relationship.

Component definitions

A component will have a definition of how to present data within its <views> section, which will be either a table or a chart.

- A table, can be made a parent component by specifying a <componentLink> element within one of its <column> elements.
- A chart's <componentLink> can be defined in its <chartSeries>.

See XML Tag Definitions for more information on how these tags can be used.

Component links

Each component link must be defined by specifying:

- the name of the child component and
- any arguments that need to be passed when the link is executed.

Such arguments must be defined with the name of a parameter in the child component. Values can either be hard-coded or reference a field or parameter (using the ${…} notation) in the parent component.

- Links in **columns** display as blue text and are underlined when you move the mouse over the cell. Open the child component by clicking the cell. Each column in a table can have links to several components, which are available through a context menu. Where multiple links are specified, one must be marked as the default and it is this link that is opened when you click in the cell.
- Single series **charts** can have only one component link specified. Linked charts are identified when you move the mouse over the chart and the standard mouse icon is replaced with a hand icon. Once in a chart, standard chart functionality, eg, Rotating and Slicing, is available from a context menu.

How to create a visualisation component

The creation of a Visualisation Component definition requires that users be knowledgeable in:

- Extensible Markup Language (XML)
- XSD schema descriptions
- SQL SELECT statements to source the data for their components
- Data Retrieval Objects (DROs)

Components are defined in an XML file, named definition.xml. Each component's definition is in a separate file. A visualisation component can be defined in an XML file using any XML editor tool.

Follow these steps to create a valid XML component file:

1. Create an xml file in an xml editor of your choice.
2. Create a <component> tag to specify the following:

Category	The group definition of the component. This can be an existing or a new category and can be used to group components.
Name	The name of the component.
Schema Version	The version of the XML Schema Document (XSD) the component will reference. This is used to ensure the component is compatible with the installed version of the HiPortfolio Web UI.
Component Version	The version of the component created. This is used to keep a record of changes made to the component as it is developed.

3. Create the following 3 additional tags:

<Criteria>	Any parameters required to run the component data query.
<Data>	The SQL statement that will be run by the component to return data for display.
<Views>	How the returned data from SQL will be viewed.

4. Save the xml file as *definition.xml*.
5. Zip the xml file.
6. Upload the XML file using the Visualisation Components Manager.

Component xsd

The component.xsd file is used to configure the different components used in the HiPortfolio Web UI. For more information about the component.xsd file, see:

component.xsd file

Component.xsd Example

Component XSD File Example

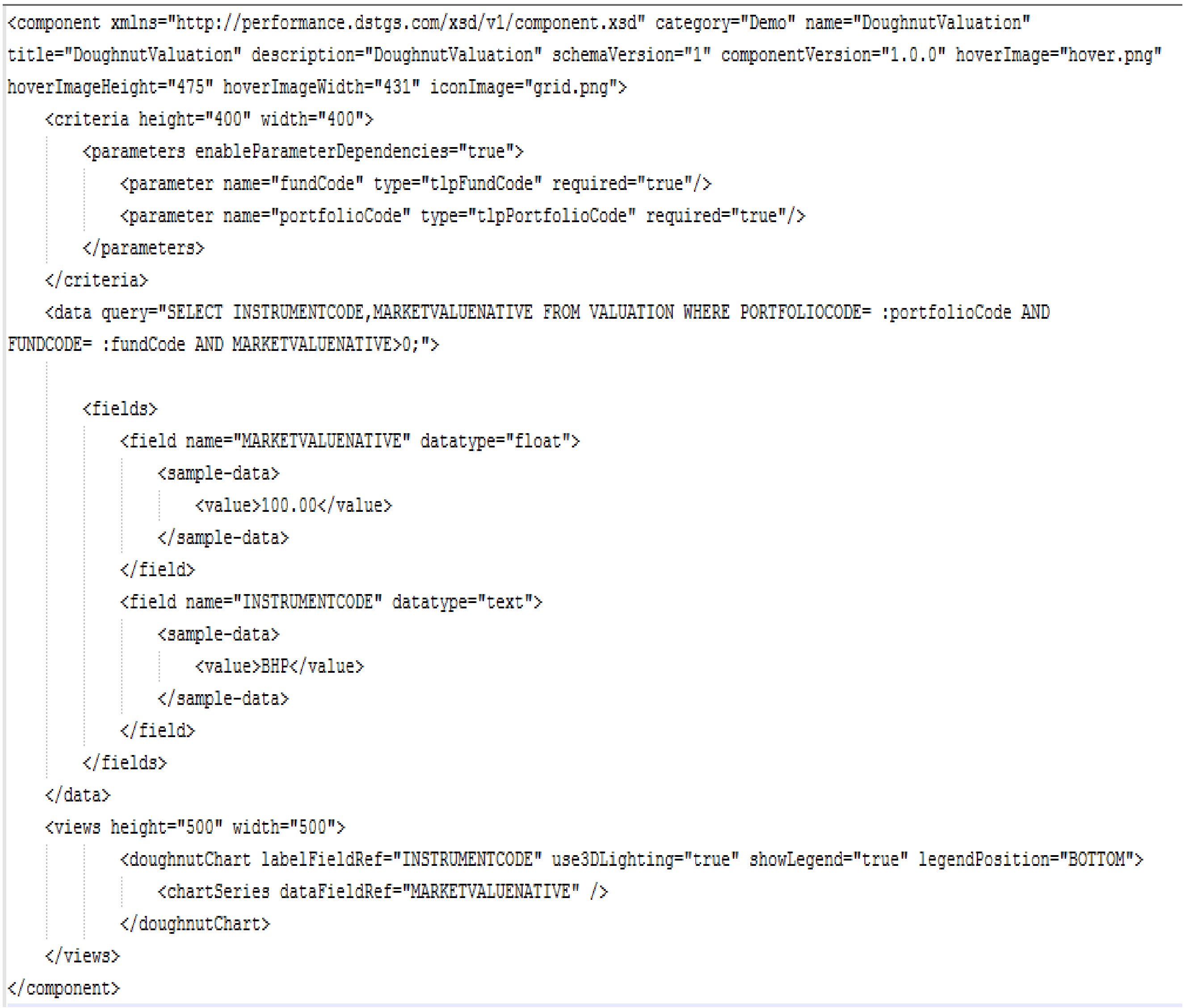
The following .xsd file can be used as an example to create your own component.xsd files:

xsd

XML File Example

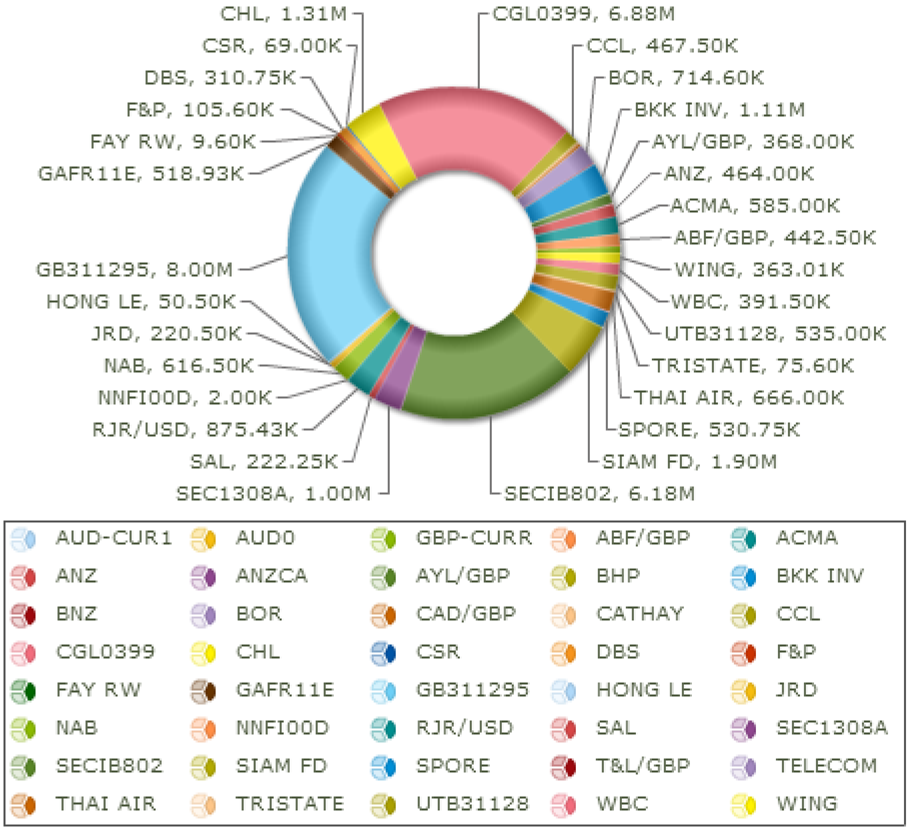
The following is an example of the xml used to produce a component:

Doughnut Chart



```
<component xmlns="http://performance.dstgs.com/xsd/v1/component.xsd" category="Demo" name="DoughnutValuation"
title="DoughnutValuation" description="DoughnutValuation" schemaVersion="1" componentVersion="1.0.0" hoverImage="hover.png"
hoverImageHeight="475" hoverImageWidth="431" iconImage="grid.png">
    <criteria height="400" width="400">
        <parameters enableParameterDependencies="true">
            <parameter name="fundCode" type="tlpFundCode" required="true"/>
            <parameter name="portfolioCode" type="tlpPortfolioCode" required="true"/>
        </parameters>
    </criteria>
    <data query="SELECT INSTRUMENTCODE,MARKETVALUENATIVE FROM VALUATION WHERE PORTFOLIOCODE= :portfolioCode AND
FUNDCODE= :fundCode AND MARKETVALUENATIVE>0;">

        <fields>
            <field name="MARKETVALUENATIVE" datatype="float">
                <sample-data>
                    <value>100.00</value>
                </sample-data>
            </field>
            <field name="INSTRUMENTCODE" datatype="text">
                <sample-data>
                    <value>BHP</value>
                </sample-data>
            </field>
        </fields>
    </data>
    <views height="500" width="500">
            <doughnutChart labelFieldRef="INSTRUMENTCODE" use3DLighting="true" showLegend="true" legendPosition="BOTTOM">
                <chartSeries dataFieldRef="MARKETVALUENATIVE" />
            </doughnutChart>
    </views>
</component>
```


CHL, 1.31M
CSR, 69.00K
DBS, 310.75K
F&P, 105.60K
FAY RW, 9.60K
GAFR11E, 518.93K
GB311295, 8.00M
HONG LE, 50.50K
JRD, 220.50K
NAB, 616.50K
NNFI00D, 2.00K
RJR/USD, 875.43K
SAL, 222.25K
SEC1308A, 1.00M
CGL0399, 6.88M
CCL, 467.50K
BOR, 714.60K
BKK INV, 1.11M
AYL/GBP, 368.00K
ANZ, 464.00K
ACMA, 585.00K
ABF/GBP, 442.50K
WING, 363.01K
WBC, 391.50K
UTB31128, 535.00K
TRISTATE, 75.60K
THAI AIR, 666.00K
SPORE, 530.75K
SIAM FD, 1.90M
SECIB802, 6.18M
AUD-CUR1
AUD0
GBP-CURR
ABF/GBP
ACMA
ANZ
ANZCA
AYL/GBP
BHP
BKK INV
BNZ
BOR
CAD/GBP
CATHAY
CCL
CGL0399
CHL
CSR
DBS
F&P
FAY RW
GAFR11E
GB311295
HONG LE
JRD
NAB
NNFI00D
RJR/USD
SAL
SEC1308A
SECIB802
SIAM FD
SPORE
T&L/GBP
TELECOM
THAI AIR
TRISTATE
UTB31128
WBC
WING

Definitions.xml File

The definitions.xml file is used to define the **Component Configuration** window parameter values that will be used to run the component. For more information about the Definitions.xml file, see:

XML Tag Definitions

Standard Parameters

Dashboard Level Parameters

XML Tag Definitions

The following is a description of the XML tags that are used to make up a visualisation component definition.xml file.

Component Attributes

Criteria Attributes

Parameters Attributes

Parameter Attributes

Data Attributes

Field Attributes

Value Attributes

Views Attributes

Table Attributes

Column Attributes

ComponentLink Attributes

Argument Attributes

PieChart Attributes

ColumnChart Attributes

DoughnutChart Attributes

LineChart Attributes

ParetoChart Attributes

BarChart Attributes

AreaChart Attributes

Other Attributes

Component Attributes

<component>- this element is used to define the visualisation component.

Attributes	Description	DataType	Required	Default
name	The name of the component.	text	Y	
schemaVersion	The version of the XML Schema Document (XSD). When first creating a definition.xml file, this attribute should match the version attribute of the schema itself. This is used to ensure the component is compatible with the installed version of the HiPortfolio/3 Web UI.	text	Y	
componentVersion	Supplied by the user to enable client versioning.	text	Y	
category	The category of the component, this is used for grouping purposes. This defines which folder the component will appear under in the component list. This can be a new category or one that already exists.	text	Y	
title	The title of the component as presented in the Component Toolkit on the Dashboard Manager page.	text	N	
description	A description of the component.	text	N	
hoverImage	The name of an image to display when hovering over the component in the Dashboard Manager component Toolkit. This image must be zipped up and imported with the definition.xml file.	text	N	
hoverImageHeight	The height of the hover image.	integer	N	
hoverImageWidth	The width of the hover image.	integer	N	
iconImage	The name of an image to display as an icon next to the name of the component on the Dashboard Manager component Toolkit.	text	N	
visible	Hide/show the component in the dashboard editor frame.	boolean	N	true

Criteria Attributes

<criteria>- this element is used to define the criteria required by the visualisation component.

Attributes	Description	DataType	Required	Default
height	The height in pixels of the component’s criteria view when first displayed.	integer	N	
width	The width in pixels of the component’s criteria view when first displayed.	integer	N	

Parameters Attributes

<Parameters>- container element for the individual criteria.

Attributes	Description	DataType	Required	Default
enableParameterDependencies	This is used in conjunction with dashboard level parameters. If set to true, this will allow the visualisation component to have the fund code and portfolio code parameters changed at the dashboard level when being viewed or edited. See Dashboard Level Parameters for more information.	boolean	N	false

Parameter Attributes

<parameter>- this element is used to define a criterion required by the visualisation component.

Attributes	Description	DataType	Required	Default
name	The name of the parameter (used within the SQL query to reference its value).	text	Y	
type	The type of the parameter. This can be defined as tlpFundCode for a fund code column and tlpPortfolioCode for a portfolio code column. All other columns should omit the 'type' attribute.	enum	N	text
title	The text used to label the parameter on the criteria view. When not supplied, this defaults to the name.	text	N	
required	This determines whether a value for the parameter must be supplied.	boolean	N	false

Data Attributes

<data>- this element is used to define the data that will be presented by the visualisation component.

Attributes	Description	DataType	Required	Default
query	This details the SQL query used to retrieve data for the component; parameter values are incorporated by providing the parameter name immediately preceded by a ":". Eg. SELECT BidPrice,Date FROM MarketDataSecurity WHERE Code= :SecurityCode	text	Y	

Field Attributes

<field>- this element is used to define a field, or column of data returned by the SQL query.

Attributes	Description	DataType	Required	Default
name	The name of the field as defined by the SQL query.	text	Y	
datatype	The data type of the field.	text boolean integer float date datetime	Y	

Value Attributes

<value>- Any value consistent with the data type of the associated field.

Attributes	Description	DataType	Required	Num
	Any value consistent with the data type of the associated field.	text	N	unlimited

Views Attributes

<views>- the element is a container element for the multiple elements that define the ways in which the data may be presented.

Attributes	Description	DataType	Required	Default
height	The height in pixels of the component's main view when first displayed.	integer	N	
width	The width in pixels of the component's main view when first displayed.	integer	N	

Table Attributes

<table>- this element is used to define a table or grid.

Attributes	Description	DataType	Required	Default
viewTitle	The main title for this view of the data. It may contain parameter values that are populated when the component is run, and may contain carriage returns. These must use the "${...}" notation. For example "Top 10 Flows within Composite ${securitycode}, ${:\n}since ${date}". The parameter name within the ${} must match the value of a name attribute of one of the <parameter> elements of the <criteria>.	text	N	
sortColumn	The name of the column to be used to sort the contents of the table when initially displayed.	text	N	
sortDirection	The direction for any initial sort. The valid values are: ▫ ascending ▫ descending	enum	N	
displayAsTree	This attribute determines whether the view is displayed as a tree grid.	boolean	N	false
wrapCells	This attribute determines whether any text that exceeds the column width should be wrapped on whitespace, with a corresponding increase in row height.	boolean	N	false

Column Attributes

<column>- this element is used to define each column in the table.

Attributes	Description	DataType	Required	Default
fieldRef	The name of <field> the data values of which will be presented in this column.	text	Y	
title	The text displayed in each column heading. It may contain parameter values that are populated when the component is run. These must use the "${...}" notation. If not specified the text is taken from the <field> name.	text	N	
decimalPlaces	Decimal number precision.	integer	N	2
visible	This determines whether the column is visible or not.	Boolean	N	true
groupOrder	This determines if the column should form part of a fixed grouping structure. This attribute should be set to 1 for the first column, 2 for the second, and so on. Setting this disables grouping from the context menu.	text	N	N

ComponentLink Attributes

<componentLink>- this element is used to define a link to another visualisation component.

See Drill-down Functionality for more details.

Attributes	Description	DataType	Require-d	Defaul-t
component	The name of the target component.	text	Y	

Argument Attributes

<argument>- this element is used to define a value to be passed to the target component when the link is activated.

Attributes	Description	DataType	Required	Default
parameter	The name of the parameter in the target component the value of which is being supplied.	text	N	
value	The value to be passed to the target component; this can contain either a simple value ("abc", "123", etc) or a parameter substitution reference using the "${}" notation; this attribute is required unless fieldRef is specified, in which case it is disallowed.	text	N	
fieldRef	The name of the <field> in the current row the value of which should be passed to the target component; this attribute is required unless value is specified, in which case it is disallowed.	text	N	

PieChart Attributes

<pieChart>- this element is used to define a Pie Chart.

Attributes	Description	Data-Type	Require-d	Defau-lt	Nu-m
labelFieldRef	The name of the field that contains the labels for the data values.	text	Y	-	
threeD	This determines whether the chart is displayed as 2D or 3D.	boolean	N	false	
viewTitle	The title for the view. The parameter name within the ${} must match the value of a name attribute of one of the <parameter> elements of the <criteria>.	text	N	-	
use3DLighting	This determines whether to use advanced gradients and shadow effects to create better looking 3D charts.	boolean	N	false	
showValues	This allows you to show data values on labels.	boolean	N	true	
showPercentValues	This determines whether to show percentage values inchart labels.	boolean	N	false	
decimals	The precision applied to numerical values that are displayed, ie, number of decimal places to display.	integer	N	2	10
forceDecimals	This attribute determines whether to force the display of decimal numbers by adding padding if necessary, eg, 12.34 ->12.3400 when 4 decimals is specified.	boolean	N	false	
numberSuffix	This is the number suffix.	text	N	-	
skipOverLapLabels	This determines whether to skip labels that are overlapping. If not, they might overlap if there are too many labels.	boolean	N	true	
showToolTip	This determines whether to show tool tips on chart.	boolean	N	false	
show-PercentInToolTip	This determines whether to show percentage values in the tooltip.	boolean	N	false	
showLegend	This determines whether the legend is shown on the chart.	boolean	N	false	
legendPosition	The legend can be plotted at two positions on the chart - below the chart (BOTTOM) and on the RIGHT side of the chart.	RIGHT / BOTTOM	N	RIGHT	

columnChart Attributes

<columnChart>- this element is used to define a Column Chart.

Attributes	Description	Data-Type	Require-d	Defau-lt	Nu-m
labelFieldRef	The name of the field that contains the labels for the data values.	text	Y	-	
threeD	This determines whether the chart is displayed as 2D or 3D.	boolean	N	false	
viewTitle	The title for the view. The parameter name within the ${} must match the value of a name attribute of one of the <parameter> elements of the <criteria>.	text	N	-	
use3DLighting	Whether to use advanced gradients and shadow effects to create better looking 3D charts.	boolean	N	false	
showValues	Show data values on labels.	boolean	N	true	
rotateValues	Rotate the values from horizontal to vertical.	boolean	N	false	
decimals	The precision applied to numerical values that are displayed, ie, number of decimal places to display.	integer	N	2	10
forceDecimals	This attribute determines whether to force the display of decimal numbers by adding padding if necessary, eg, 12.34 ->12.3400 when 4 decimals is specified.	boolean	N	false	
numberSuffix	This is the number suffix.	text	N	-	
skipOverLapLabels	This determines whether to skip labels that are overlapping. If not, they might overlap if there are too many labels.	boolean	N	true	
showToolTip	This determines whether to show tool tips on chart.	boolean	N	false	
labelDisplay	This attribute allows you to customize the alignment of data labels (x-axis labels). There are 5 options: AUTO, WRAP, STAGGER, ROTATE or NONE. **AUTO**: Default setting. Alignment of the data labels is determined automatically depending on the size of the chart. **WRAP**: wraps the label text if it is too long to fit in one line. **ROTATE**: rotates the labels vertically. **STAGGER**: divides the labels into multiple lines.	auto wrap stagger none	N	none	

DoughnutChart Attributes

<DoughChart>- this element is used to define a Doughnut Chart.

Attributes	Description	Data-Type	Require-d	Defau-lt	Nu-m
labelFieldRef	The name of the field that contains the labels for the data values.	text	Y	-	
threeD	This determines whether the chart is displayed as 2D or 3D.	boolean	N	false	
viewTitle	The main title for this view of the data. It may contain parameter values that are populated when the component is run. These must use the "${...}" notation. For example "Current Allocation by Asset Class for ${SecurityCode}". The parameter name within the ${} must match the value of a name attribute of one of the <parameter> elements of the <criteria>.	text	N	-	
use3DLighting	This determines whether to use advanced gradients and shadow effects to create better looking 3D charts.	boolean	N	false	
showValues	This allows you to show data values on labels.	boolean	N	true	
showPercentValues	This determines whether to show percentage values in labels of the chart.	boolean	N	false	
decimals	The precision applied to numerical values that are displayed, ie, number of decimal places to display.	integer	N	2	10
forceDecimals	This attribute determines whether to force the display of decimal numbers by adding padding if necessary, eg, 12.34 ->12.3400 when 4 decimals is specified.	boolean	N	false	
numberSuffix	This attribute allows a suffix to be applied to each value that is displayed. This provides the ability to describe the type of value, eg, 50%.	text	N	-	
skipOverLapLabels	This attribute determines whether to avoid displaying labels with smaller values if their text overlaps.	boolean	N	true	
showToolTip	This attribute determines whether the label and percent is displayed on the chart when the mouse is moved over each data point.	boolean	N	false	
showLegend	This determines whether the legend is shown on the chart.	boolean	N	false	
legendPosition	The legend can be plotted at two positions on the chart - below the chart (BOTTOM) and on the RIGHT side of the chart.	RIGHT / BOTTOM	N	RIGHT	

LineChart Attributes

<lineChart>- this element is used to define a Line Chart.

Attributes	Description	Data-Type	Required	Default	Num
labelFieldRef	The name of the field that contains the labels for the data values.	text	Y	-	
viewTitle	The title for the view. The parameter name within the ${} must match the value of a name attribute of one of the <parameter> elements of the <criteria>.	text	N	-	
showValues	This allows you to show data values on labels.	boolean	N	true	
rotateValues	Rotate the values from horizontal to vertical.	boolean	N	false	
decimals	The precision applied to numerical values that are displayed, ie, number of decimal places to display.	integer	N	2	10
forceDecimals	This attribute determines whether to force the display of decimal numbers by adding padding if necessary, eg, 12.34 ->12.3400 when 4 decimals is specified.	boolean	N	false	
numberSuffix	This attribute allows a suffix to be applied to each value that is displayed. This provides the ability to describe the type of value, eg, 50%.	text	N	-	
labelDisplay	This attribute allows you to customize the alignment of data labels (x-axis labels). There are 5 options: AUTO, WRAP, STAGGER, ROTATE or NONE. **AUTO**: Default setting. Alignment of the data labels is determined automatically depending on the size of the chart. **WRAP**: wraps the label text if it is too long to fit in one line. **ROTATE**: rotates the labels vertically. **STAGGER**: divides the labels into multiple lines.	auto wrap stagger rotate none	N	none	
showToolTip	Determines whether to show a tool tip on the chart.	Boolean	N	false	

ParetoChart Attributes

<paretoChart>- this element is used to define a Pareto Chart.

Attributes	Description	Data-Type	Require-d	Defau-lt	Nu-m
labelFieldRef	The name of the field that contains the labels for the data values.	text	Y	-	
threeD	This determines whether the chart is displayed as 2D or 3D.	boolean	N	false	
viewTitle	The title for the view. The parameter name within the ${} must match the value of a name attribute of one of the <parameter> elements of the <criteria>.	text	N	-	
use3DLighting	This determines whether to use advanced gradients and shadow effects to create better looking 3D charts.	boolean	N	false	
showValues	This allows you to show data values on labels.	boolean	N	true	
rotateValues	Rotate the values from horizontal to vertical.	boolean	N	false	
decimals	Decimal number precision.	integer	N	2	10
skipOverLapLabels	This attribute determines whether to avoid displaying labels with smaller values if their text overlap.	boolean	N	true	
forceDecimals	This attributes forces decimal numbers. This adds decimal padding to numbers.	boolean	N	false	
numberSuffix	This is the number suffix.	text	N	-	
labelDisplay	This attributes allows you can customize the alignment of data labels (x-axis labels). There are 5 options: AUTO, WRAP, STAGGER, ROTATE or NONE. By default, this attribute is set to AUTO mode which means that the alignment of the data labels is determined automatically depending on the size of the chart. WRAP wraps the label text if it is too long to fit in one line. ROTATE rotates the labels vertically. STAGGER divides the labels into multiple lines.	auto wrap stagger rotate none	N	none	
showToolTip	Whether to show tool tip on chart.	Boolean	N	false	

BarChart Attributes

<barChart>- this element is used to define a Bar Chart.

Attributes	Description	Data-Type	Require-d	Defau-lt	Nu-m
labelFieldRef	The name of the field that contains the labels for the data values.	text	Y	-	
viewTitle	The title for the view. The parameter name within the ${} must match the value of a name attribute of one of the <parameter> elements of the <criteria>.	text	N	-	
showValues	This allows you to show data values on labels.	boolean	N	true	
decimals	The precision applied to numerical values that are displayed, ie, number of decimal places to display.	integer	N	2	10
forceDecimals	This attribute allows a suffix to be applied to each value that is displayed. This provides the ability to describe the type of value, eg, 50%.	boolean	N	false	
numberSuffix	This is the number suffix.	text	N	-	
showToolTip	Determines whether to show a tool tip on the chart.	Boolean	N	false	

AreaChart Attributes

<AreaChart>- this element is used to define an Area Chart.

Attributes	Description	Data-Type	Require-d	Defau-lt	Nu-m
labelFieldRef	The name of the field that contains the labels for the data values.	text	Y	-	
viewTitle	The title for the view. The parameter name within the ${} must match the value of a name attribute of one of the <parameter> elements of the <criteria>.	text	N	-	
rotateValues	Rotate the values from horizontal to vertical.	boolean	N	false	
numberSuffix	This is the number suffix.	text	N	-	
labelDisplay	This attributes allows you can customize the alignment of data labels (x-axis labels). There are 5 options: AUTO, WRAP, STAGGER, ROTATE or NONE. By default, this attribute is set to AUTO mode which means that the alignment of the data labels is determined automatically depending on the size of the chart. WRAP wraps the label text if it is too long to fit in one line. ROTATE rotates the labels vertically. STAGGER divides the labels into multiple lines.	auto wrap stagger rotate none	N	none	
showToolTip	Whether to show a tool tip on the chart.	Boolean	N	false	

Other Attributes

Attribute	Description
<componentLinks>	A container element for the individual component links.
<columns>	A container element for the individual columns in the table.
<fields>	A container element for the individual field definitions.
<sample-data>	A container element for sample values used to display within the visualisation component before it is run.

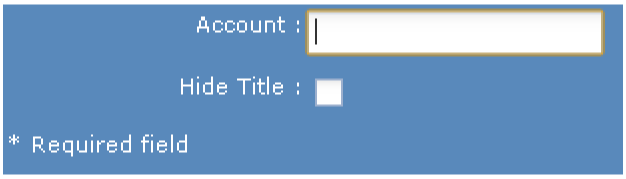
Standard Parameters

When creating a visualisation component, you can specify the parameters that drive your SQL query. The parameter definitions in the definition.xml file are shown as input boxes on your visualisation component.

You can define a <parameter> under the <parameters> element:

```
<parameters>
<parameter name="account title=”Account”/>
</parameters>
```

This is rendered on a component like this:



Account :

Hide Title : □

* Required field

A <parameter> element allows the following attributes:

Name	the parameter name corresponds to one of the SQL SELECT parameters. If the SQL statement has a parameter ‘:id’ then the Name is ‘id’.
Title	the title for this parameter that displays on the criteria page of the component.
Type	the type of the parameter. The following types can be defined: ▪ boolean ▪ date ▪ float ▪ integer ▪ text
Required	whether this parameter has to be entered before the component can be run. Valid values are true or false.

See XML Tag Definitions for more details on how these fit in definition.xml.

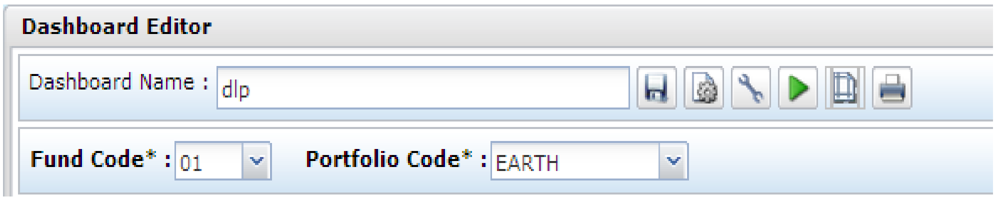
Dashboard Level Parameters

In addition to the standard parameters that can drive your SQL query, there are two advanced types of parameter that work with the Dashboard Level Parameters feature in the Dashboard Manager.

The two advanced parameter types allow your components to use the dashboard level parameters:

- **tlpFundCode** - allows the use of the Fund Code dashboard level parameter
- **tlpPortfolioCode** - allows the use of the Portfolio Code dashboard level parameter

The dashboard level parameters allow all dashboard parameters that are of the type tlpFundCode and tlpPortfolioCode to use the values that displayin the Fund Code and Portfolio Code drop-down boxes. These drop-down boxes can be changed at execution time by users who view the dashboard, allowing them to view visualisation components that represent data from different funds and portfolios without the need to create several separate dashboards.



Dashboard Editor

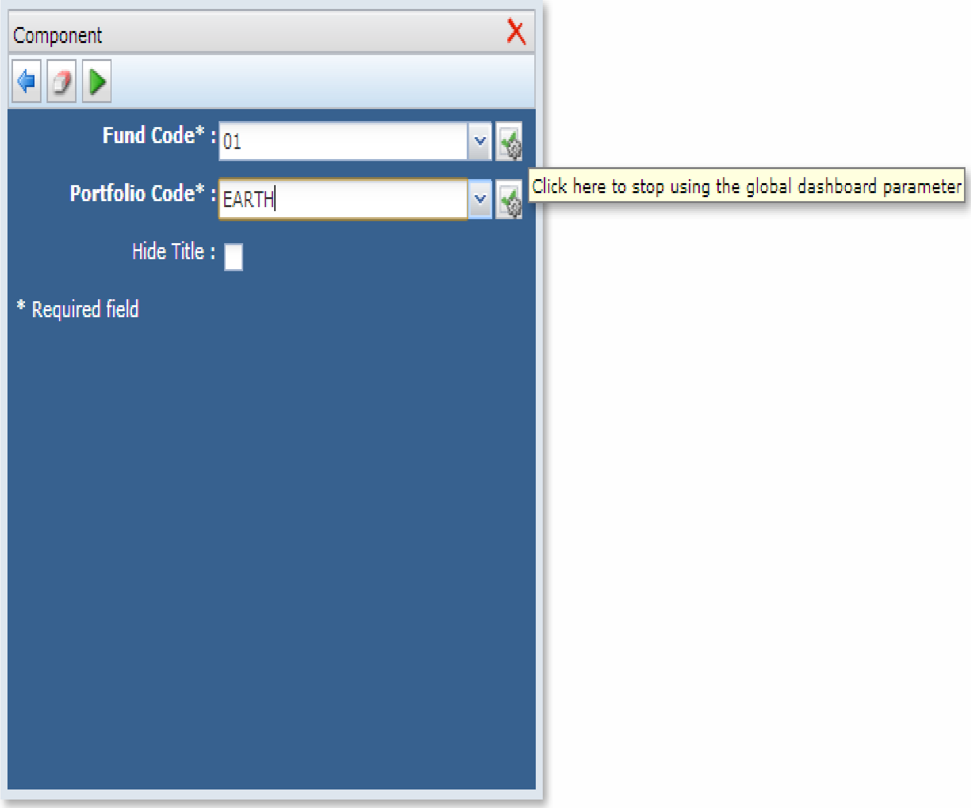
Dashboard Name : dlp

Fund Code* : 01

Portfolio Code* : EARTH

The fund code and portfolio code drop-down boxes are only populated with the fund codes and portfolio codes that the user has permission to access. These permissions are controlled by the **Data Objects Data Permissions** window in HiPortfolio/3. See HiPortfolio help for more information.

The dashboard level parameter functionality can be switched off at the visualisation component level by de-selecting the tick box next to the fund code or portfolio code drop-down boxes.



Component

Fund Code* : 01

Portfolio Code* : EARTH

Hide Title : □

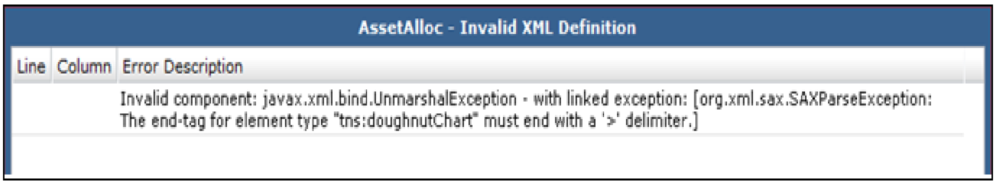
* Required field

Click here to stop using the global dashboard parameter

When the dashboard level parameter tick box is blank, these parameters will behave like standard parameters.

XML Validation

If a component containing invalid XML is made available, you will be presented with a message similar to the one shown in the example below when the component is dropped onto a dashboard. In which case, you should contact the system administrator or visualisation component developer.



AssetAlloc - Invalid XML Definition

Line	Column	Error Description
		Invalid component: javax.xml.bind.UnmarshalException - with linked exception: [org.xml.sax.SAXParseException: The end-tag for element type "tns:doughnutChart" must end with a '>' delimiter.]

Web UI Permissions

This enhancement adds the following types of authorisation to the HiPortfolio Web UI:

- HiPortfolio Web UI Function Permissions
- Policy.xml

HiPortfolio Web UI Function Permissions

This enhancement adds the Policy.xml to the HiPortfolio Web UI configuration. This XML file is used to specify a user's access and authority to functions within the HiPortfolio Web UI.

In the Policy.xml you can specify whether a user can:

- import/export components
- change HiPortfolio Web UI configuration settings
- create, amend or view dashboards
- specify what dashboards they view
- group users under specific Role IDs

The Policy.xml file contains a number of tags:

Tag	Description
<Roles>	Section tag used for defining the different roles.
<Role id>	Element tag used for defining a role name and title.
<permissions>	Group tag used for specifying permissions.
<permission name>	Element tag used for defining the dashboards for which the role ID has permissions. This tag is used within the <permissions> tag.
<resource type>	Section tag used for defining the object type the actions are applicable to eg, DASHBOARD.
<default permissions>	Group tag used for defining the permissions of the resource.
<objectpermission>	Element tag for defining the specific permission for the resource eg, DELETE.

Policy.xml

The HiPortfolio Web UI supports role based authorization and allows access to application modules (such as the Admin Dashboard, Configuration, Business Intelligence) to be controlled using the configuration of a security policy XML file. This policy file replaces the old security XML file that was implemented in previous versions for the Admin Dashboard.

Policy Configuration

The HiPortfolio Web UI policy configuration file contains three main elements:

- Resources
- Roles
- Users

Resources

`<resources></resources>`

The resources section allows administrators to define default permissions for instance level authorization to a number of system defined resources. Currently, the only resource implemented is the visualisation components within the Business Intelligence (BI) module.

Each resource has a separate <resource> section which defines a set of authorization attributes. These attributes are used in authorization expressions in order to grant access to specific resource instances.

For example:

```
<resources>
<resource type="DASHBOARD">
<authorizationAttribute name="id" type="integer"/>
<authorizationAttribute name="name" type="string"/>
<authorizationAttribute name="createdByUserName" type="string"/>
<authorizationAttribute name="createdDate" type="date"/>
<authorizationAttribute name="modifiedByUserName" type="string"/>
<authorizationAttribute name="modifiedDate" type="date"/>
<defaultPermissions canCreate="true">
<objectPermission condition="createdByUserName == #user.name" actions="*"/>
</defaultPermissions>
</resource>
</resources>
```

The above example defines some default permissions for the DASHBOARD resource, that correspond to Dashboards created in the Dashboard Manager. It defines that only users whose user name is the same as the user that created a dashboard, has access to that dashboard.

It also defines that the default permissions for the DASHBOARD resource are that anyone can create dashboards (provided they have first been granted access to the Dashboard Module page). This can be changed by setting canCreate to false.

Resource permissions are additive and therefore if a user is given object permissions to operate on dashboards with 'id lt 1000' and their role grants them access to operate on dashboards with 'id gt 2000' then that user will have access to all dashboards with IDs less than 1000 and greater than 2000.

It is not currently possible to create an objectPermission which denies access when it evaluates to true.

Therefore, in order to deny access to a specific user you must:

- set the default permissions to deny access to everyone
- grant access on a role or user basis to all roles/users except the one that you do not want to have access.

When a condition expression evaluates to true, the user is granted permission to perform the actions defined in the actions attribute. The available actions are:

- UPDATE
- DELETE
- EXECUTE

The special wildcard action (*) can be used to grant permissions on all actions.

Multiple actions can be defined within the same attribute by separating them with commas, for example:

```
<objectPermission condition="createdByUserName == #user.name" actions="READ,
UPDATE"/>
```

Roles

`<roles></roles>`

The roles section allows configuration of roles (user groups). These groups are not related to user groups found in HiPortfolio/3. Each <role> can have a set of <permissions> which grants that role access to a given page or module. A role can also define resourcePermissions to allow users who have been assigned this role to have this level of authorization applied.

Example:

```
<roles>
<role id="DASHBOARD_ROLE" title="Dashboard Developer" defaultCommand="dashboard_
manager">
<permissions>
<permission name="MODULE_DASHBOARD"/>
<resourcePermissiontype="DASHBOARD">
<objectPermissioncondition="id lt 1000" actions="*" />
</resourcePermission>
</permissions>
</role>
</roles>
```

The above example declares a DASHBOARD_ROLE role. This role only has permission to access the Dashboard Manager page and further will only be allowed to operate on dashboards which have an ID value of less than 1000.

Users

`<users></users>`

The users section allows configuration of authorization for a specific HiPortfolio/3 user that can be used to log into HiPortfolio/3. Permissions are attached to users in the same way that they can be attached to roles, allowing you to configure at a user level which users can access which pages or resources.

For example:

```
<users>
<user name='johns'>
<permissions>
<permission name="MODULE_DASHBOARD"/>
<resourcePermission type="DASHBOARD">
<objectPermission condition="name matches '^VALUATION'" actions="*" />
</resourcePermission>
</permissions>
</user>
</users>
```

The above example declares a user with a login of ‘johns' and grants him permission to the MODULE_DASHBOARD, providing him access to the Business Intelligence module.

Further resourcePermissions are defined that restrict the user from only operating on dashboards that begins with ‘VALUATION’.

The application is configured so that changes to the policy file can be made without having to restart the web application. For changes to be applied, users must log out of the HiPortfolio Web UI and log back in again and at this point the policy file will be reloaded.

Data Object Changes

This enhancement makes changes to the DRO:

- Adds a **Data Objects Data Permissions List** window to the *Data Objects* module
- Makes changes to the DRO Schema

For more details on these changes, see:

Data Objects Data Permission List Window

Data Permissions DRO

Additional Data Object Changes

Data Objects Data Permission List Window

This enhancement adds a new window **Data Objects Data Permissions List** to HiPortfolio/3. In this window, you can manage the data permissions for each user and populate data permissions in the HiPortfolio database.

The **List** tab displays a list of funds and portfolios that a user is authorised to view in the HiPortfolio Web UI. You can choose to add, change, view or delete funds or portfolios. This list can be filtered by *User* on the **Filter** tab.

For more details on the **Data Objects Data Permission List** window see

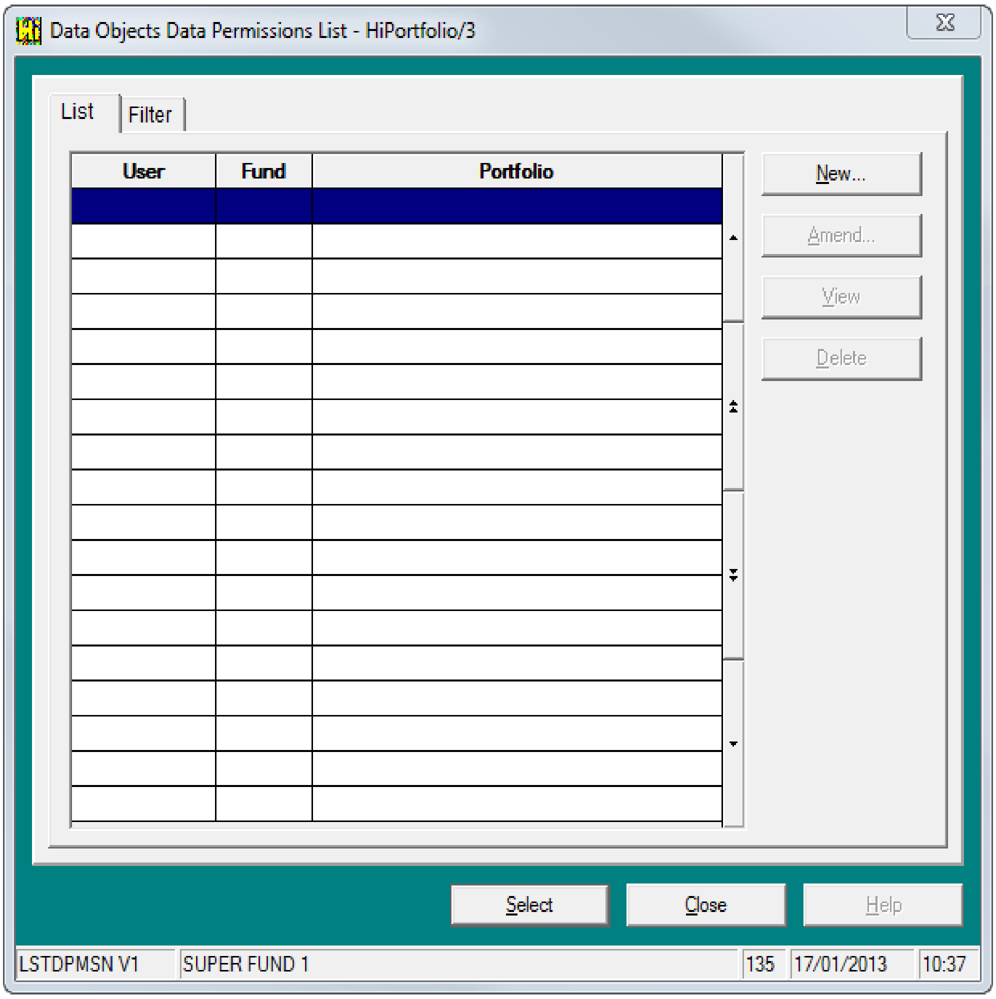
List tab

Filter tab

List tab - Data Objects Data Permission List window

The **List** tab displays a list of funds and portfolios that a user is authorised to view in the HiPortfolio Web UI.

Reports > Data Objects > Data Objects Data Permissions List



Columns

All columns are Alphanumeric.

User

Description	The user ID for whom permissions have been granted .

Fund

Description	The code of the fund the user has been given permission to view in the HiPortfolio Web UI.

Portfolio

Description	The code of the portfolio the user has permission to view in the HiPortfolio Web UI.

Buttons

New

Description	Click this button to open the **Data Object Data Permissions Maintenance** window in create mode.

Amend

Description	Click this button to open the **Data Object Data Permissions Maintenance window** in amend mode, displaying the highlighted record. Amendments can be made to the *Fund code* and *Portfolio code* fields only. The *User Code* field is displayed but is inactive.

View

Description	Click this button to open the **Data Object Data Permissions Maintenance** window in view mode, displaying the highlighted record.

Delete

Description	Click this button to delete the highlighted permissions record.

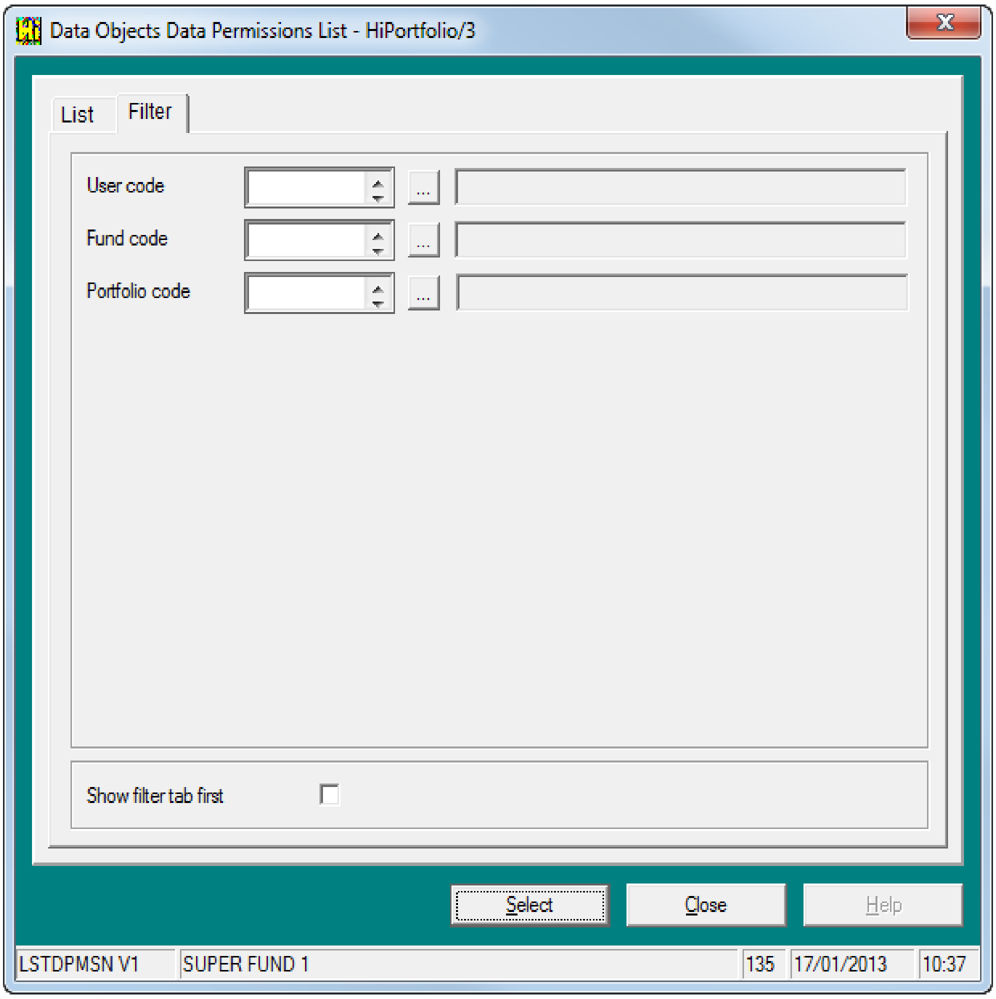
Select

Description	Click this button to open the **Data Object Data Permissions Maintenance** window in view mode, displaying the highlighted record.

Filter tab - Data Objects Data Permission List window

The **Filter** tab is used to filter the list of permissions by fund, portfolio or user that are displayed on the **List** tab.

Reports > Data Objects > Data Objects Data Permissions List > Filter tab



Fields

User Code

Description	The User code the list is to be filtered on.
Mandatory	No
Type	Alphanumeric 6, spinner
Messages	User code is not valid

Fund Code

Description	Fund code the list is to be filtered on.
Mandatory	No
Type	Alphanumeric 6, spinner
Messages	Fund code is not valid

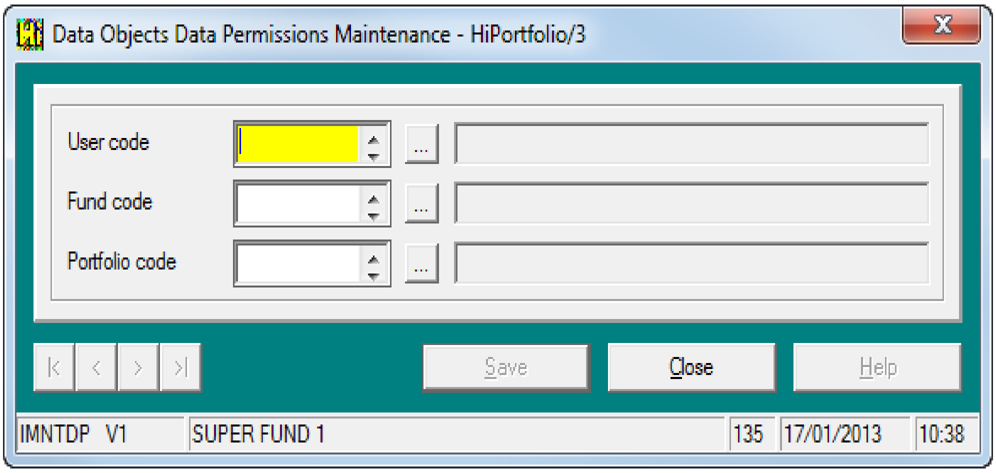
Portfolio Code

Description	Portfolio code the list is to be filtered on.
Mandatory	No
Type	Alphanumeric 6, spinner
Messages	Portfolio code is not valid

Data Objects Data Permissions Maintenance

In the **Data Objects Data Permissions Maintenance** window you can, specify a new fund or portfolio and change or view an existing fund or portfolio.

Reports > Data Objects > Data Objects Data Permissions List > New, Amend or View



Fields

User Code

Description	The code of the user being granted permission.
Mandatory	Yes
Type	Alphanumeric 2, spinner
Messages	User code is not valid
	Mandatory field is blank

Fund Code

Description	The code of the fund for which the user is being granted permission to view in the HiPortfolio Web UI.
Mandatory	Yes
Type	Alphanumeric 2, spinner
Messages	Fund code is not valid

Portfolio Code

Description	The code of the portfolio for which the user is being granted permission to view in the HiPortfolio Web UI.
Mandatory	Yes
Type	Alphanumeric 6, spinner
Messages	Portfolio code is not valid

Data Permissions DRO

This enhancement adds the DATAPERMISSIONS data object. This DRO produces a list of funds and portfolios for which a user is authorised to view data of through the HiPortfolio Web UI. This DRO is used by the HiPortfolio Web UI during the login process to create the user's session data authorisations. This function ensures that users are restricted to viewing data only for specific funds and portfolios. If you have permissions to the OJ module it will be possible to INSERT, UPDATE and DELETE data from this table via the ODBC interface.

Note

If changes are made to a user's permission during an active session, they must log out and then back in for the changes to take effect.

DRO schema:

Column	Description	Remark	Valid values	Data type
UserCode	User ID	Valid HiPortfolio user ID	User ID	Alphanumeric
FundCode	Fund Code	Fund Code the user is authorised to view data for	Valid Fund Code	Alphanumeric
PortfolioCode	Portfolio Code	Portfolio Code the user is authorised to view data for	Valid Portfolio Code	Alphanumeric

DRO Server Data Permissions Restrictions

The following rules apply to data permissions based on the entries in the DRO table:

- If there are no entries specific to a user, that user will not have permissions to extract data that is fund or portfolio specific.
- If there are entries in the DRO for a user and the FundCode is blank, the user will have access to all funds.
- If there are entries in the DRO for a user and the PortfolioCode is blank the user will have access to all portfolios for a fund.

Data Permissions History DRO

This enhancement adds the DATAPERMISSIONSHIST data object. This data object produces the history of the funds and portfolios a user can view through the HiPortfolio Web UI. This data object is available in the HiPortfolio Data Audit function.

Column	Description	Remark	Valid values	Data type
UserCode	User ID	Valid HiPortfolio user ID	User ID	Alphanumeric
FundCode	Fund Code	Fund Code the user is authorised to view data for	Valid Fund Code	Alphanumeric
PortfolioCode	Portfolio Code	Portfolio Code the user is authorised to view data for	Valid Portfolio Code	Alphanumeric
LastUpdatedAction	Last Action performed on the file	Value can be Create, delete, modify	Create, Delete, Modify	Alphanumeric
LastUpdatedDetailsUser	The user that last updated the record	HiPortfolio User ID of user who last update the file	User ID	Alphanumeric
LastUpdatedTimestamp	Timestamp of when the record was created or modified		Timestamp	Time

Permissions History DRO

This enhancement adds the PERMISSIONHIST data object. This data object produces the history of data object tables a user has permission to use. This data object is available in the HiPortfolio Data Audit function.

Column	Description	Remark	Valid values	Data type
AllowDelete	Whether the user has delete rights		1/0	Boolean
AllowInsert	Whether the user has insert rights		1/0	Boolean
AllowUpdate	Whether the user has update rights		1/0	Boolean
AllowView	Whether the user has view rights		1/0	Boolean
Table	Data Object the permissions pertains to		DRO Table name	Alphanumeric
UserCode	User code		User ID	Alphanumeric
LastUpdatedAction	Last Action performed on the file	Value can be Create, delete, modify	Create, Delete, Modify	Alphanumeric
LastUpdatedDetailsUser	The user that last updated the record	HiPortfolio User ID of user who last update the file	User ID	Alphanumeric
LastUpdatedTimestamp	Timestamp of when the record was created or modified		Timestamp	Time

Additional Data Object Changes

The following data objects are changed to include an additional column, *PortfolioCode*, to enable the DRO table permissions to be implemented. This additional column is populated with the same value as the existing *Portfolio* field in the DRO.

- ATTRIBUTIONVALUE
- ATTRIBUTIONVALUEHIST
- CASHCLIENTTRANSACTIONS
- CASHCLIENTTRANSACTIONSHIST
- GLANALYSISCODEDEF
- GLANALYSISCODEDEFHIST
- GLGROUPDEFINITION1
- GLGROUPDEFINITION1HIST
- GLGROUPDEFINITION2
- GLGROUPDEFINITION2HIST
- GLGROUPDEFINITION3
- GLGROUPDEFINITION3HIST
- GLGROUPDEFINITION4
- GLGROUPDEFINITION4HIST
- GLGROUPDEFINITION5
- GLGROUPDEFINITION5HIST
- GLGROUPDEFINITION6
- GLGROUPDEFINITION6HIST
- GLGROUPDEFINITION7
- GLGROUPDEFINITION7HIST
- GLGROUPDEFINITION8
- GLGROUPDEFINITION8HIST
- GLINCOMEACCDEF1
- GLINCOMEACCDEF1HIST
- GLINCOMEACCDEF2
- GLINCOMEACCDEF2HIST
- GLINCOMEACCDEF3
- GLINCOMEACCDEF3HIST
- GLINCOMEACCDEF4
- GLINCOMEACCDEF4HIST
- GLINCOMEACCDEF5
- GLINCOMEACCDEF5HIST
- GLINTEGPARAMETERS
- GLINTEGPARAMETERSHIST
- GLMONEYMARKETDEF1
- GLMONEYMARKETDEF1HIST
- GLMONEYMARKETDEF2
- GLMONEYMARKETDEF2HIST
- GLPFOLIOMARKETDEF1
- GLPFOLIOMARKETDEF1HIST
- GLPFOLIOMARKETDEF2

- GLPFOLIOMARKETDEF2HIST
- GLPFOLIOMARKETDEF3
- GLPFOLIOMARKETDEF3HIST
- GLPFOLIOMARKETDEF4
- GLPFOLIOMARKETDEF4HIST
- GLPFOLIONONSPECDEF
- GLPFOLIONONSPECDEFHIST
- GLTRANTYPEACCDEF
- GLTRANTYPEACCDEFHIST
- PERFORMANCE
- PERFORMANCEHIST
- REQCODEFORSNAPSHOTS
- SHORTTERMTRANSACTION
- SHORTTERMTRANSACTIONHIST
- UNITPRICETRANSACTIONS
- UNITPRICETRANSACTIONSHIST

Column	Description	Remark	Valid Values	Data Type
PortfolioCode	HiPortfolio Portfolio Code		Portfolio Code	Alphanumeric