TIBCO EBX® Graph View Add-on

Version 1.5.6
December 2023

Table of contents

Graph View Add-on Documentation

User Guide

Release Notes

Graph View Add-on Documentation

User Guide

CHAPTER 1

Introduction

This chapter contains the following topics:

1.1 Overview

The TIBCO EBX® Graph View Add-on gives you a global view of your data models and data hierarchy. Its visual representation provides an alternative to standard tabular views. The Graph View provides functionality that makes it simple to query tables, easily access their records and quickly understand the relationships between tables and records.

All of the options provided by the EBX® Graph View Add-on make it simple to effectively communicate the structure of data to users. Even something as simple as the 'Export image' service that captures the current Graph View screen can assist team collaboration and organizational communication efforts.

1.2 Basic concepts

You register tables with the add-on to enable Graph View display. Registration involves pointing the add-on to a table, or set of tables via a data space and data set.

After registration, you can tailor Graph View look and feel to fit your specific needs. For example, you can:

- Change standard node shapes, colors and text size.
- Hide, move and drag-and-drop to rearrange nodes.
- Save a reusable display configuration.
- Limit what data users can open in the Graph View.

Graph View display

The Graph View uses nodes to represent your data structure (tables) and data values (records). Arrows that connect nodes denote relationships and relationship direction. Each node label shows which table or record it represents and contains symbols used to convey valuable information. See 'Understanding node labels and symbols' for an explanation of all possible symbols.

Understanding relationships

Foreign key (FK) relationships are represented by arrows between nodes. These arrows also indicate relationship direction. For example, an 'Order portfolio' table may hold a FK to a 'Product' table. The arrow in this case connects the two nodes and points from the 'Order portfolio' table to the 'Product' table.

When you perform a query, an algorithm uses the relationship to return data values to the correct table. This document refers to this action as node resolution; relationships resolve missing data values. For more detailed information regarding the resolution algorithm, see the 'Graph resolution algorithm' section.

When you see the **[↑]** icon on a node label it indicates a recursive relationship. After performing a data query, the recursive node's value displays as separate values on the node label. The resolution algorithm first gets the current node's context by resolving it through a related node. With the context set, the algorithm finalizes node resolution of the current recursive node by following its recursive relationships. Additionally, recursive relationships allow the Graph View to display data values as a hierarchy.

The next section in this document describes how to setup the add-on for use with your data.

Special notation key:	
✔	Important recommendation for the use of the feature
✖	This feature is not yet available in the current release

CHAPTER **2**

Configuring the add-on

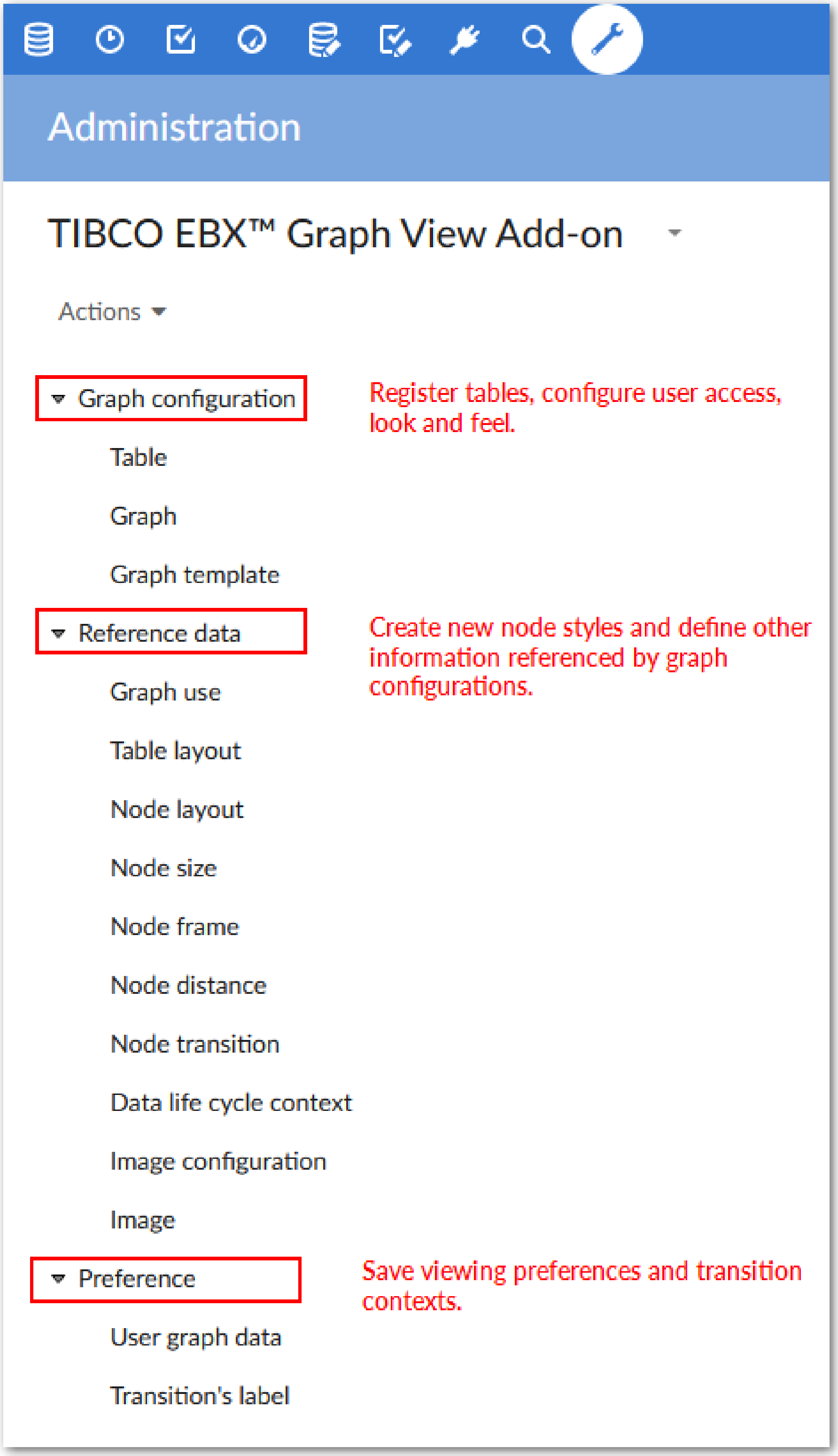
This chapter contains the following topics:

2.1 Example configurations

This section describes how to configure the add-on for initial use. The following examples take broad strokes and show a general use case. However, you can build on these principals to take a more methodical approach when configuring the add-on for your specific needs.

See the 'Graph configuration domain' reference section for descriptions of the properties that provide a fine grained control over how your data displays and what data specific users can see and query.

As shown below, all configuration settings for the add-on are located under the 'Administration' tab > 'TIBCO EBX® Graph View Add-on' dataset.



Administration

TIBCO EBX™ Graph View Add-on

Register tables, configure user access, look and feel.

Create new node styles and define other information referenced by graph configurations.

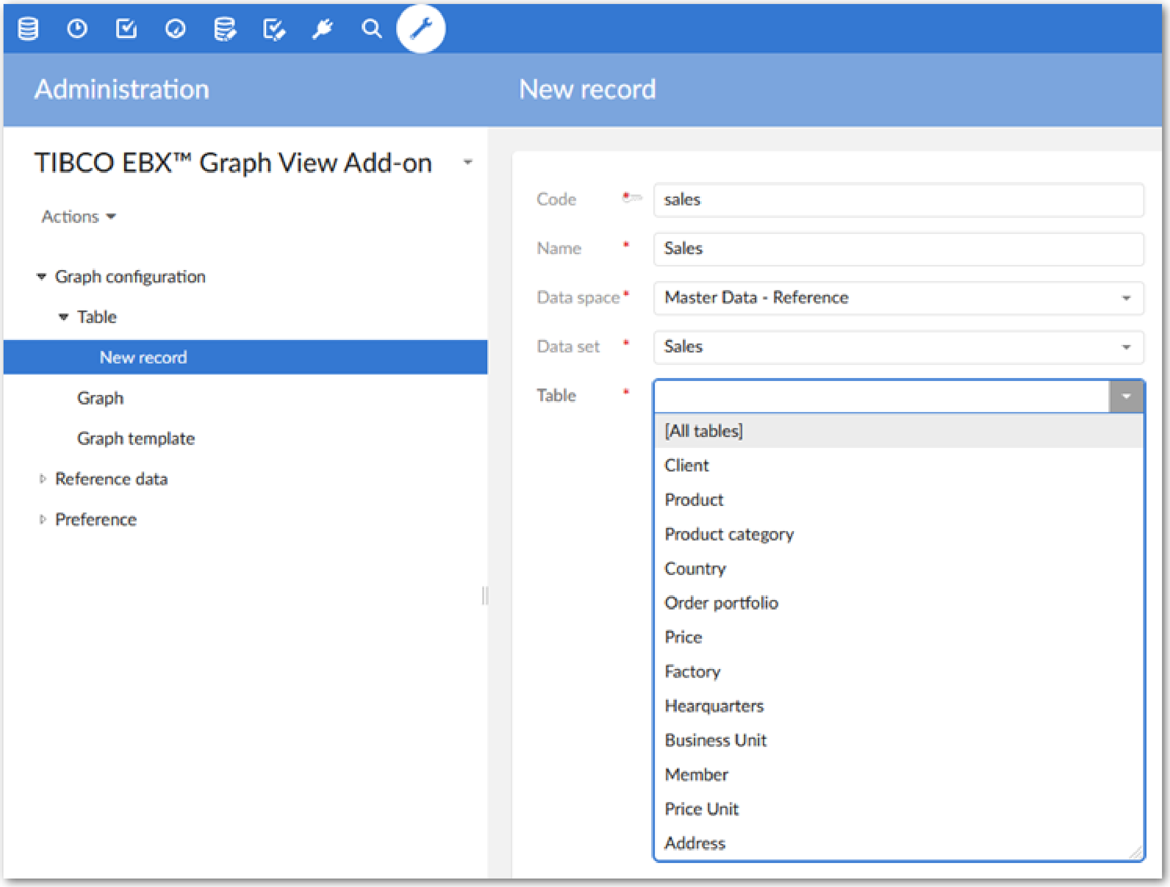
Save viewing preferences and transition contexts.

2.2 Registering tables with the add-on

By registering a table with the add-on you allow that table to display in the Graph View. The following steps describe how to register a set of tables with the add-on:

- Navigate to: 'Administration' → 'TIBCO EBX® Graph View Add-on' → 'Graph configuration' → 'Table'
- Click the '+' icon to create a new record. The screen displays an empty set of fields that you use to define the record.
- Enter a unique name in the 'Code' field (without white spaces).
- Enter a unique name in the 'Name' field (white spaces are allowed). Keep in mind that this name will be saved and used in other configuration settings. You may want to use a specific naming convention. As shown in the illustration below 'Sales' is used as the name when accessing data related to the sales department.
- Using the drop-down lists, specify the desired 'Data space' and 'Data set'.
- Select [All tables] from the drop-down list. Alternatively, you can select one table from the specified data set.
- Click 'Save' to register this table, or tables, with the add-on. Clicking 'Save and Close' saves your changes and returns you to the previous screen.

You can now display the table(s) you registered in the Graph View.



Code * sales

Name * Sales

Data space * Master Data - Reference

Data set * Sales

Table *

2.3 Managing user access to the Graph View

You may want to limit user access to the Graph View, or allow a set of users limited functionality when it comes to manipulating the Graph View. These settings are available in the 'Graph configuration' domain's 'Graph' table.

The records here contain many settings that inherit their value from a graph template. These settings deal with both Graph View appearance and user access to tables registered with the add-on. This section only outlines configuration options related to user access. For information pertaining to graph templates that define default look, feel and other behavior, see 'Customizing Graph View display and behavior templates'.

The record properties listed below the 'Graph template' drop-down list inherit their default value from a graph template located in the 'Graph template' table. Each property's inheritance can be overwritten or re-instated by toggling the small icon between a property's name and its value. This provides two places where you can edit property values—at this level and at the 'Graph template' level.

Note: You can prefabricate graph templates that cover the basic scope of your organization's needs. The templates are readily accessible via the 'Graph template' drop-down list, or you can quickly override a property or two-from an existing template-that helps you meet a certain requirement.

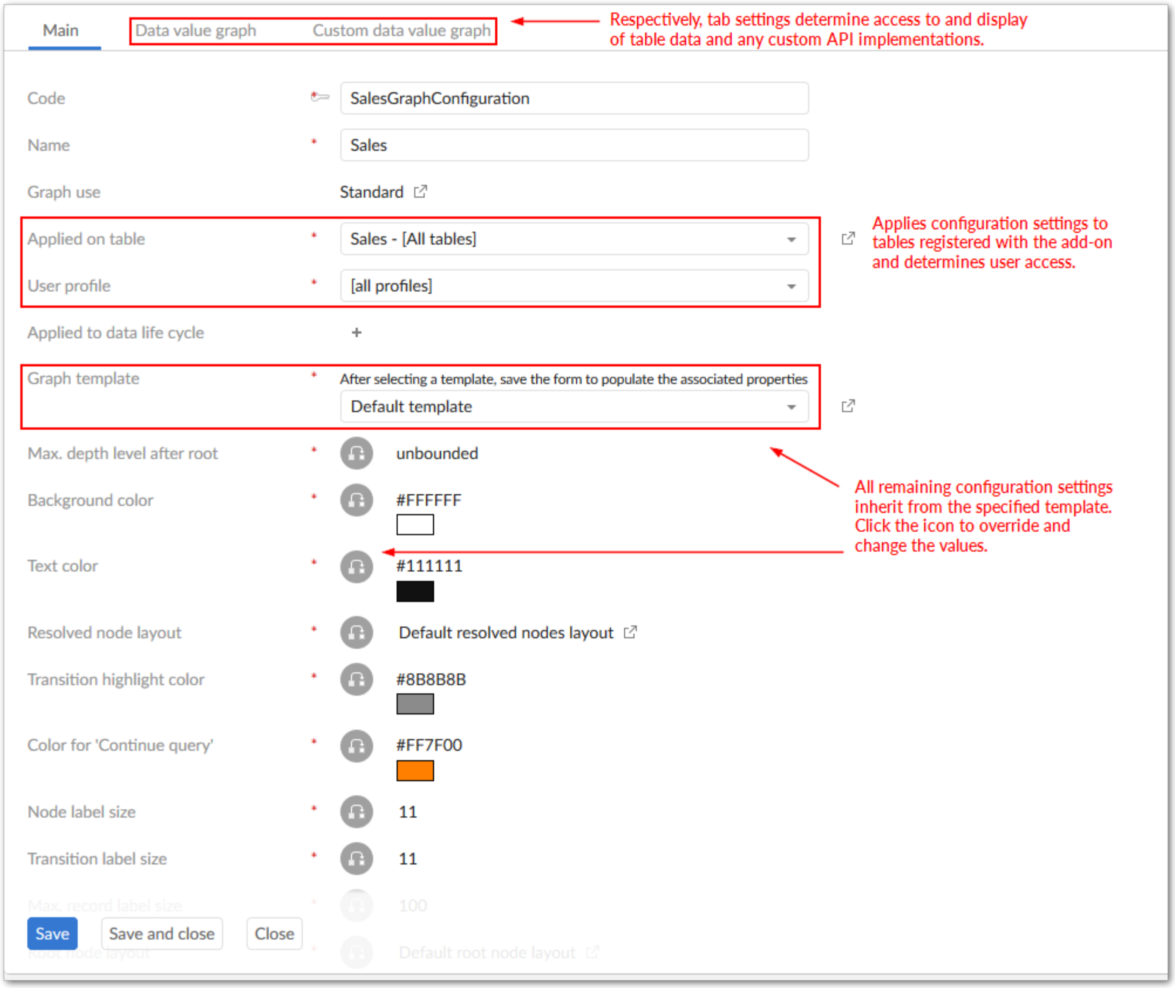
The following example makes use of a default graph template included with the add-on:

- Navigate to: 'Administration' tab → 'TIBCO EBX® Graph View Add-on' → 'Graph configuration' → 'Graph'
- Click the '+' icon to create a new record.
- Enter a code and a name in their corresponding fields. Remember, no white spaces can be used in the 'Code' field and it is a good practice to use a meaningful naming convention.
- Select a record from the 'Applied on table' drop-down menu. Your choice here determines the table, or tables to which this configuration applies. If you haven't registered any tables, you can select '+ Create' from this drop-down menu to do so. See the above section for more information on registering tables.

This list populates with names of records created in the 'Table' table. Each of these records contains tables registered with the add-on. When the list of tables is extensive, it can be helpful to use the 'Selector' option to display a tabular view of registered tables. You can then use filtering criteria and search options to find and select the desired table.

- Select either a specific user profile, or 'all profiles' from the 'User profile' drop-down list. This property specifies to which profile(s) the current configuration applies. Essentially, this determines who can access data using the Graph View.
- Select a template from the 'Graph template' property's drop-down list. Each template contains settings that determine graph behavior and display. In the illustration below, the default template is used. Note: After selecting a template, you need to click the 'Save' button at the bottom of the page to populate the remaining properties with values from the selected graph template.
- Determine whether or not users with access to this configuration can rearrange nodes and save the layout. The 'Is moveable' property inherits from the associated 'Graph template', but can be overridden here.
- The 'Data value graph' tab contains a property called 'Display node by value' which allows you to specify whether or not users can view table records in the Graph View.

- Click 'Save and close' at the bottom of the page.



Code	SalesGraphConfiguration
Name *	Sales
Graph use	Standard
Applied on table *	Sales - [All tables]
User profile *	[all profiles]
Applied to data life cycle	+
Graph template *	After selecting a template, save the form to populate the associated properties Default template
Max. depth level after root *	unbounded
Background color *	#FFFFFF
Text color *	#111111
Resolved node layout *	Default resolved nodes layout
Transition highlight color *	#8B8B8B
Color for 'Continue query' *	#FF7F00
Node label size *	11
Transition label size *	11
Max. record label size *	100
Root node layout	Default root node layout

Respectively, tab settings determine access to and display of table data and any custom API implementations.

Applies configuration settings to tables registered with the add-on and determines user access.

All remaining configuration settings inherit from the specified template. Click the icon to override and change the values.

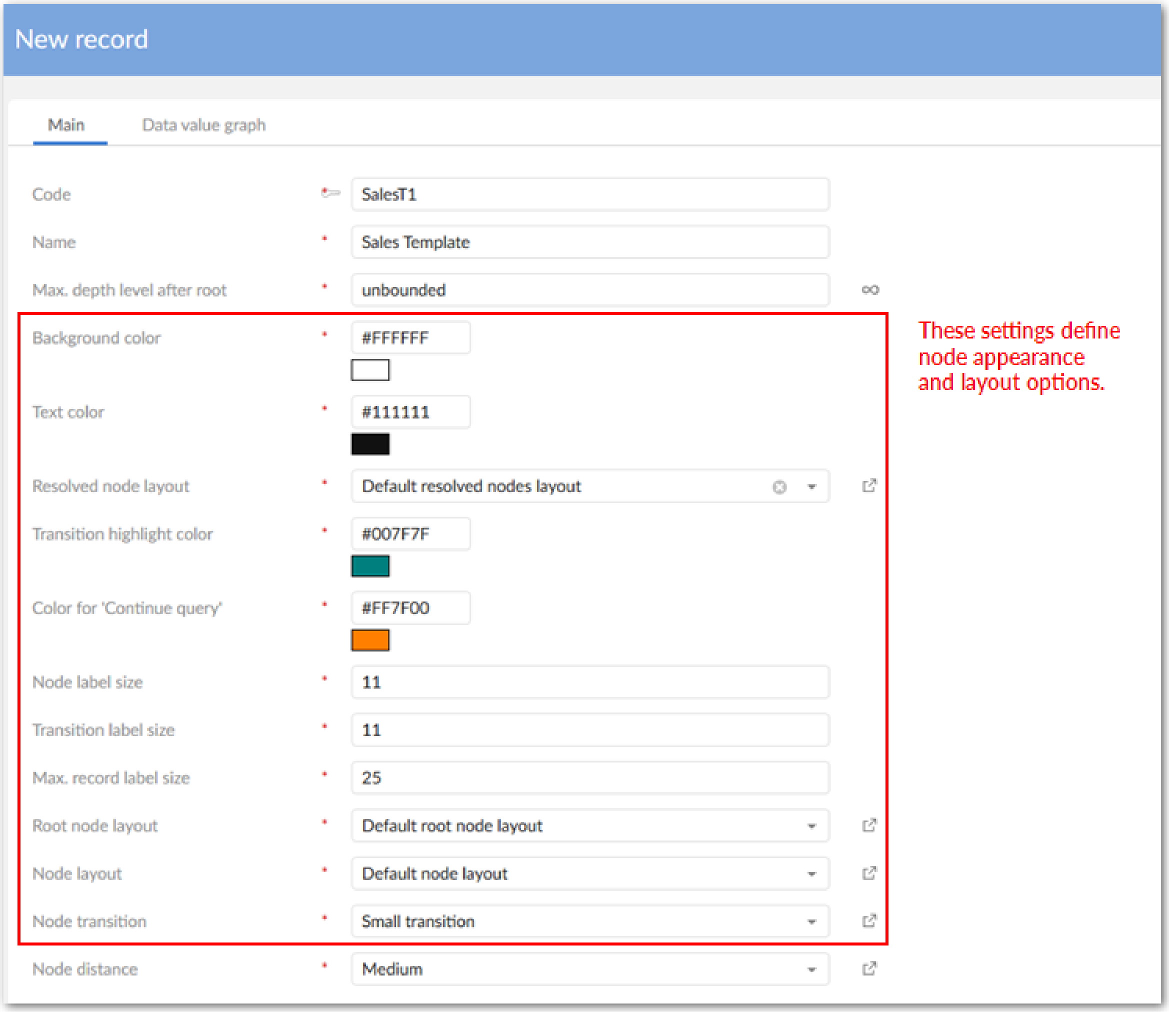
2.4 Customizing Graph View display behavior with templates

This section describes how to use a graph template to define a custom look and feel for the Graph View, as well as, set properties that can affect performance. Once a template has been completed and saved, it will be available for selection in the 'Graph template' drop-down list.

The following steps focus on a few properties. For a description of all properties, see the 'Graph template' table section.

- If you haven't already done so, select the 'Graph template' table located under the 'Graph configuration' domain.
- Click the '+' icon to add a new record to this table where you'll configure your graph display for optimum look and feel. For this example, we'll start a template from scratch.
- Add a code and name in the corresponding fields.
- The 'Max depth level after root' property limits the number of nodes that display after the root node. For this example, we'll leave the value unbounded.

- The next four properties determine graph color. You can either enter an HTML color code in the field or click in the colored box to choose a color from a color picker. Choose wisely.
- Some tables and records may have longer names that would clutter the Graph View. The **Max record label size** property allows you to specify a table label's maximum length.
- The 'Root node layout' and 'Node layout' properties let you select configurations defined in the 'Reference data' domain's 'Node layout' table. These configurations can determine node color, shape and size. See 'Using images to represent graph nodes' for information on displaying nodes as images instead of using standard shapes.



New record

Code	SalesT1
Name *	Sales Template
Max. depth level after root *	unbounded
Background color *	#FFFFFF ⬜
Text color *	#111111 ⬛
Resolved node layout *	Default resolved nodes layout
Transition highlight color *	#007F7F 🟦
Color for 'Continue query' *	#FF7F00 🟧
Node label size *	11
Transition label size *	11
Max. record label size *	25
Root node layout *	Default root node layout
Node layout *	Default node layout
Node transition *	Small transition
Node distance *	Medium

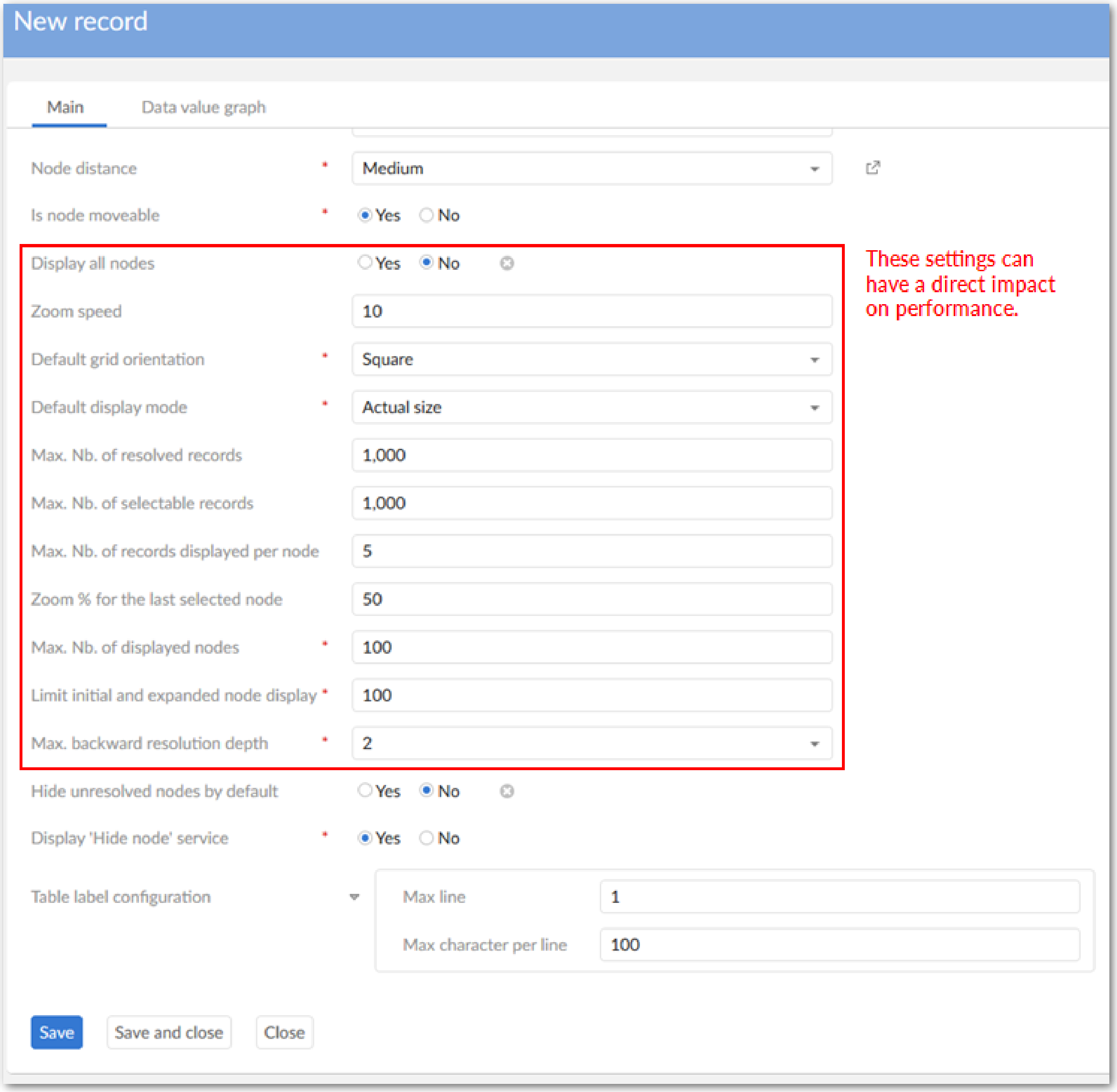
These settings define node appearance and layout options.

- The 'Is node moveable ' property determines whether users can rearrange the nodes in their views.
- Set a threshold on the number of records that can be resolved when a query runs using the 'Max. Number of resolved records' property. Depending on your system capabilities, you'll need to increase/decrease this number as this property has a direct effect on performance.
- Set a maximum number of records that can be selected when querying a table using the 'Max. number of selectable records' property This property also has an effect performance. The resolution algorithm follows all relationships and if the number of records is extremely high, system performance can be an issue.

- Another setting that has an impact on graph performance is the property—'Max. number of displayed nodes'. This property restricts the number of table, or record nodes displayed in a graph. Reducing the number of displayed nodes can improve performance.

  When the limit specified is reached a symbol (->) indicates that related nodes exist, but aren't displaying. To view them, use the 'Display graph from current node' service which acts as a paging mechanism and opens a new graph displaying the remaining nodes and relationships. The node from which the service was run becomes the new graph's root node. If the number of nodes still exceeds the specified limit, you will be able to choose which nodes you want to display (up to the specified limit).

  Note: if you are using Firefox version 3.6. this value should be set no higher than 50.
- The 'Data value graph' tab contains settings pertaining to table records and allows you to enable, or disable user ability to view data values (table records) in the Graph View.



New record

Node distance *	Medium
Is node moveable *	● Yes ○ No
Display all nodes	○ Yes ● No
Zoom speed	10
Default grid orientation *	Square
Default display mode *	Actual size
Max. Nb. of resolved records	1,000
Max. Nb. of selectable records	1,000
Max. Nb. of records displayed per node	5
Zoom % for the last selected node	50
Max. Nb. of displayed nodes *	100
Limit initial and expanded node display *	100
Max. backward resolution depth *	2
Hide unresolved nodes by default	○ Yes ● No
Display 'Hide node' service *	● Yes ○ No
Table label configuration	Max line: 1 Max character per line: 100

These settings can have a direct impact on performance.

2.5 Using images to represent graph nodes

You can further customize the Graph View by using images to represent table and record nodes. Using different configurations you can:

- Use the same image to represent all table nodes in a graph view. By default, this image is also used for table records displayed using the 'Display graph data value' service.
- Customize the display of each node independently. For example, you can use a different image, or layout to represent each table in a graph and you can use a different image for each record in a table.

Image configuration options are available in the TIBCO EBX® 'Administration' tab → 'TIBCO EBX® Graph View Add-on' domain. However, for image storage, you will need a system folder that EBX® can access. This task may require a system administrator.

The points below are covered in detail in the following sections and outline the basic steps to represent nodes with images:

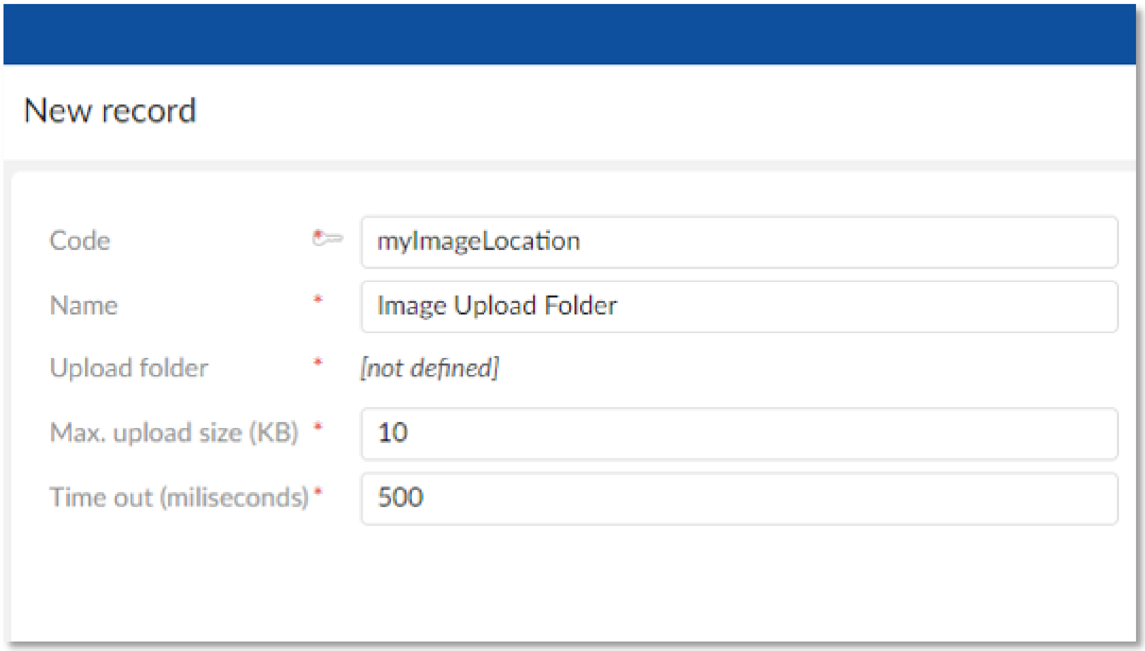
- Create a image storage folder in a location accessible by EBX® (System Administrator task)
- Upload images to this folder
- Use the same image to represent all nodes in a graph, or use a different image to represent each node

Uploading images

Before uploading an image, a dedicated folder must be in a location accessible by EBX®. A system administrator may need to complete this task. Once the folder is available, complete the following steps to upload images:

- Navigate to: 'Administration' tab → 'TIBCO EBX® Graph View Add-on' domain → 'Reference data' → 'Image configuration'.
- Click on the '+' icon to create a new record.

- Specify the location of the desired folder in the 'Upload folder' field. You can limit the file size and specify a timeout threshold using the 'Max. upload size (KB)' and 'Time out' properties, respectively.

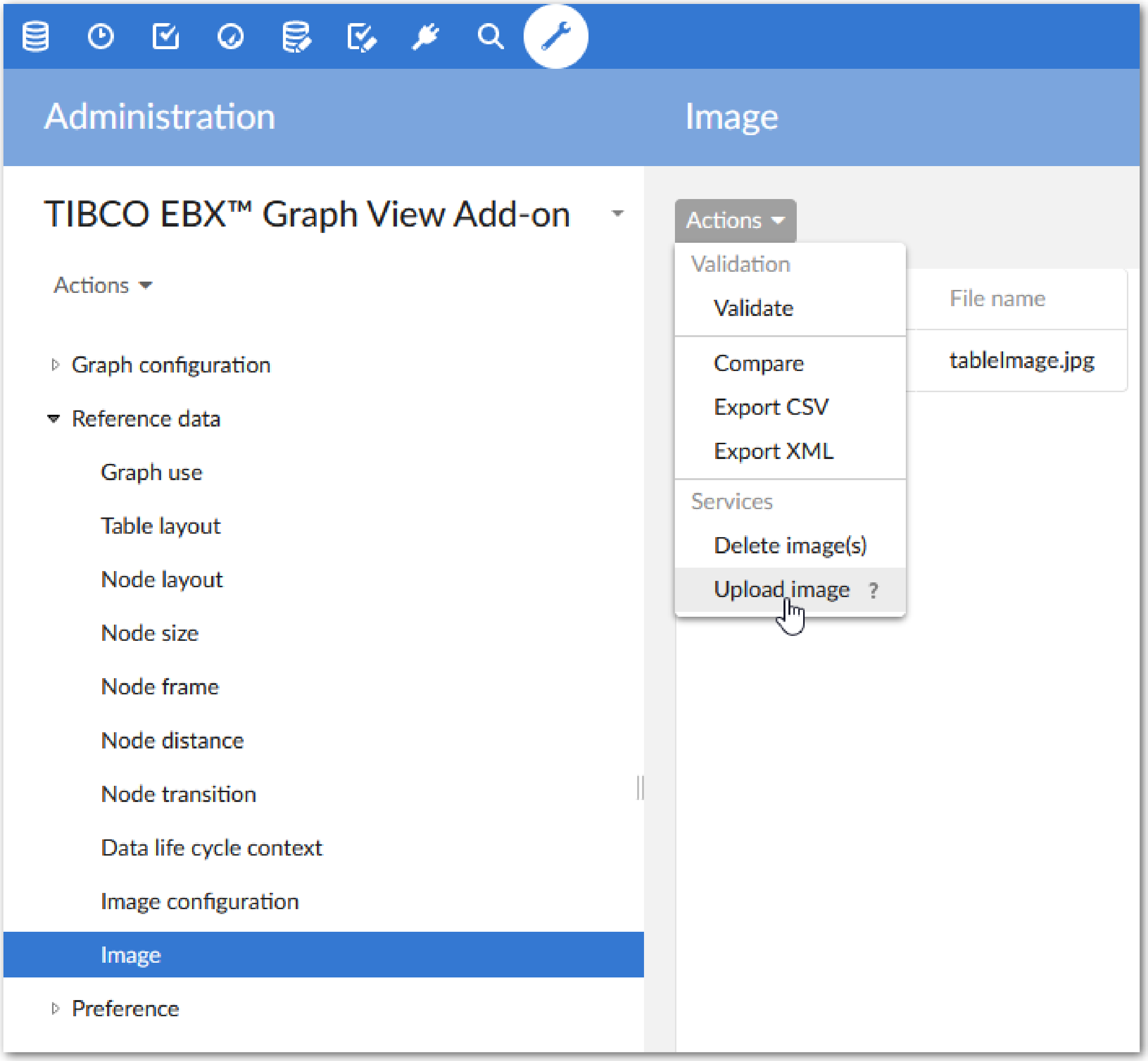


New record

Code	myImageLocation
Name *	Image Upload Folder
Upload folder *	*[not defined]*
Max. upload size (KB) *	10
Time out (miliseconds) *	500

- Click 'Save and Close' to save your settings and exit.
- Navigate to the 'Reference data' domain's ' → 'Image' table.

- The 'Services' drop-down menu allows you to upload and delete images. Selecting 'Upload image' allows you to specify a name for the image and the file's location. Note: If the 'Upload image' service isn't available, you need to specify an 'Upload folder'. See, step 3 above.



The next section describes how to use an image to represent all nodes in a 'Graph configuration'.

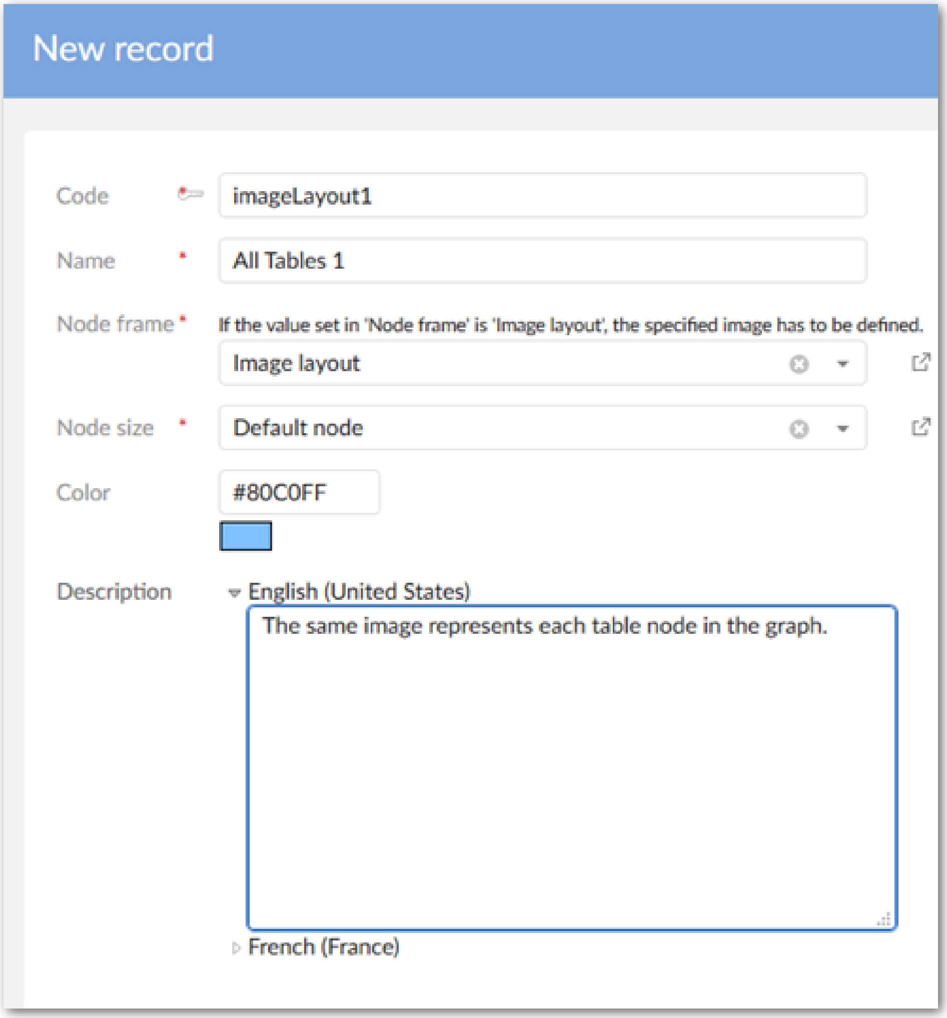
Customizing a 'Graph configuration' to use an image

The 'Reference data' domain's 'Node layout' table stores configuration settings that allow you to customize node display. Using this table, you can specify a node's frame, size, shape and color. Note: In order to complete the steps in this section, images need to be uploaded to a folder accessible by EBX®.

The following steps outline how to use an image to represent all nodes in a graph configuration:

- Navigate to 'Reference data' → 'Node layout' and select the '+' icon at the top of the page to create a new record.
- Enter a code (unique without white spaces) and a name.
- From the 'Node frame' drop-down list, select 'Image layout' and click 'Save' at the bottom of the page to display all related fields.

- Specify a display size for your image in the 'Node size' field. You can use one of the existing sizes, or click '+ Create' to open a new page where you can define a custom node size. For this example we defined a custom size called 'Images' with dimensions of 100 pixels high by 100 pixels wide.
- Select the image you want to use in the 'Image' drop-down list. This drop-down list displays the images uploaded to the dedicated folder. If you haven't uploaded any images yet, see the previous section. The graphic below gives an idea of what your settings may look like.



New record

Code: imageLayout1

Name *: All Tables 1

Node frame *: If the value set in 'Node frame' is 'Image layout', the specified image has to be defined.
Image layout

Node size *: Default node

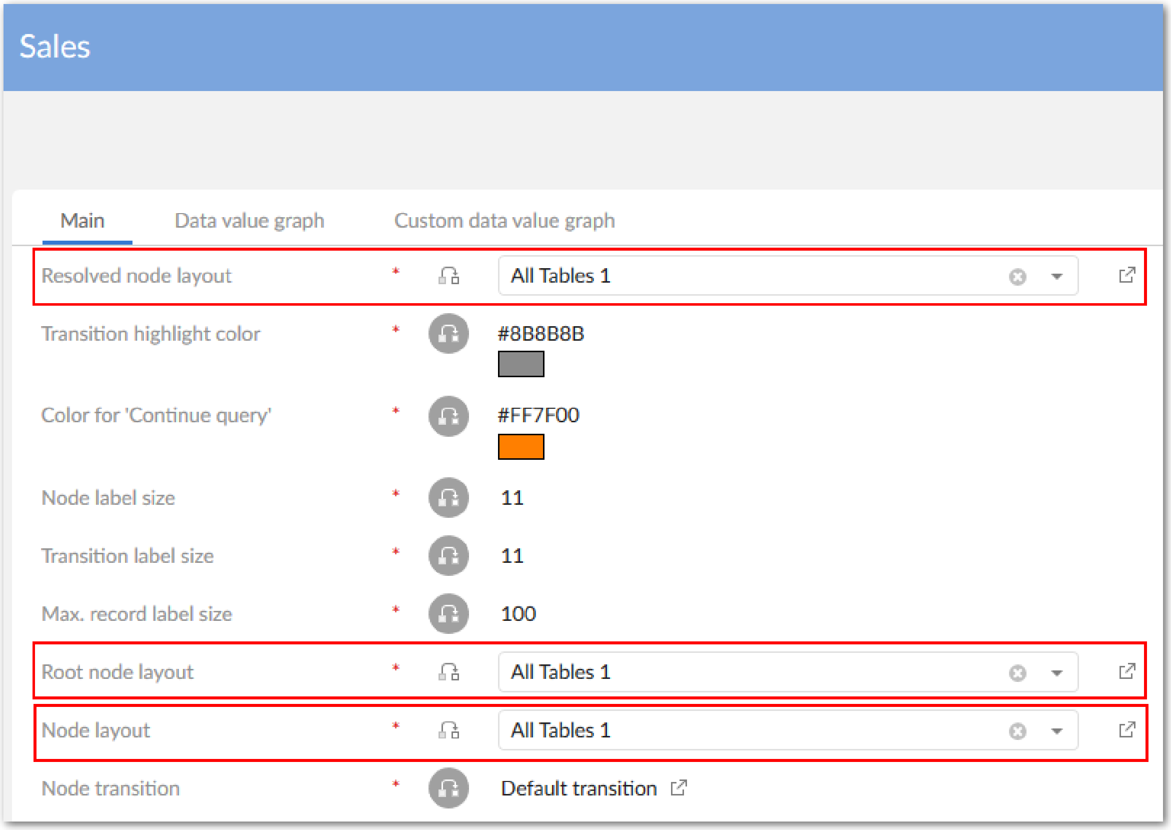
Color: #80C0FF

Description: English (United States)
The same image represents each table node in the graph.

French (France)

- Save and close.
- Navigate to the 'Graph configuration' domain's → 'Graph' table. Open the graph configuration that you want to apply this image layout to.
- The following properties derive their values from the 'Node layout' table: 'Resolved node layout', 'Root node layout' and 'Node layout'. It is common to use different sizes and colors for each type of node layout. The differences simplify graph viewing by providing a prominent visual indicator of a node's state. For simplicity's sake, we'll use the layout we just defined for all three properties.

Click the small gray box to overwrite default inheritance and select the node layout configuration you just created from the drop-down list.



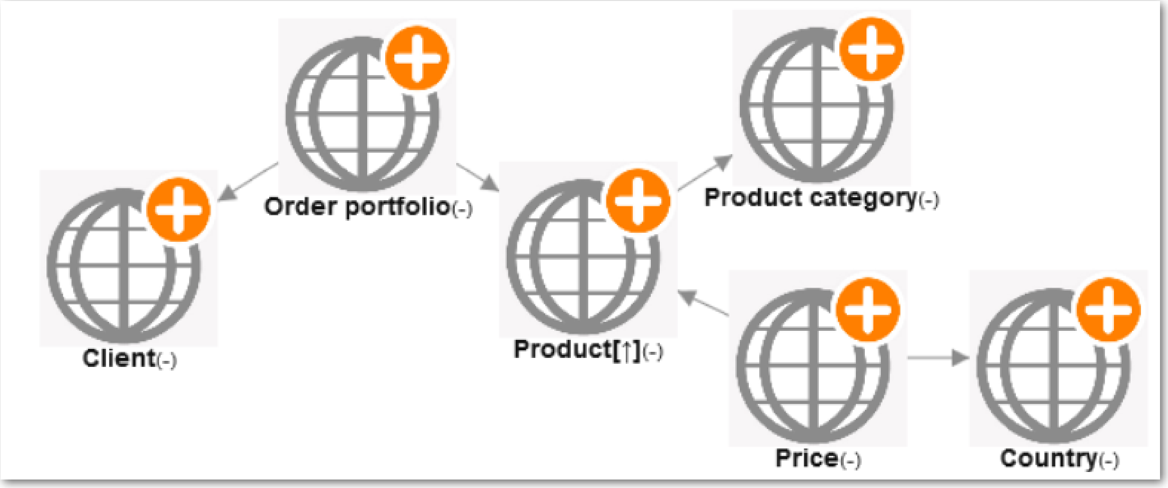
Sales

Main | Data value graph | Custom data value graph

Resolved node layout	*	All Tables 1
Transition highlight color	*	#8B8B8B
Color for 'Continue query'	*	#FF7F00
Node label size	*	11
Transition label size	*	11
Max. record label size	*	100
Root node layout	*	All Tables 1
Node layout	*	All Tables 1
Node transition	*	Default transition

- Save and close.

As shown in the image below, when you open tables in the graph view that use this graph configuration, all nodes will display as the specified image. Keep in mind that node layout configuration settings can be changed in 'Graph templates' and any graph configurations using templates inherit the node layout settings.



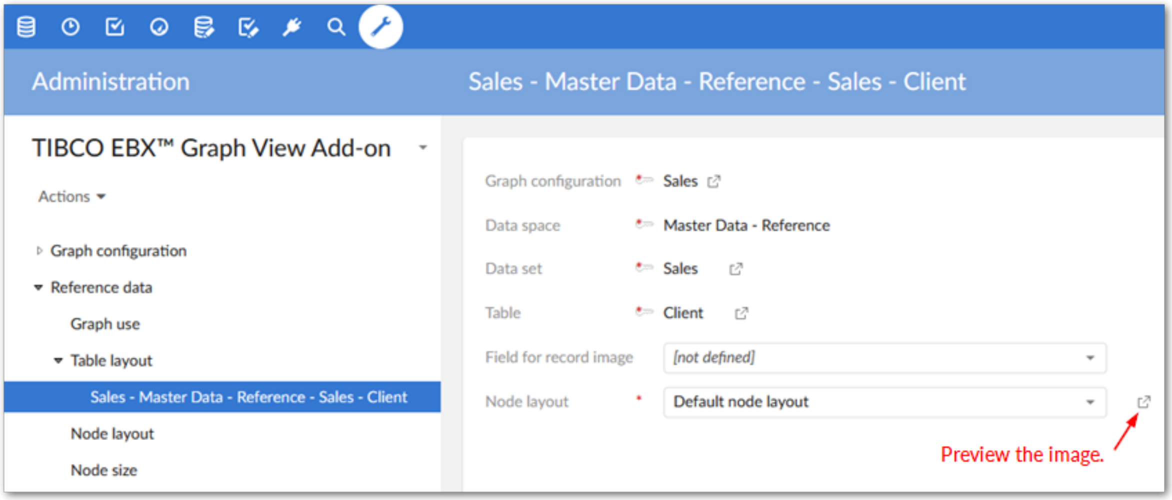
Customizing individual node display

You can customize the display of each node independently. For example, you can use a specific icon to represent a 'Client' table and use a profile picture to represent each record contained in the 'Client' table.

To display each table record as a different image, the record has to contain a field that specifies an accessible path to an image. Also, the field type must be set to URI. This behavior enables you to use locations other than the folder used to upload images. Keep in mind that the value set in the 'Image configuration' table's 'Max. upload size (KB)' property also applies to the images used to display records. So, if the images referenced for record display exceed this value, they will display as a colored rectangle. Additionally, if it takes longer to load an image than the timeout threshold specified in the 'Image configuration' table, the records display as a colored rectangle.

The following steps outline independent node display:

- Navigate to the 'Reference data' domain's → 'Table layout' table.
- Select '+' to add a record.
- Specify the 'Graph configuration', 'Data space', 'Data set' and 'Table' to which you want to apply this configuration.
- The 'Field for record image' drop-down list displays any valid URI fields from records contained in the specified table. Select the field that contains the link to the desired image.
- The 'Node layout' drop-down list displays configurations specified in the 'Node layout' table. The value you select here determines how the table displays. You can click the 'Preview' icon next to this property to open a preview window.



Administration
Sales - Master Data - Reference - Sales - Client
TIBCO EBX™ Graph View Add-on
Actions
Graph configuration
Reference data
Graph use
Table layout
Sales - Master Data - Reference - Sales - Client
Node layout
Node size

Graph configuration: Sales

Data space: Master Data - Reference

Data set: Sales

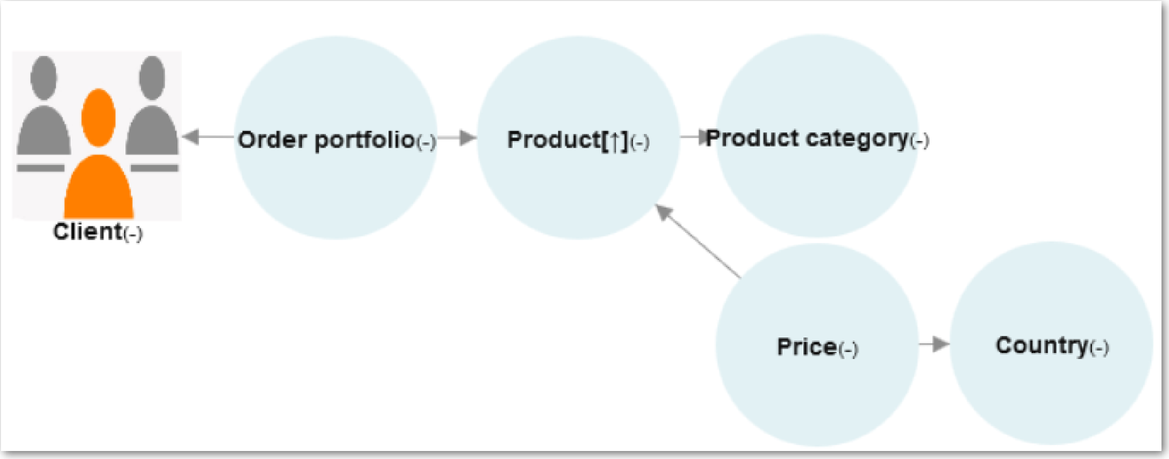
Table: Client

Field for record image: [not defined]

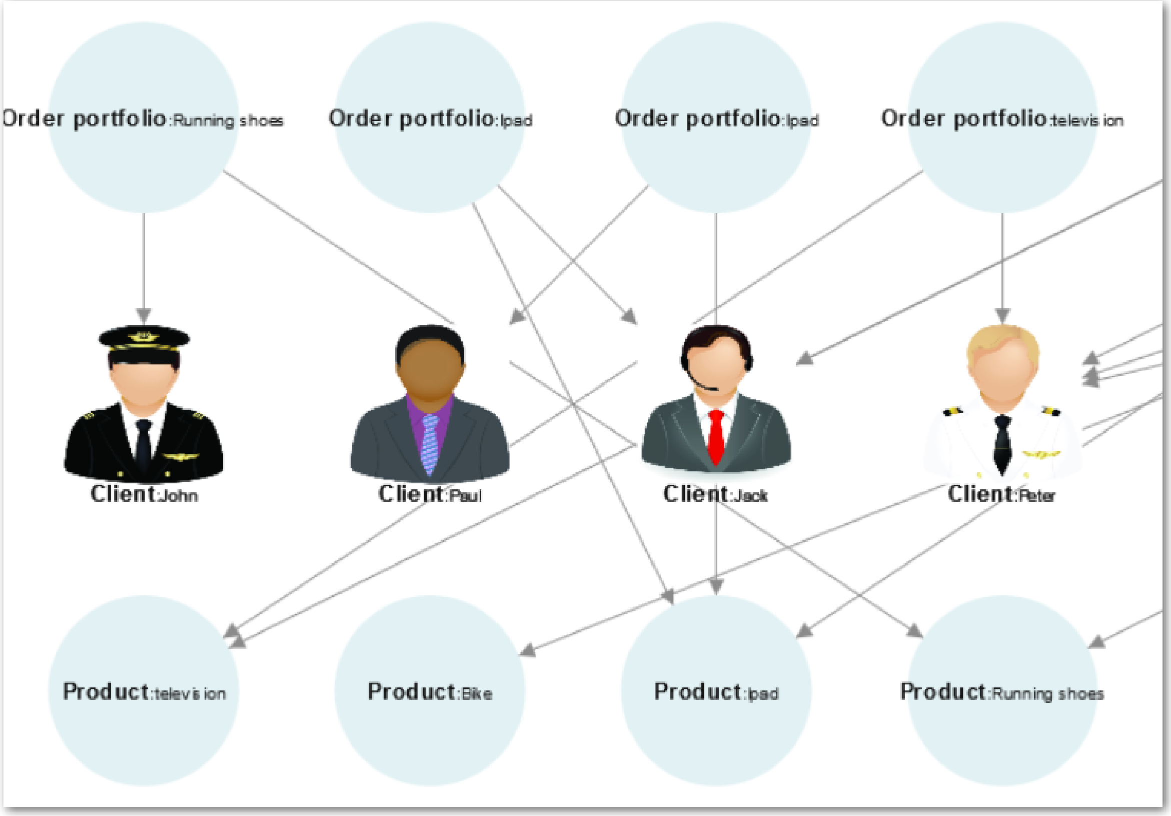
Node layout: Default node layout

Preview the image.

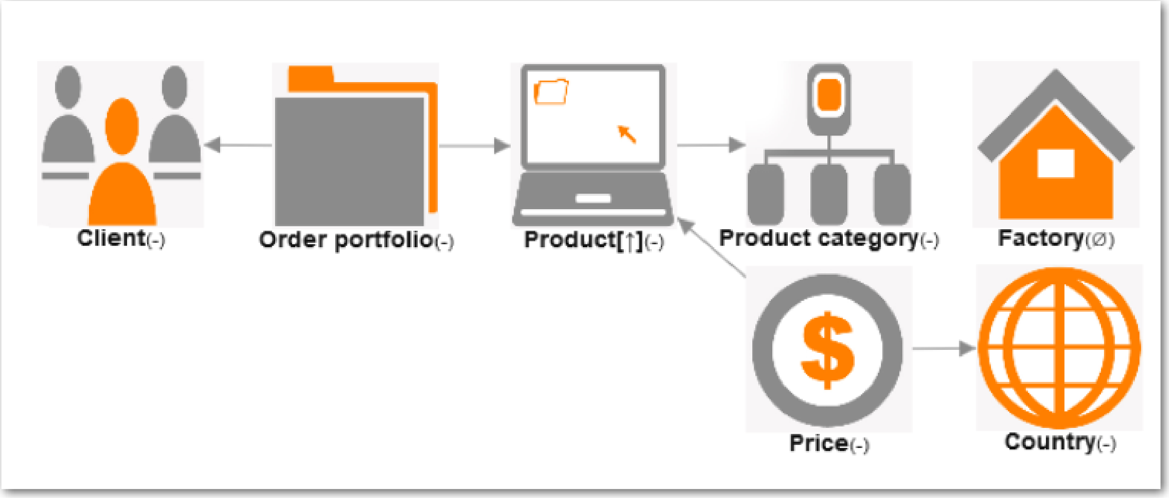
- Save and close. When you open the table in the Graph View it now shows the table as the image you chose and each record displays as its linked image.



- View the table's records using the 'Data value graphs' service. All records display as their profile picture setting.



The following image shows an example of how the graph looks when you create a table layout configuration for each table.



The above selections have covered general configuration guidelines. The next section in this document describes how you can query data when in the Graph View.

CHAPTER 3

Using the add-on

This chapter contains the following topics:

3.1 Entering the Graph View

Once you register a table with the add-on, you can execute services to display the table, or the data it contains, in the Graph View. After the graph loads you can execute data queries, have one-click data access, and can perform other actions specific to the Graph View.

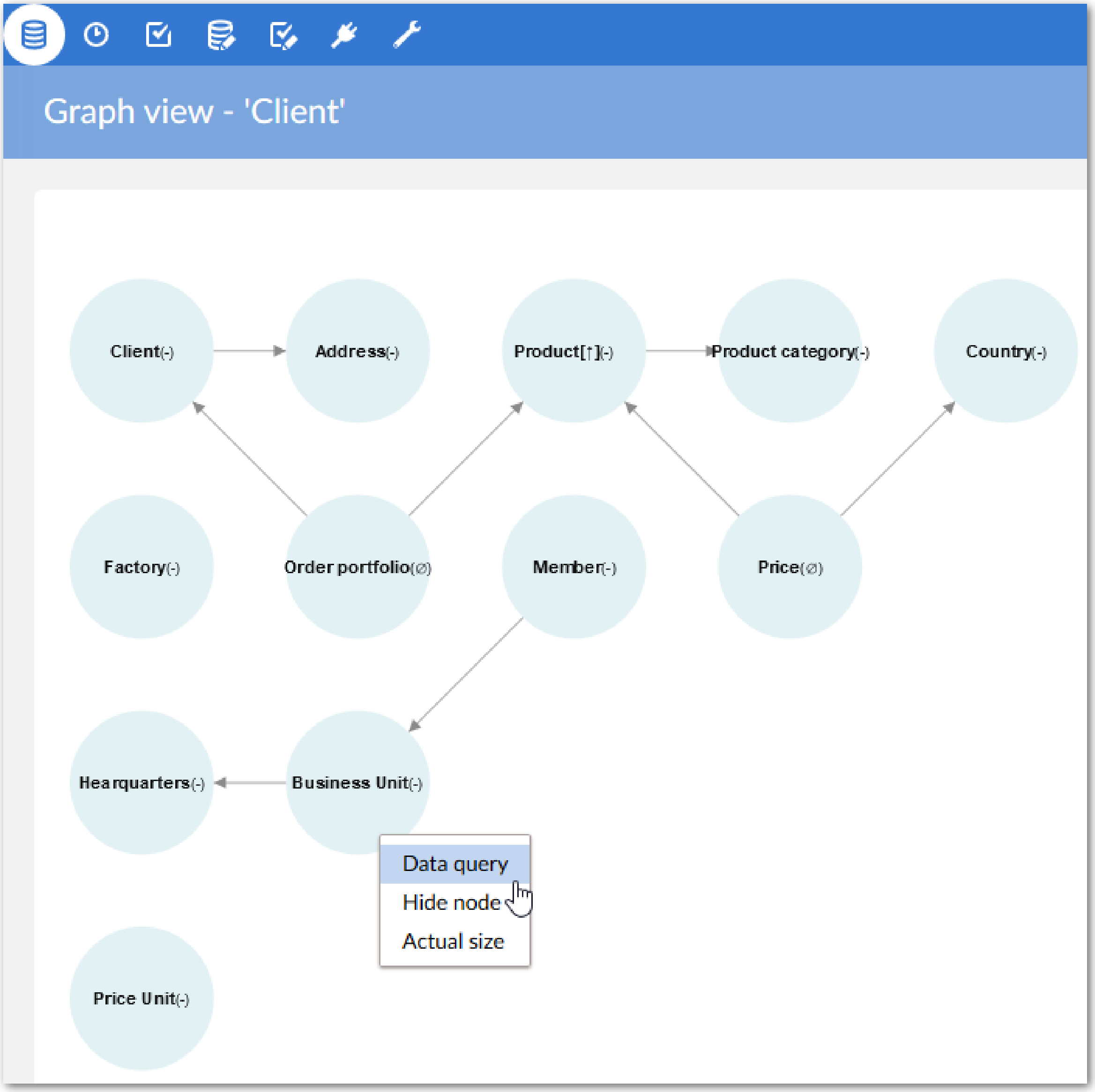
The following table lists procedures to enter the Graph View and their expected outcomes:

Expected Outcome	Procedure
Display a graph showing the data values of selected records. **Note:** You can determine whether users can view data values at the graph configuration level. See the information contained in 'Configuring the add-on' for information on configuration options.	To execute the 'Display the selected data value' service on a table: • Select a table registered with the Graph View. • Select one or more records. The records you select provide the context for the generated graph. • From the 'Actions' drop-down menu select: 'Graph View' → 'Display the selected data value'.
Display the table and its related tables in the Graph View. The table becomes the root node in the context of this graph. The following describes the behavior based on record selection: • When you don't select any records, all nodes display without relationship resolution-only the associated table names display. Unresolved nodes are indicated on each label by the '(-)' symbol. You'll need to perform a data query to complete resolution. • If you select one or more records, the Graph View uses the selected records as the graph's context and resolves nodes.	To execute the 'Display data structure' service on a table: • Select a table registered with the Graph View. • Optionally, select one or more records to automatically provide this graph with a context. • From the 'Actions' drop-down menu select 'Display data structure'.
Display graph nodes in a hierarchy based on recursive relationships, or simple join tables. To access this option from a table's menu, you must enable it at the graph configuration level. See 'Directly accessing hierarchy display from tables' for more information.	To execute the 'Display recursive hierarchy' service: • Select a table registered with the Graph View that has this service enabled in its configuration options. • From the 'Actions' drop-down menu select 'Display recursive hierarchy'.

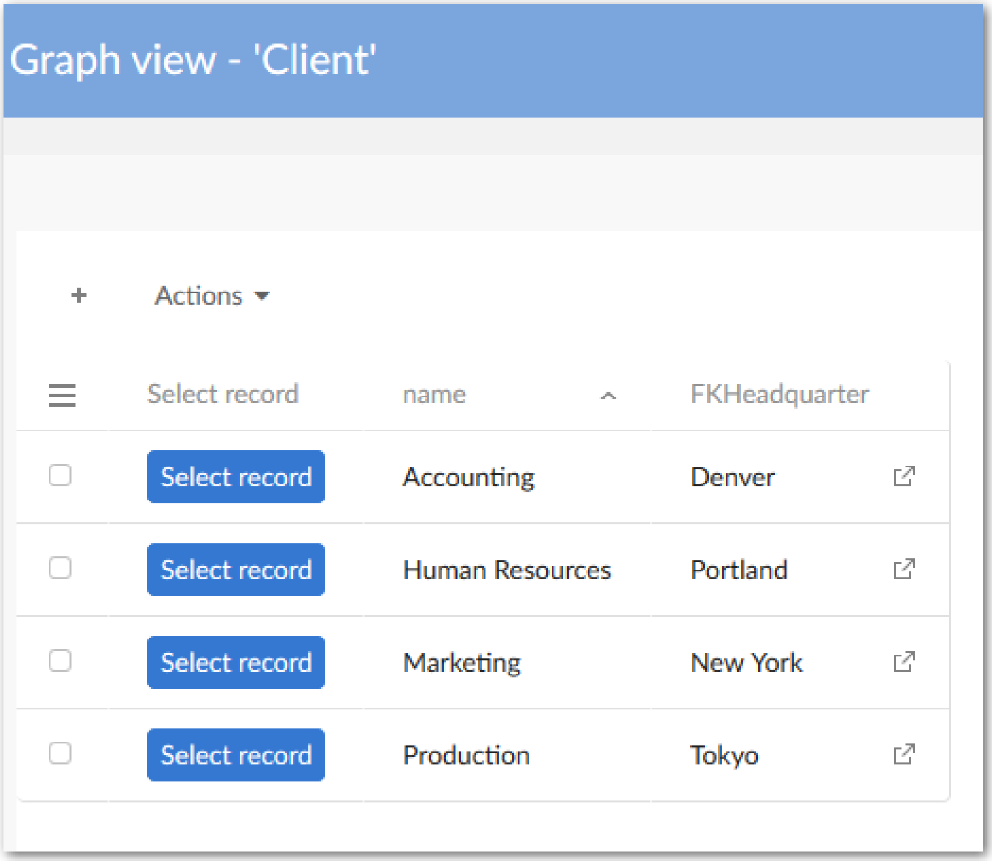
3.2 Querying and accessing data

Once the Graph View is open, you can run data queries and access data returned by queries. The following example shows how to query an unresolved node and view table records in the Graph View. You can run queries using these same steps on a resolved node. To execute a query:

- Left-click the desired node and select 'Data query'. The node's corresponding table displays.



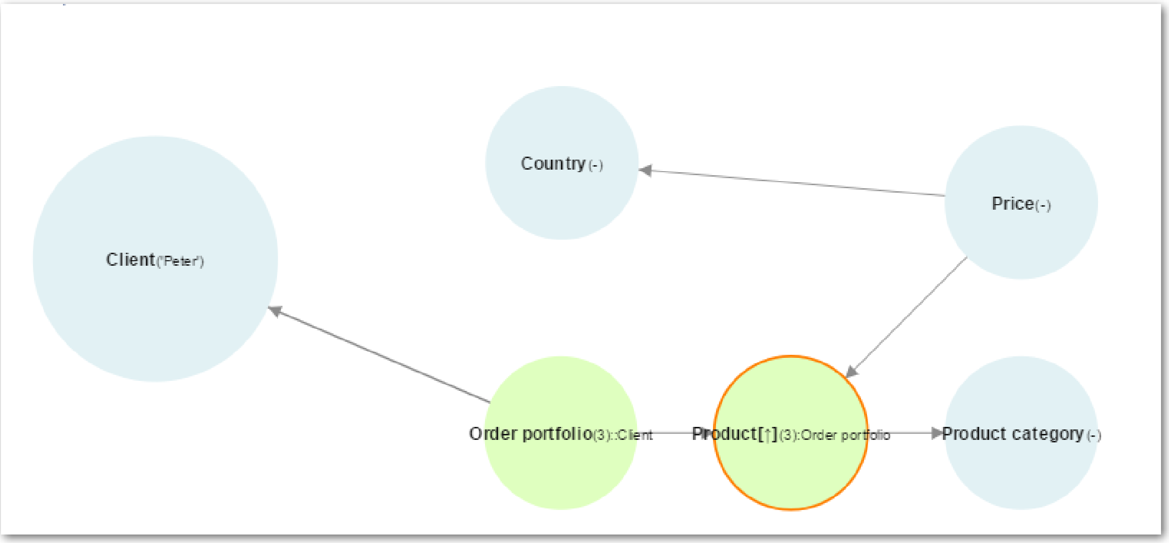
- Select one or all records in the table for the query. For this example, we'll use our favorite client 'Peter'. Because we are only using a single record, we'll perform a quick select by clicking the 'Select record' button. Alternatively, you can use the checkboxes to select multiple records and run the 'Select record(s)' service from the 'Services' menu. Note: to assist in graph performance you can limit the maximum number of selectable records using the 'Max. Number of selectable records' property. If the number of selected records exceeds this threshold, an error message displays. See 'Graph' table section for a description of this property.



Graph view - 'Client'

+ Actions ▾

☰	Select record	name ^	FKHeadquarter	
□	Select record	Accounting	Denver	↗
□	Select record	Human Resources	Portland	↗
□	Select record	Marketing	New York	↗
□	Select record	Production	Tokyo	↗

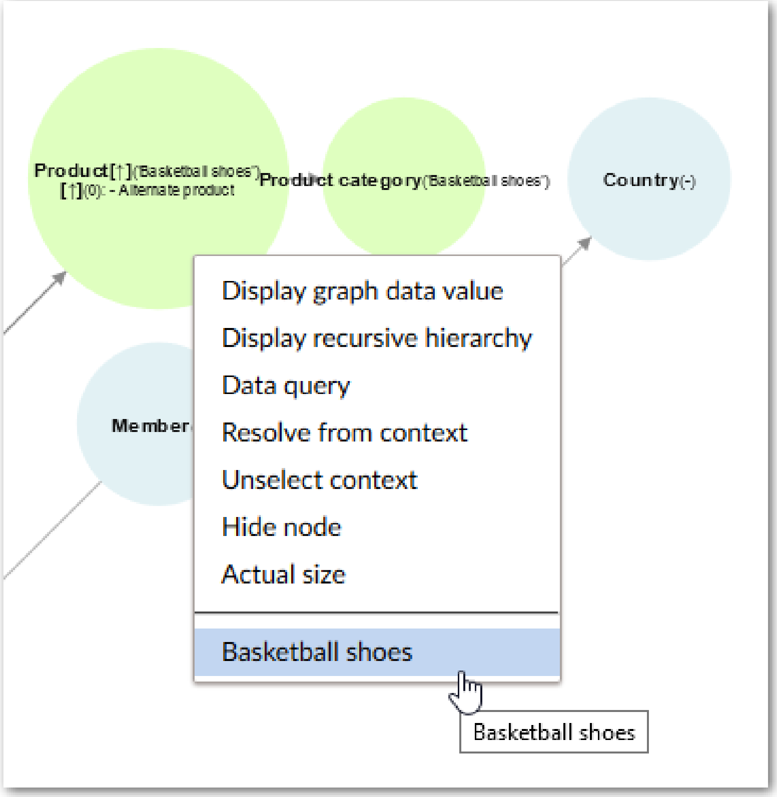


Notice that node layout has changed to indicate which nodes have resolved and the last queried node displays larger than the others. This also helps identify the context of the graph. From the graph layout we can determine that a query was initiated from the 'Client' table using the 'Peter' record. The resolution reached as far as the 'Product' table. See the 'Query continuation' section for information on continuing resolutions. You can use the settings in the 'TIBCO EBX® Graph View

Add-on configuration' → 'Reference data' → 'Node layout' table to alter display of each node type: standard node, resolved node and root node.

After a node has been resolved, the label displays the corresponding table's record. If the table contains more than one resolved record, the number in parenthesis indicates the number of resolved records in the table. Keep in mind that this is the number of resolved records or records in this context, and may not be the total number of records in the table. For example, the above node label **Product[↑](3): Order portfolio** indicates that the product table has three records. The Graph View makes accessing these records easy-see the next step.

- Left-click a resolved node and select the desired value. For this example we selected the 'Product' node.



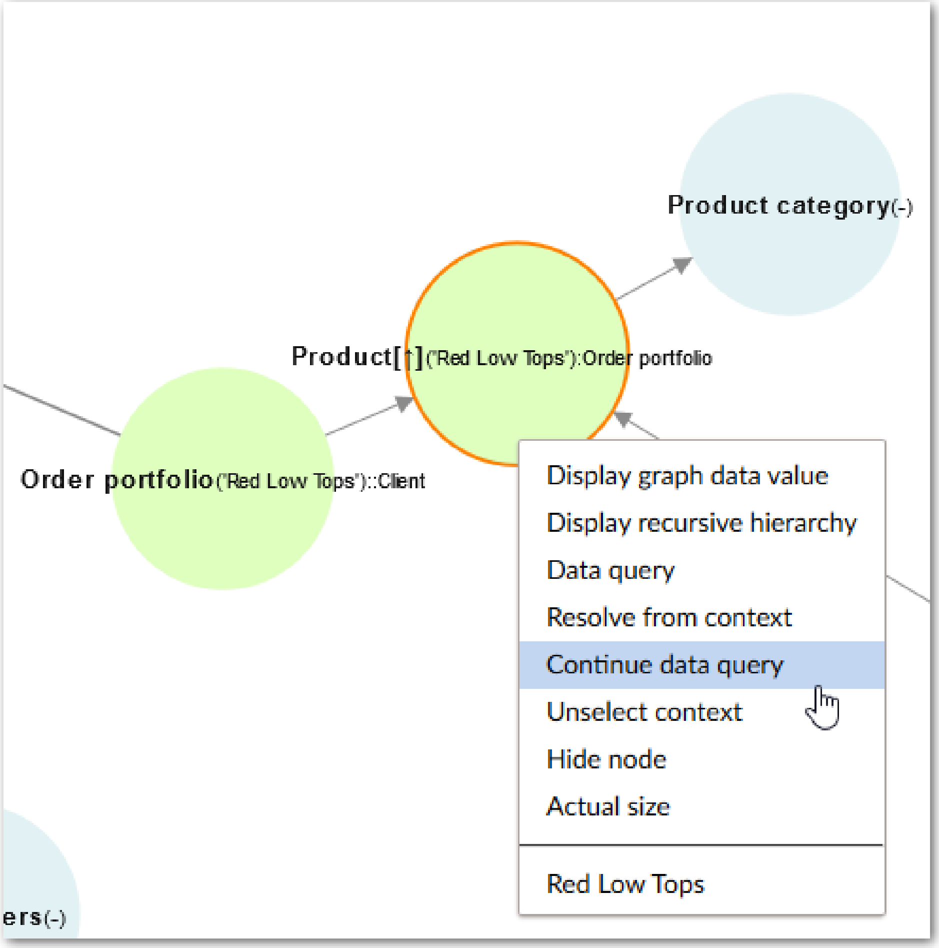
- Selecting 'Display graph data value' opens another graph that shows the data values, or table records, for current graph context. This graph only shows as far as the currently executed query (really the graph context). Note: If the number of records is large, the limit set by the 'NbMaxNodeDisplay' property may be exceeded and node display per page may be limited. You can left-click on a node and select the 'Display graph from current node' service to page through the remaining records.

Query continuation

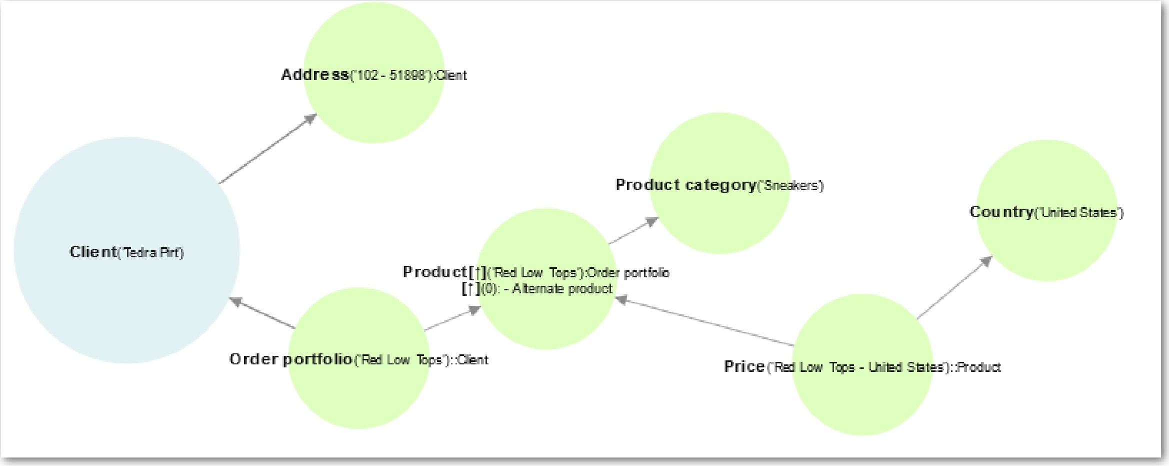
In the previous example, the query stopped at the 'Product' table. As shown in the corresponding image, an orange circle borders that node. This border indicates availability of the 'Continue data query' service. This service allows you to continue a query without resetting the graph's context and subsequently changing data values. You can customize the continuation node's border color used in each graph configuration.

To continue a query:

- Left-click on the highlighted node where the previous query stopped. In our example this is the 'Product' node.



- Select 'Continue data query'. The image below shows the result.


Address('102 - 51898'):Client
Client('Tedra Pirt')
Product category('Sneakers')
Country('United States')
Product[↑]('Red Low Tops'):Order portfolio
[↑](0): - Alternate product
Order portfolio('Red Low Tops')::Client
Price('Red Low Tops - United States')::Product

3.3 **Displaying data as hierarchies**

The Graph View can display data value nodes in a hierarchy when data structures include:

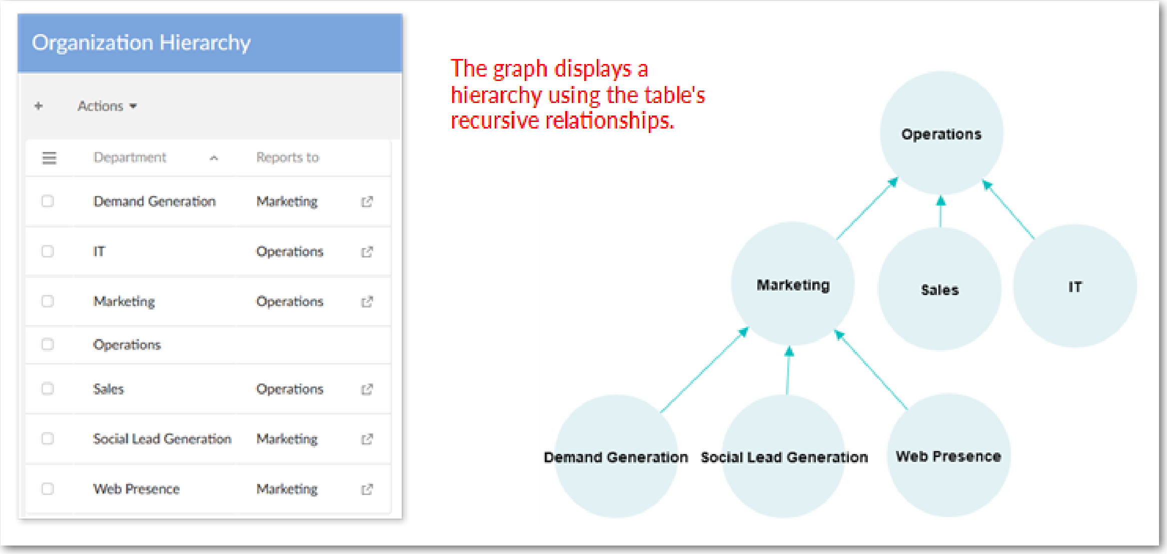
- Self-referencing foreign keys (recursive relationships)
- Foreign key relationships
- Simple Join tables (multiple foreign keys from a source table to a target)

Self-referencing foreign key hierarchies

A self-referencing foreign key, or recursive relationship occurs when a foreign key references its containing table. In these instances, the Graph View creates a hierarchy of the table's data values based on the following guidelines:

- Top level (parent) node: This node has the most nodes referencing it, or it does not refer to any other nodes.
- Sub-level (child) nodes: The Graph View follows the relationships to create the child levels. The first level of child nodes all directly reference the top level node. Each subsequent child level holds a relationship to a node in the level above it.

The following image shows an example of a hierarchy based on recursive relationships shown in the table:



Organization Hierarchy

+ Actions ▾

Department	Reports to
Demand Generation	Marketing
IT	Operations
Marketing	Operations
Operations	
Sales	Operations
Social Lead Generation	Marketing
Web Presence	Marketing

The graph displays a hierarchy using the table's recursive relationships.

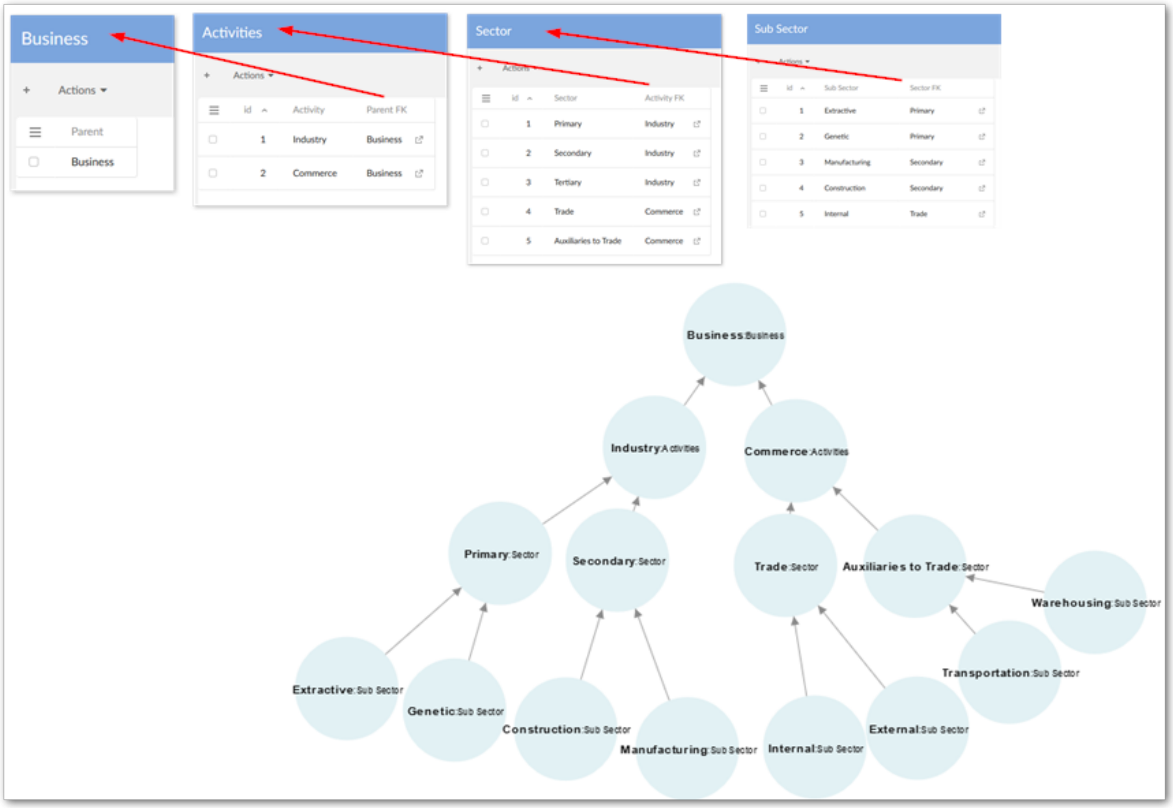
To show a hierarchy based on a recursive relationship:

- Run the 'Display recursive hierarchy' service located in one the following locations:
  - At the table level. From a table's 'Actions' menu, select 'Graph View' → 'Display recursive hierarchy'. Before running this service at the table level, you need to enable it via the table's configuration. See the section below, 'Directly accessing hierarchy display from tables', for more information.
  - At the node level. While viewing a graph, you can left-click on nodes that have recursive relationships (indicated by the [↑] symbol) and select 'Display recursive hierarchy'.
- After running the service, a list of recursive relationships contained in the table displays. Select which relationship (if more than one recursive relationship exists) to base the hierarchy on.
- Decide if you want to qualify the relationships shown in the graph by displaying values from a selected field on the transitions between nodes. If you do want to use a qualifying field, check the 'Field used to qualify the relationship' box and click the desired field's radio button.

- You can check 'Save for reuse' to retain the selected options. When satisfied with the configuration, click 'Submit'.

Hierarchical node arrangement based on foreign keys

A hierarchical arrangement of data value nodes based on foreign keys—when no recursive relationships exist—puts root table records at the topmost hierarchy level. The Graph View considers the table from which you ran the 'Display graph data value' service as the root table. Following the foreign key relationships between tables, each subsequent node level includes nodes from the child tables. As with the orientation options for graphs showing data values, each hierarchy row contains nodes from the same table.



Business

Parent
Business

Activities

id	Activity	Parent FK
1	Industry	Business
2	Commerce	Business

Sector

id	Sector	Activity FK
1	Primary	Industry
2	Secondary	Industry
3	Tertiary	Industry
4	Trade	Commerce
5	Auxiliaries to Trade	Commerce

Sub Sector

id	Sub Sector	Sector FK
1	Extractive	Primary
2	Genetic	Primary
3	Manufacturing	Secondary
4	Construction	Secondary
5	Internal	Trade

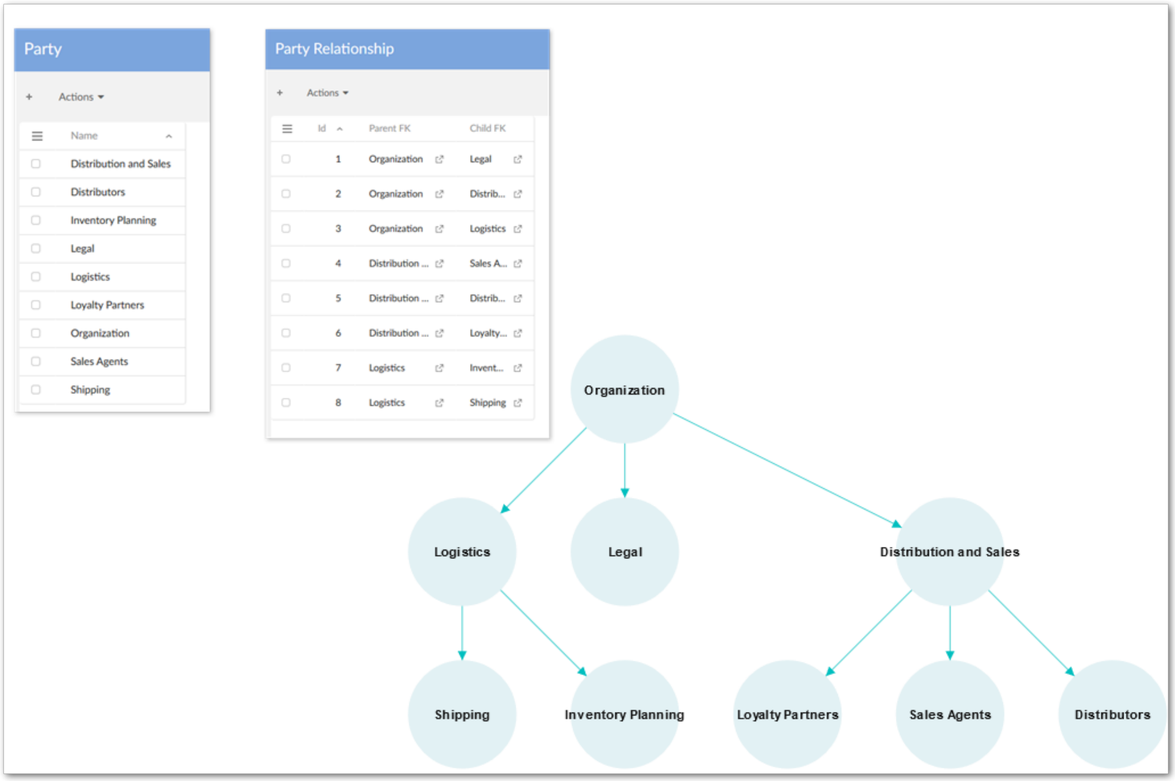
You can display a data value node hierarchy using the following methods:

- Manually: When viewing a graph of data values, from the 'Services' menu, select 'Arrange nodes' → 'In a hierarchy'.
- Automatically: Open the desired graph configuration's 'Data value graph' tab and set the 'Default node orientation' property to 'Hierarchy'. For more information on default node orientations, see the 'Changing default node orientation' section.

Hierarchies from simple join tables

Your data structure contains a simple join table when a source table holds multiple foreign keys that refer to the same target table. The Graph View allows you to select one of these relationships and generates a hierarchical view of the corresponding data values.

The following image shows a graph based on the 'Relation Party' table's 'Parent' and 'Child' fields which hold foreign keys to the 'Party' table.



Party

☰	Name
□	Distribution and Sales
□	Distributors
□	Inventory Planning
□	Legal
□	Logistics
□	Loyalty Partners
□	Organization
□	Sales Agents
□	Shipping

Party Relationship

☰	Id	Parent FK	Child FK
□	1	Organization	Legal
□	2	Organization	Distrib...
□	3	Organization	Logistics
□	4	Distribution ...	Sales A...
□	5	Distribution ...	Distrib...
□	6	Distribution ...	Loyalty...
□	7	Logistics	Invent...
□	8	Logistics	Shipping

To render a graph showing values from simple join tables as a hierarchy:

- Run the 'Display recursive hierarchy' service located in one the following locations:
  - At the table level. From a table's 'Actions' menu, select 'Graph View' → 'Display recursive hierarchy'. Before running this service at the table level, you need to enable it via the table's configuration. See the section below, 'Enabling hierarchy access at the table level', for more information.
  - At the node level. While viewing a graph, you can left-click on nodes that have recursive relationships and select 'Display recursive hierarchy'.
- After running the service, a list of foreign key relationships contained in the table displays. Select which relationship (if more than one exists) to base the hierarchy on. The records that belong to the foreign key you select display as the parent of the remaining foreign keys.
- Decide if you want to use an additional field to qualify the relationship. When you choose a field here, node transition labels display the field's values that correspond to each relationship. If you want to display this information, select the check box. A list of all fields in the table displays. Select the radio button corresponding to the desired field.
- You can check 'Save for reuse' to retain the selected options. When satisfied with the configuration, click 'Submit'.

Directly accessing hierarchy display from tables

The 'Display recursive hierarchy' service does not automatically show in the Graph View sub-menu for all tables registered with the add-on. A table must have a recursive or a simple join relationship to display as a hierarchy in the Graph View. And, you must make the Graph View check to determine that these types of relationships exist. Use the following instructions to have the Graph View check for these relationships (you should perform this action after updates to the data model that may affect these relationships):

- Navigate to 'Administration' → 'TIBCO EBX® Graph View Add-on configuration' → 'Graph configuration' → 'Table' and select the check box corresponding to the table configuration.
- From the 'Actions' menu's 'Services' group, select 'Update hierarchy view'.

After running the service, a confirmation screen displays a list of tables checked by the service. A value of 'Yes' in the 'Result' column indicates the table met the requirements and you can run the 'Display recursive hierarchy' from its 'Actions' menu.

When the data model changes, you have to run this service again to get the update accordingly.

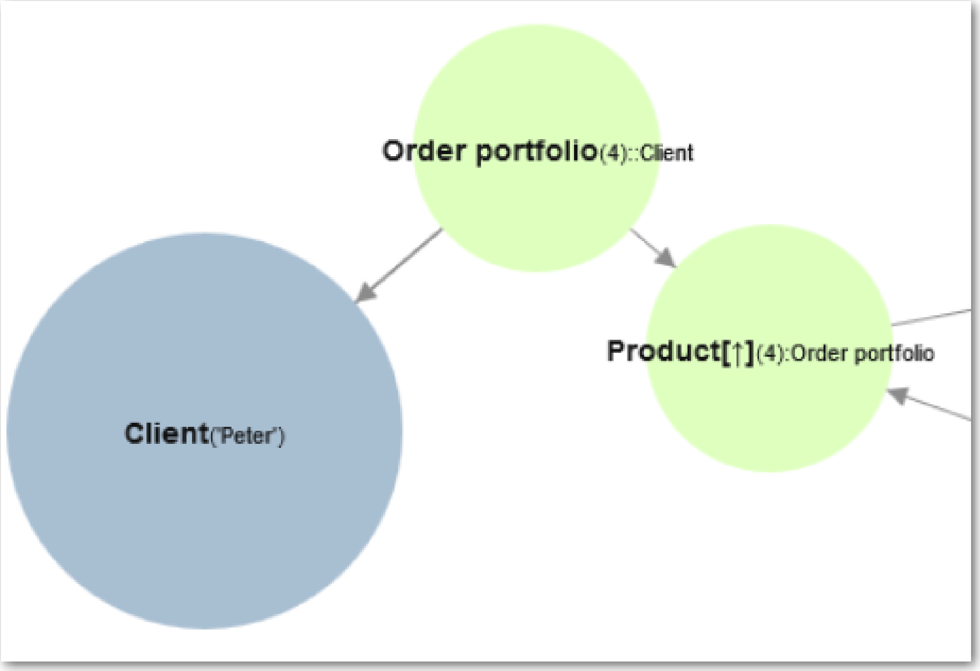
3.4 **Understanding node labels and symbols**

Each node label conveys specific information about that node, such as how many records its corresponding table contains, whether there are reflexive relationships, whether it has undergone resolution and the resolution direction.

In reference to the below illustration, the following examples translate two node labels:

- The 'Product' node label — shows the table contains '(4)' records and ': Order portfolio' indicates that resolution has occurred in a forward direction (forward resolution) through the path indicated by the node label: 'Product' > 'Order Portfolio'. Note: this is the number of resolved records, and may not be the total number of records in the table.
- The 'Order portfolio' node label — shows that the table contains '(4)' records and ' :: Client' indicates that resolution has occurred in a reverse direction (backwards resolution) through the path indicated by the node label: 'Order portfolio' > 'Client'.
- The 'Product' node label — contains the symbol [↑]. This symbol indicates the node contains a recursive relationship.

- If a node has the (→) symbol, it indicates the node has a relationship that isn't displayed. Using the 'Display graph from current node' service, allows you to view the hidden nodes.



3.5 'Services' menu

When in the Graph View the 'Services' drop-down menu contains a list of executable services that can be very useful. Please, see the reference section's 'Services' menu for more information regarding each service.

3.6 Optimizing node display

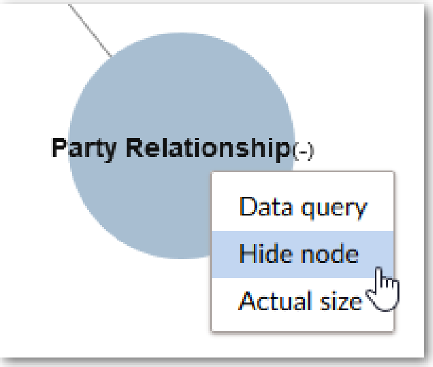
At times, you may want to temporarily hide nodes that don't provide additional value and only serve to clutter your graph. Other times, you could more readily read and understand a graph if its nodes displayed differently. To meet these challenges, the Graph View allows you to hide nodes and change a graph's display orientation. The following sections describe how to work with these options:

- Showing and hiding nodes
- Expanding and collapsing nodes
- Specifying graph display modes
- Changing default node orientation
- Configuring default grid display

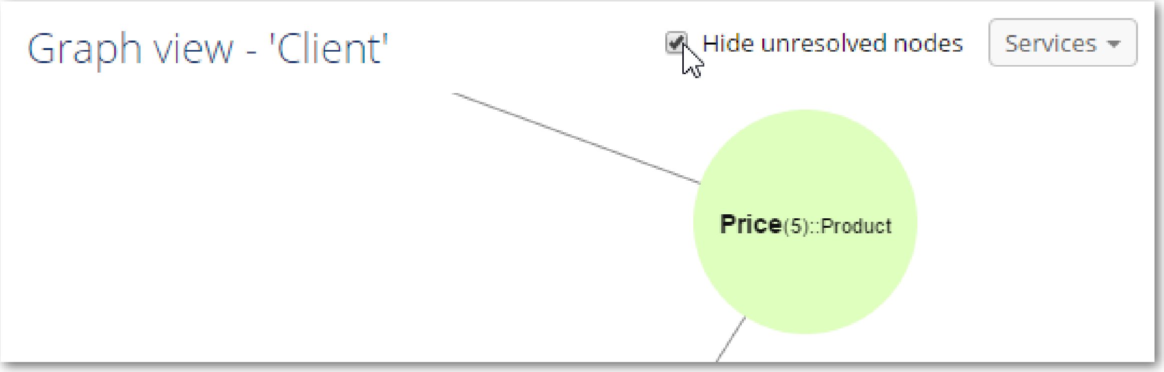
Showing and hiding nodes

The Graph View provides the following two ways to hide standard nodes:

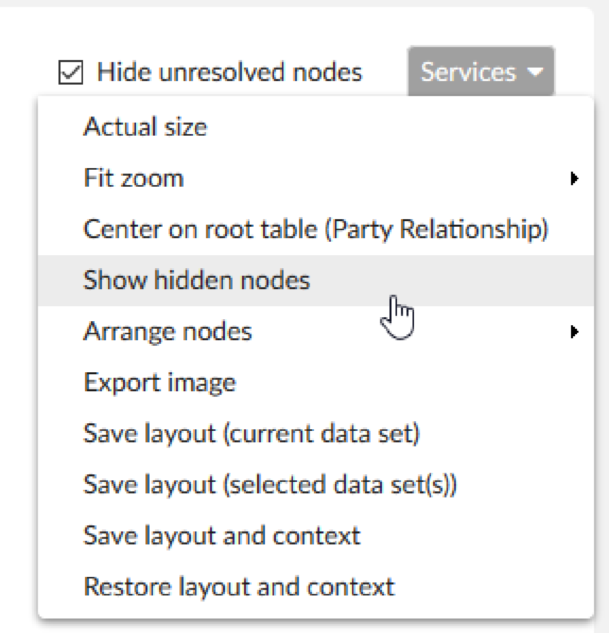
- You can hide nodes individually, by clicking on the desired node and selecting 'Hide node'. Graph configuration options allow you to enable, or disable this functionality using the 'Display 'Hide node' service'.



- You can hide all unresolved nodes in a graph by selecting the 'Hide unresolved nodes' check box next to the 'Services' menu. At the graph configuration level, you can auto-enable this feature causing the Graph View to automatically hide unresolved nodes after you perform a query.



At anytime, you can open a graph's 'Services' menu and select 'Show hidden nodes' to reveal previously hidden nodes. It does not matter what methods you used to hide nodes; running this service displays all hidden nodes.


☑ Hide unresolved nodes
Services ▾
Actual size
Fit zoom ▸
Center on root table (Party Relationship)
Show hidden nodes
Arrange nodes ▸
Export image
Save layout (current data set)
Save layout (selected data set(s))
Save layout and context
Restore layout and context

Expanding and collapsing nodes

When viewing a graph that displays recursive relationships, you can expand and collapse nodes to view/hide related nodes. Administrators can configure display behavior that affects the number of nodes that display at any one time by:

- determining the total number of nodes a graph can display. The 'Max. Nb. Of displayed nodes' property, located in a 'Graph' and 'Graph template' configuration, sets this limit.
- specifying how many nodes display when the graph initially loads and limiting the number of nodes that display when a user expands a node. The 'Limit initial and expanded node display' property, located in a 'Graph' and 'Graph template' configuration, determines these limits. When users expand nodes, the nodes display according to the selected graph orientation. See the *Changing default node orientation* section for more information on setting the orientation. When the graph orientation is set to:
    - Square, Default grid — expanded nodes display in randomly placed positions. You can run the arrange service to match the current orientation, or choose a different node arrangement.
    - Center aligned hierarchy — nodes automatically re-arrange to accommodate the expanded nodes.
    - Horizontal, Vertical and Hierarchy — expanded nodes are automatically oriented according to the current arrangement settings.

The expand/collapse options are only available for data values in a recursive hierarchy. See the *Showing and hiding nodes* section for options in other types of graphs. To expand a node and view related nodes:

- Left-click a node that has collapsed related nodes. Underneath the node's label, a number inside of parenthesis (< 6) indicates there are collapsed nodes.

- Select 'Expand nodes'. The number of nodes that display depends on this graph's configuration. For example, a node may have five collapsed related nodes. However, if the 'Limit initial and expanded node display' property is set to '4', only four of the five nodes will expand. You can run the service again to expand the fifth node.

You can collapse any node that has displayed related nodes. When you click a node and select 'Collapse nodes', all nodes that have forward transitions from the selected node and any node related to these forward nodes collapses. If at anytime you would like to display all available nodes, select 'Show hidden nodes' from the 'Services' menu.

Specifying graph display modes

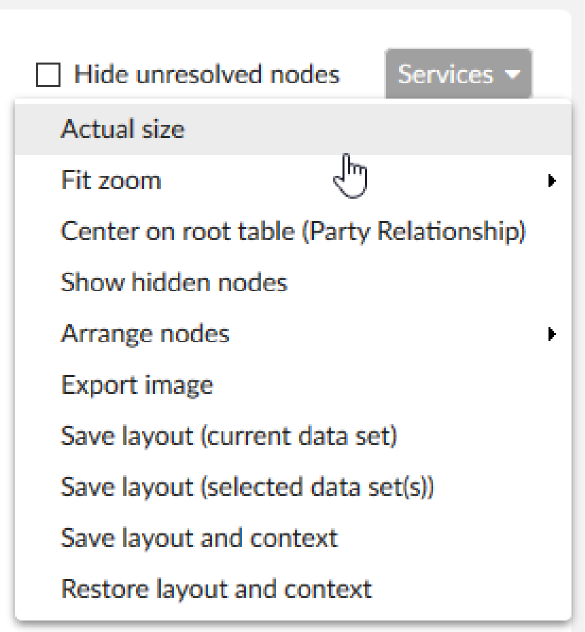
Some of your graphs may only have a few nodes while others have a plethora. Upon opening a graph containing many nodes, some nodes can render outside of the graph display area. When this occurs, you have to either zoom out or drag the graph to see these nodes. On the other hand, if a graph with few nodes opens to a distant zoom you might have to zoom in to read labels and symbols. To improve your viewing experience, the Graph View allows you to specify a default display mode for a graph configuration and change the display mode while viewing a graph.

To specify a default display mode used by a graph configuration:

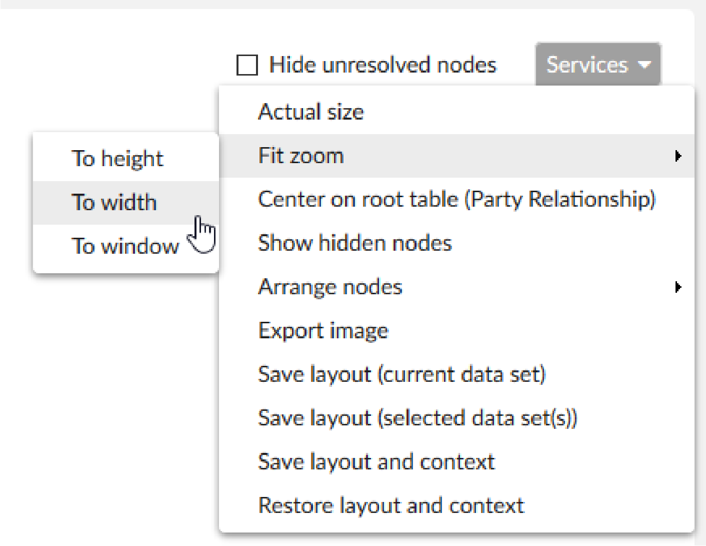
- Navigate to 'Administration' → 'User interface' → 'TIBCO EBX® Graph View Add-on configuration'.
- Determine whether you want to set this behavior in a template, or override an existing graph configuration and select the appropriate 'Graph', or 'Graph template' configuration.
- Locate the 'Default display mode' property and choose from:
  - Actual size (default value): Displays the nodes at their physical size.
  - Fit zoom to height: Adjusts the zoom level so that no nodes are located above or below the graph viewing area.
  - Fit zoom to width: Adjust the zoom level so that no nodes are located outside of the viewing area's horizontal axis.
  - Fit zoom to window: Adjusts the zoom level so that all nodes display in the viewing area.

To specify a display mode while viewing a graph, choose from the following options located under the 'Services' menu:

- 'Actual size': All nodes display at their physical size (default value). If the graph's default display mode is 'Fit to window', clicking 'Actual size' centers the selected node in the window and zooms in. This prevents you from losing your place while studying a large graph and having to zoom in/out.


□ Hide unresolved nodes
Services ▾
Actual size
Fit zoom
Center on root table (Party Relationship)
Show hidden nodes
Arrange nodes
Export image
Save layout (current data set)
Save layout (selected data set(s))
Save layout and context
Restore layout and context

- 'Fit zoom': The graph automatically zooms out so that all nodes display according to window height, width or both using the options 'To height', 'To width', or 'To window', respectively.



Changing default node orientation

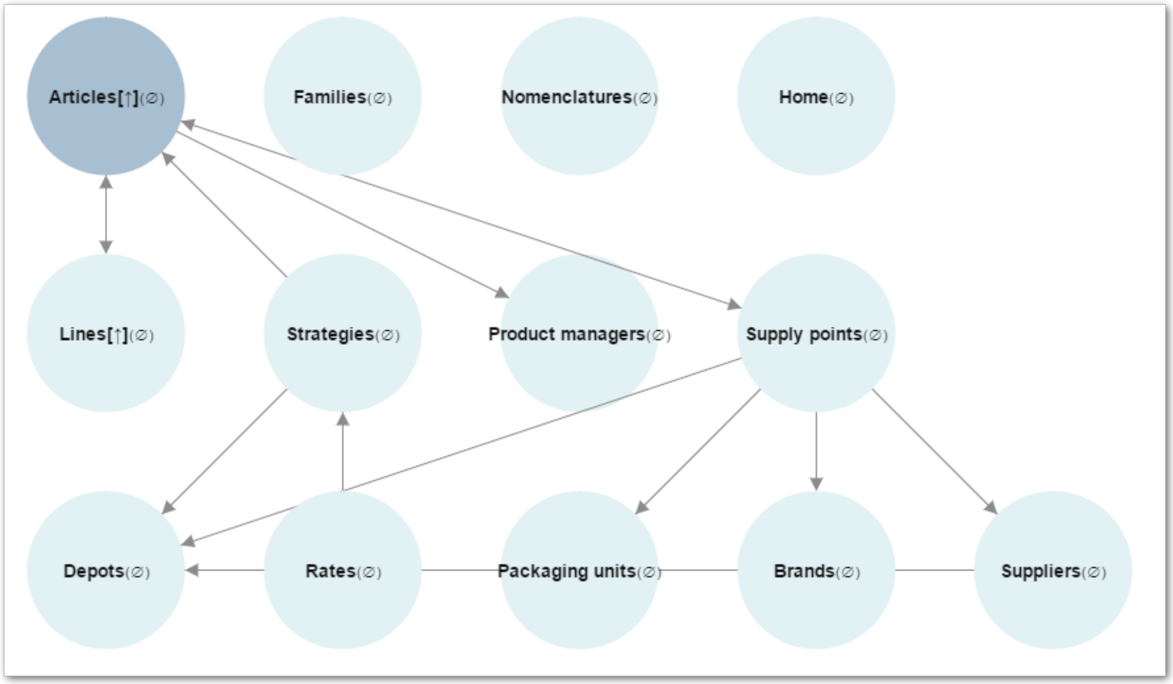
The Graph View optimizes node display by arranging nodes in a grid. This functionality applies to graphs showing data structure and graphs showing data values. You can use the following layouts:

- Hierarchy: You can only choose this option for graphs displaying data values. All nodes from the same table display in one row, or level. The uppermost level contains the records from the table you ran the 'Display graph data value' service, or the root table. When you display data values as a hierarchy, the 'Max. node (horizontally)' property no longer applies and some nodes may render off screen.

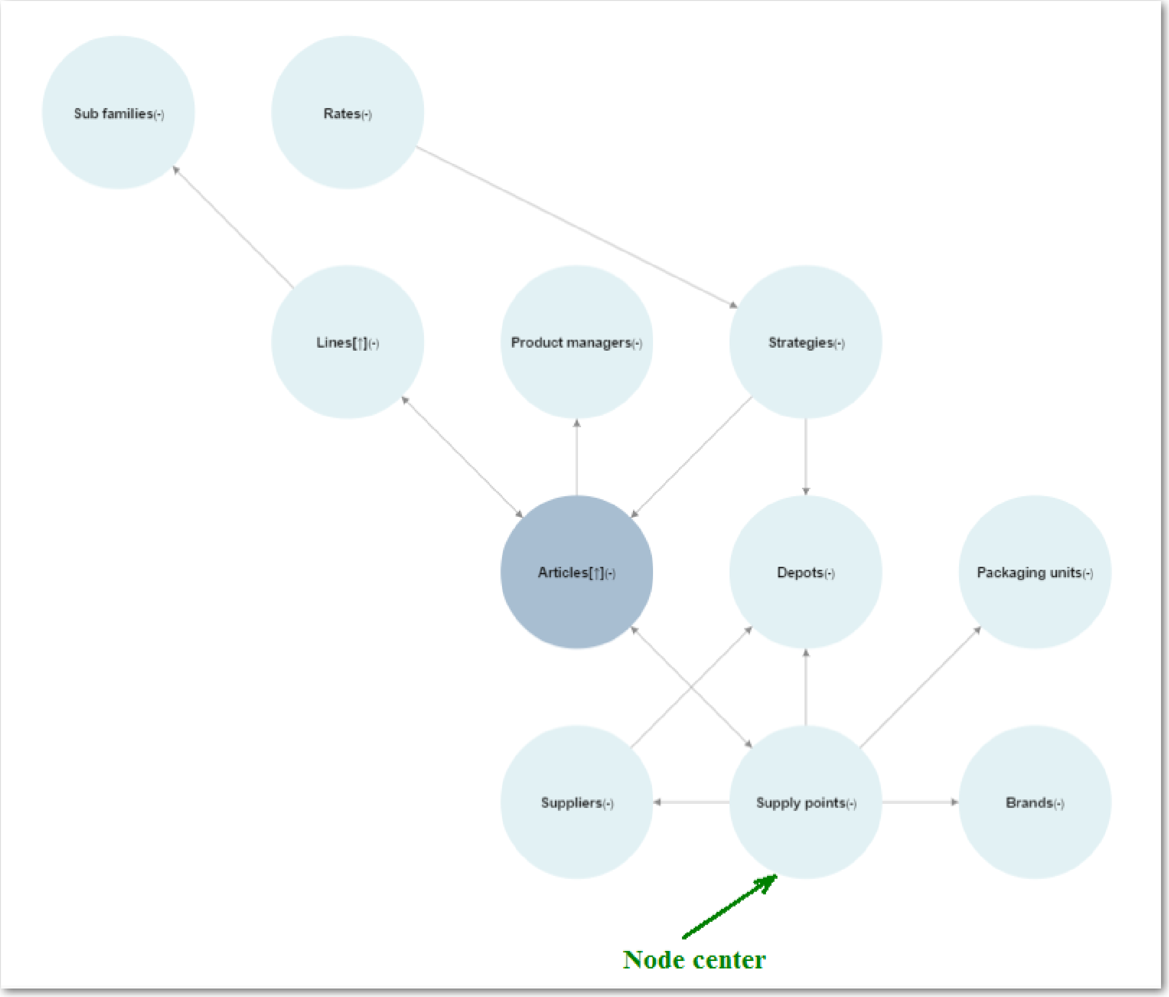
- Center aligned hierarchy: You can only choose this option for graphs that show data values. This orientation aligns nodes differently depending on your data structure. Generally speaking, this type of hierarchy arrangement displays a more horizontally oriented graph than a traditional vertical hierarchy.
- Square (default layout for data structure graphs): the node with the most relationships displays in the center. Subsequent levels in the grid—working outward from the center—contain related nodes. The first square around the center node contains nodes related to the node in the center. The next square out contains nodes related to nodes in the first square, and so on.
- Horizontal: the root node and any nodes without relationships display in the top row. With this type of orientation, graphs will generally be wider and shorter.
- Vertical: the root node and any nodes without relationships display in the top row. With this type of orientation, graphs will generally be taller and more narrow.
- Default grid: only available in graphs displaying data values. Each row contains data value nodes from the same table.
- Random display: only available in graph structures.

You can determine the preferred layout using default configuration settings, or by selecting an alternative option from a graph's 'Services' dropdown menu. See the 'Configure default grid display' section below for information on setting the default layout. Examples of each display option are shown in the following images.

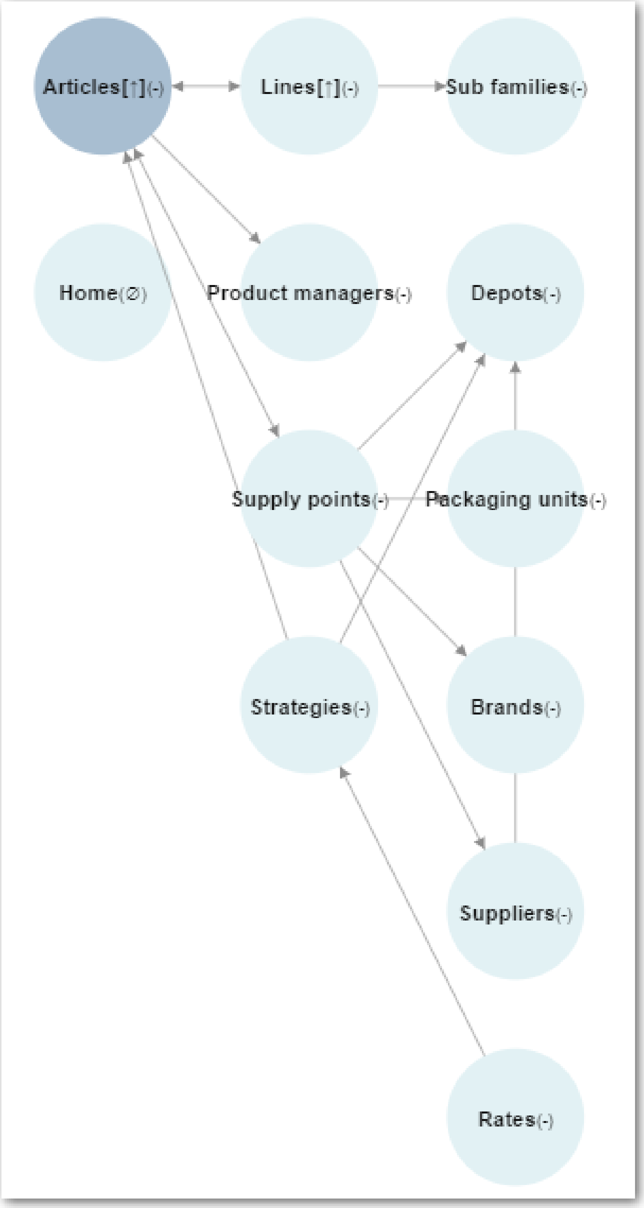
Horizontal display:



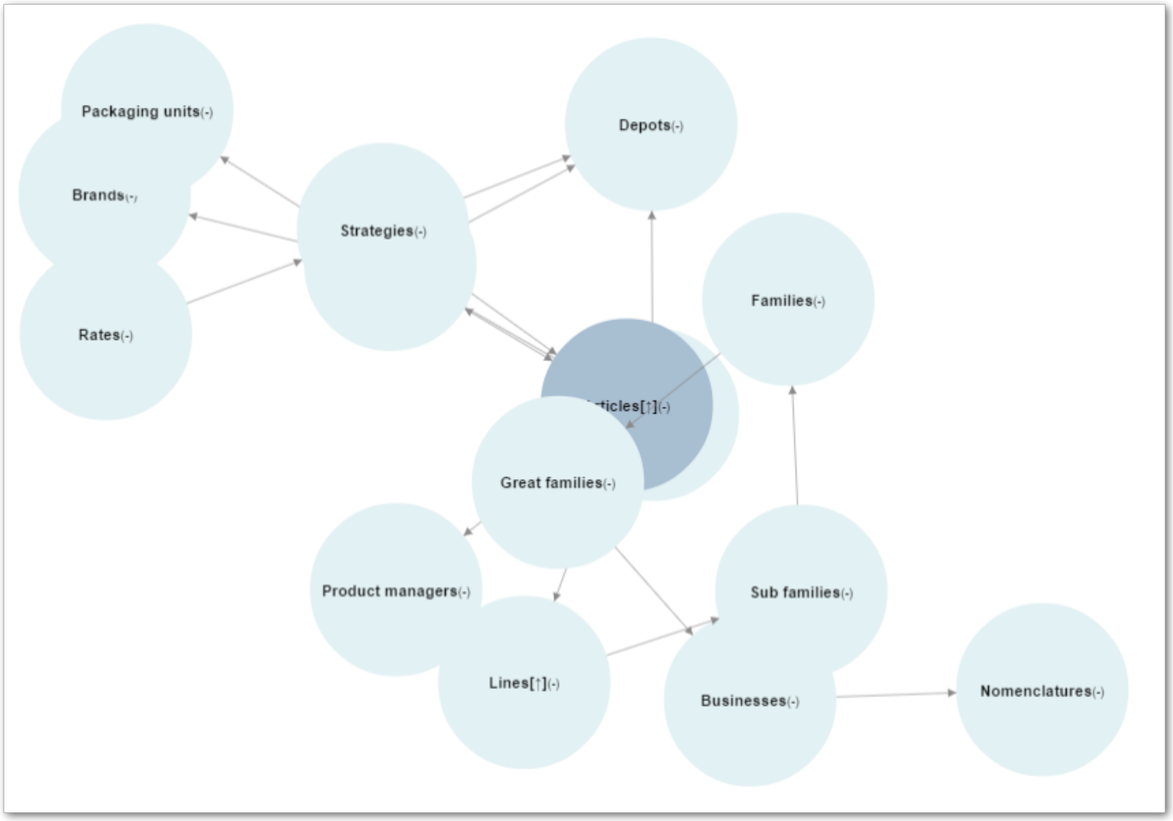
Square display:



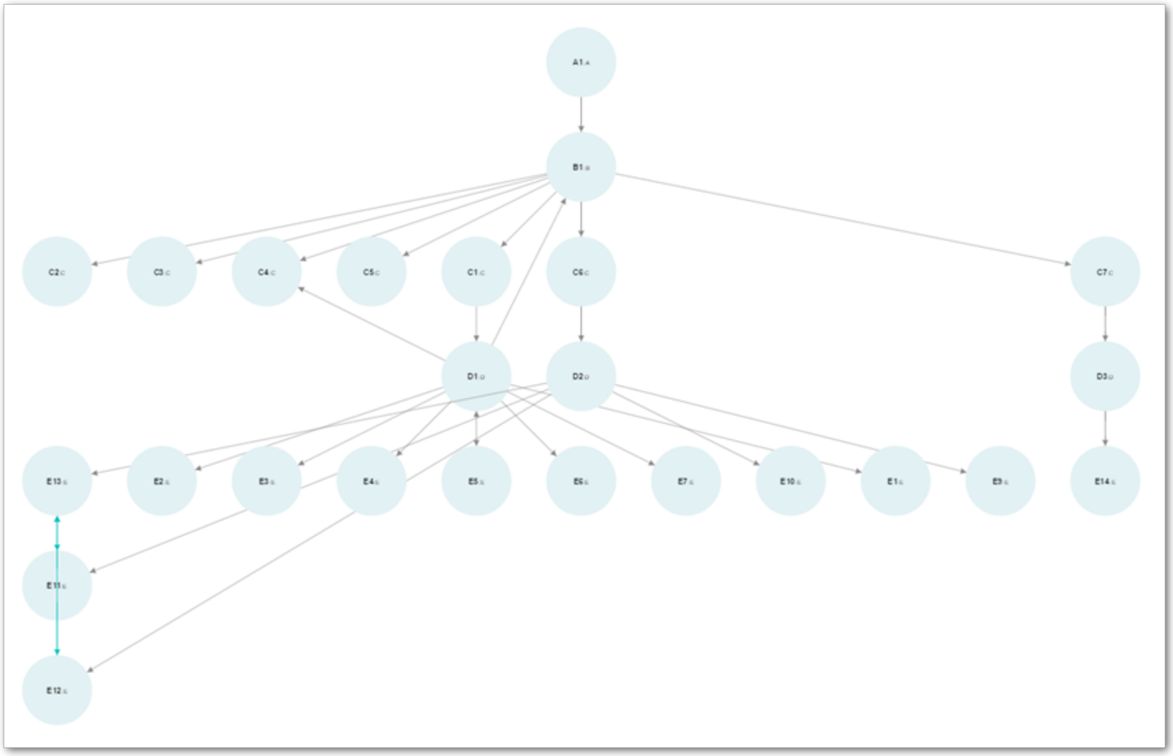
Vertical display:



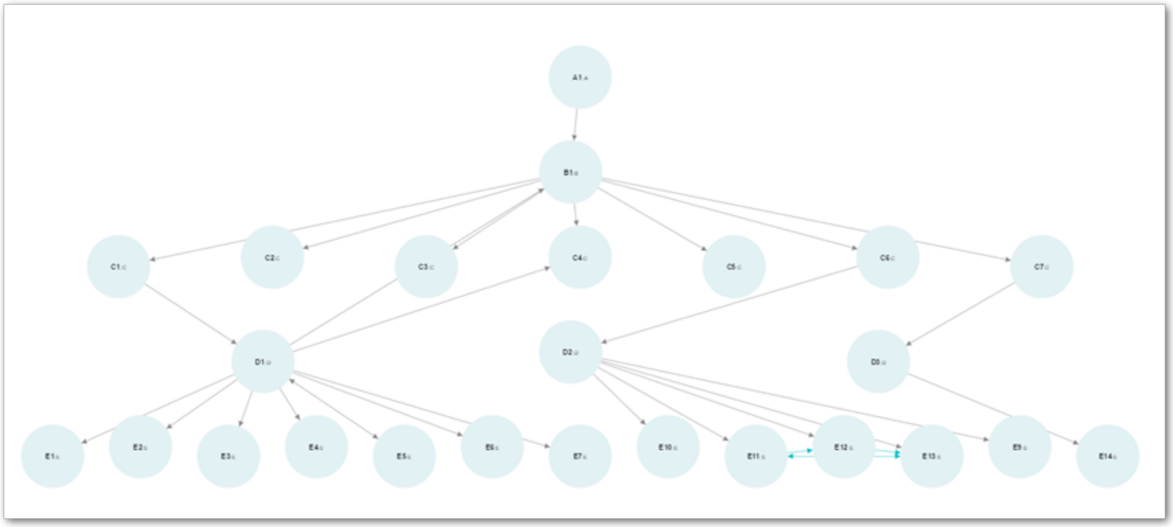
Random display:



Center aligned hierarchy:



In a hierarchy:



Configuring default grid display

By default, grid orientation is square for data structure graphs, and center aligned hierarchy for data value graphs. The following steps show how to change default grid layout:

- Navigate to: 'Administration' tab → 'TIBCO EBX® Graph View Add-on configuration' domain → 'Graph configuration' → 'Graph', or 'Graph template'.
- Open the graph configuration that you want to modify.
- You can set default display for the following:
  - Graphs showing your data structure: On the home tab, locate the 'Default grid orientation' property and select one of the available options: 'Horizontal', 'Square', 'Vertical', or 'Random display'.
  - Graphs showing data values: On the 'Data value graph' tab, locate the 'Default node orientation', 'Default recursive node orientation' or 'Default custom graph node orientation' properties and select one of the available options: 'Horizontal', 'Square', 'Vertical', 'Center aligned hierarchy', 'Hierarchy' or 'Default grid'.
- Save and close to finish your modification.

CHAPTER 4

Graph view reference material

This chapter contains the following topics:

4.1 Overview

This section lists and describes properties and fields in the following UI areas :

- Graph View.
- 'Graph configuration' domain.
- 'Reference data' domain.
- 'Preference' domain.

4.2 Graph View

The following UI areas are described in this section:

- 'Services' menu.
- Left-click on a node.
- Right-click on a node.
- Node label symbols.

'Services' menu

The 'Services' drop-down menu is located at the top-right of your screen when in the Graph View. The following menu options are available:

Menu Option	Definition
Actual size	Returns the zoom amount to the default level.
Arrange nodes horizontally	Displays nodes in rows and columns that follow a horizontal orientation. Thus, the graph displays wider rather than taller.
Arrange nodes as a square	Displays nodes so that they orient squarely in rows and columns around a center node.
Arrange nodes vertically	Displays nodes so that they are in rows and columns that follow a vertical orientation. Thus, the graph displays taller rather than wider.
In a center aligned hierarchy	Displays nodes according to your data structure. Generally, this type of hierarchy will be oriented more horizontally than vertically.
Arrange nodes in a hierarchy	Displays nodes organized in parent and child levels.
Arrange nodes in the default orientation (In graph of data values)	Displays nodes in rows and columns. Each row contains nodes from the same table.
Arrange nodes in the default orientation (In graph of recursive hierarchy)	Displays nodes in rows and columns. Root nodes display in the first column, and each row contains nodes from the same connected table.
Fit zoom to height	Adjusts the graph to fit the height of the window.
Fit zoom to width	Adjusts the graph to fit the width of the window.
Fit zoom to window	Adjusts the zoom level to display the entire graph.
Center on root table	Centers the root node in your screen. The root node name displays in parenthesis.
Show hidden nodes	Displays all nodes that have been hidden in this view.
Export image	Exports an image of the current Graph View to a new browser window. You can print the image or save it for future use. The exported image mirrors what you see on your screen. For example, if you are zoomed in on only a few nodes, they are all that display in the exported image.
Save layout (current data set)	Saves the currently displayed node layout that is applied for the current data set.

Menu Option	Definition
Save layout (selected data set(s))	Saves the currently displayed node layout that is applied to the current data set and other selected data set(s) that have the same data model.
Save layout and context	Saves the currently displayed node layout that is applied for the current data set and the data query context.
Restore layout and context	Restores the last saved layout and context.

Left-click on a node

The following menu options are available when you left-click a node:

Option	Definition
Display graph data value	Displays the selected node's records in a new graph view window. The new graph is laid-out as a grid. Each row displays the records from one table. The arrows show relationships and relationship direction between records.
Display recursive hierarchy	When a node contains recursive relationships you can select this option to display a hierarchical graph of data value nodes. The Graph View allows you to choose a relationship (if more than one exist) on which to base the hierarchy. Additionally, you can use a field to qualify the relationships. Values from the field you select display on the node transitions. You can execute this service on resolved and unresolved nodes.
Data query	Opens up the corresponding table where you can select records for a query. You can execute this service on resolved and unresolved nodes.
Open context	Opens a list of this nodes records that are part of the current context. This option is only available for nodes with more than one record.
Resolve from context	Resolves all nodes based on the current context of the selected node. This option is only available for nodes with a context. Note: This option can change data values in other nodes.
Continue data query	Initiates the resolution algorithm to resolve data values without changing the graph's current context context.
Display graph from current node	Acts as a paging mechanism when the threshold defined by the 'Max. number of displayed nodes' property is reached. This service is available when the (->) symbol appears on a node label and opens a new graph displaying related nodes. The node from which this service was run becomes the new graph's root node. If the number of related nodes still exceeds the specified limit, then you will be prompted to select the nodes you want to display. Note that you cannot access this service when viewing a graph based on a recursive hierarchy. Instead, you can use the 'Expand nodes' and 'Collapse nodes' services.
Expand nodes	Expands related nodes. The number of collapsed nodes displays in parenthesis below the node label. A graph configuration's 'Limit initial and expanded node display' restrict how many nodes display upon expansion. For example, the property may be set to '4' and there may be five collapsed, related nodes. In this case only four of the five nodes display when running this service. You can run this service again to expand the fifth node.
Collapse nodes	Collapses all nodes in a forward transition of the selected node and any nodes related to the forward nodes.
Unselect context	Deselects the current context and returns all nodes to an unresolved state. This option is only available for nodes that have a context and contain records.
Hide node	Hides the selected node. If you want hidden nodes to display, select 'Show hidden nodes' from the 'Graph services' menu. You can execute this service on resolved and unresolved nodes.
Actual size	Centers the selected node in the display window and returns the zoom level to the graph's physical size. You can execute this service on resolved and unresolved nodes.

Option	Definition
List of records (below the gray line)	Displays a list of records contained in this node. Selecting a record opens it in the tabular view. The length of this list is determined by the 'Number of max display records' property. If you see a '…' at the bottom of this list you can click on it to see the remaining records.

Right-click a node

Right-clicking a node acts a "quick-paging" mechanism and executes the 'Display graph from current node' service. For more information, see this service description in the 'Left-click on node' section.

Node label symbols

The following table describes the symbols displayed on a node label:

Symbol	Description
(-)	Node is unresolved.
(text)	When a node contains only one record, its name displays between parenthesis. For example, **Client('Peter')** shows that the 'Client' table has one resolved record named 'Peter'. Note: This may not be the total number of records contained in the table.
(a number)	When a node contains more than one record, the number displayed between the parenthesis indicates how many resolved records the table contains. For example, **Client(8)** shows that the 'Client' table has eight resolved records in the current context. Note: This may not be the total number of records contained in the table.
:	Indicates a forward resolution through the path indicated on the node label. For example, **Product('running shoes'): Order portfolio** shows that the 'Product' node resolution comes in a forward direction from its parent, 'Order portfolio'.
::	Indicates a backwards resolution through the path indicated on the node label. For example, **Price('running shoes'):: Product** shows that the 'Price' node is resolved in a reverse direction from its child, 'Product'.
(*)	Indicates an unknown number of records.
(0) with a line	The empty set symbol indicates a node has no records (in other words, the table is empty).
[↑]	When a node has a recursive relationship this symbol displays on its label.
(->)	Indicates the node has a relationship that isn't displayed. Using the 'Display graph from current node' service, allows you to view the hidden nodes.

4.3 **Graph configuration' domain**

The following describes:

- 'Table'.
- 'Graph' table.

- 'Graph template' table.

'Table'

When you select the 'Table', a table displays in the main screen. Each record in this table registers a table or a set of tables with the add-on. When you create a new record or update an existing record, the following properties are available:

Property	Definition
Code	A unique name without white spaces.
Name	Any naming convention can be used to define this property.
Dataspace, dataset: These properties specify the path or location of the tables you want to register with the add-on.	
Data space	The data space containing the data set where the desired table(s) is located.
Data set	The data set containing the table you want to register with the add-on.
Table	The table you want to register with the add-on. You can select an individual table for this record to meet a specific application, or you can select [All tables] to grant access to all of the tables in the data set.
Graph	The graph configurations are applied for current table.
Recursive hierarchy service refresh date	Displays the most recent date and time that the 'Update hierarchy view' service executed on this table configuration.

'Graph' table

When you select the 'Graph' table, a table displays in the main screen. Each record in this table stores configurations that determine Graph View user access, look/feel and display behavior. It is also in this record where you specify which table-previously registered with the add-on-the configuration settings apply to. When you create a new record or update an existing record, the following properties are available:

Property	Definition
Code	A unique name without white spaces.
Name	Any naming convention can be used to define this property.
Graph use	Currently the type of use 'Standard' is available only. In a further version, other types of graphs will be possible in order to get some specific services and behaviors.
Applied on table	This property determines which 'Table' configuration these record settings apply to. The drop-down list populates with all of the records contained in the 'Graph configuration's' 'Table' node.
User profile	Allows users to access the Graph View depending on their profile. Possible values for this property are either a single profile, or [all profiles]. You can grant or limit user access to the table specified in the 'Applied on table' property using this property.
Applied to data life cycle	Allows you to configure the graph for one or many data set(s).
Graph template	A drop-down list that populates with a list of graph templates. The list is derived from the 'Graph template' table where you can store customized options as templates for reuse. Note: All of the following properties inherit from the template specified in this property. You can override the inheritance by toggling the square icon. When the icon is dark gray, the property inherits its value from the template. When the icon is light gray, inheritance is overridden and you can change the property.
'Custom data value graph' tab: Defines a list of groups allowing you to specify a Java class and name the configuration.	
Java class	The custom class you want to use for Graph View display.
Label	The label, localized by language, used to identify the configuration.

'Graph template' table

When you select the 'Graph template' table, a table displays in the main screen. Each record in this table defines a graph template. Graph templates are referenced by the 'Graph template' property in 'Graph' table records. The values you specify in these properties are inherited by records in the 'Graph' table that customize Graph View look/feel and display behavior. When you create a new record or update an existing record in the 'Graph template' table, the following properties are editable:

Property	Definition
'Home' tab	
Code	A unique name without white spaces.
Name	Any naming convention can be used to define this property.
Max depth level after root	The maximum number of nodes levels that are displayed after the root table.
Background color	Specified the color of the UI background.
Text color	Specifies the color of the nodes' labels.
Resolved node layout	Specifies which node layout is used for resolved nodes. The drop-down list contains node layout configurations that have been created in the 'Reference data' domain's 'Node layout' table.
Transition highlight color	Specifies the color of the nodes' transitions for the last selected node.
Color for 'Continue query'	Determines the color used to indicate availability of the 'Continue data query' service.
Node label size	Determines the node label's font size.
Transition label size	Determines the transition label's font size.
Max record label size	This value determines the maximum length-in characters-of the displayed record.
Root node layout	This property determines which node layout settings are used to display root nodes. The drop-down list contains node layout configurations that have been created in the 'Reference data' domain's 'Node layout' table.
Node layout	This property determines which 'Node layout' settings this template uses. The drop-down list contains node layout configurations that have been created in the 'Reference data' domain's 'Node layout' table.
Node transition	This property determines which 'Node transition' settings this template uses. See Reference data domain.
Node distance	This property specifies the distance between nodes. See Reference data domain.
Is node moveable	A boolean value that determines whether or not node can be rearranged in the graph view. You may want to restrict this option if a graph view has been laid out in a specific way for some purpose. Otherwise, a user with permission might accidentally change the layout. If set to 'Yes': The nodes in the graph view that this template applies to can be moved. If set to 'No': The nodes in the graph view that this template applies to cannot be rearranged.
Display all nodes	If set to 'Yes': All tables (nodes) of the data model are displayed.

Property	Definition
	If set to 'No': Only the tables (nodes) with direct or indirect relationships from the root table are displayed
Zoom speed	This value specifies the rate at which the mouse wheel zooms in and out on the graph view.
Default grid orientation	Indicate the default orientation of grid. The available orientation are: 'Horizontal', 'Random display', 'Square' and 'Vertical'. The default value of this property is 'Square'.
Default display mode	Determines the display mode when the Graph View opens. You can specify that a graph displays all nodes at their actual size, or adjust the zoom level to fit all nodes in the window, or its height, or its width.
Max. Nb. of resolved records	This value specifies the maximum number of records that are queried before the incremental data query is stopped. A '*' on a node's label indicates that the query has stopped and resolution has not completed because the threshold specified by this property has been reached.
Max. Nb. of selectable records	This value specifies the maximum number of records that can be selected when querying a table. If the number of selected records exceeds this value, an error message displays.
Max. Nb. Of records displayed per node	This value specifies the maximum number of records that display in a node's pop-up menu. '...' indicates that there are more records beyond those displayed.
Zoom % for last selected node	Each time a node is queried the screen may change to show query results. When you return to the graph view it may be confusing-if there are many nodes-to remember which was the last node you were working with. This is why the last selected node displays larger than the others. This property determines the display percentage over regular nodes.
Max. Nb. of displayed nodes	This property restricts the number of table, or record nodes displayed in a graph. Reducing the number of displayed nodes can improve performance. If you are using Firefox version 3.6. this value should be set no higher than 50.
Limit initial and expanded node display	Restricts the number of nodes that display when a graph loads and the number of nodes that display when users expand nodes. This property only applies to graphs of recursive hierarchies. Additionally, this property can not be set higher than the 'Max. Nb. of displayed nodes' property.
Max. backward resolution depth	Determines the maximum number of transitions the resolution algorithm navigates to resolve node data values. This number defaults to '2'. However, you can limit it to '1' for large volumes of data.
Hide unresolved nodes by default	Determines whether the 'Hide unresolved nodes' check box is automatically selected.
Display 'Hide node' service	Determines whether users can access the 'Hide node' service.
'Data value graph' tab	
Access data value graphs	This property determines whether or not users who access the graph view using this template can view records and recursive relationship values using the 'Display graph data value' and 'Display recursive hierarchy' services.
Max. node (horizontally)	Limits the number of nodes that display on the grid's horizontal axis.

Property	Definition
Max. Nb. of levels	Limits the maximum number levels that can be paged through after the table that the 'Display graph data value' service was run from.
Node layout	Determines how records display. The drop-down list references configurations made in the 'Reference data' domain's 'Node layout' table.
Recursive transition	Determines the transition layout used for recursive nodes and using the 'Graph data values' service.
Default node orientation	Indicates the default grid orientation for data value nodes. You can choose from: 'Horizontal', 'Square', 'Hierarchy', 'Center aligned hierarchy', 'Vertical' and 'Default grid'. The property defaults to 'Center aligned hierarchy'.
Default recursive node orientation	Indicates the default grid orientation for data value nodes with recursive relationships. You can choose from: 'Horizontal', 'Square', 'Hierarchy', 'Center aligned hierarchy', 'Vertical' and 'Default grid'. The property defaults to 'Center aligned hierarchy'.
Default custom graph node orientation	Indicates the default grid orientation when viewing a custom graph of data value nodes. You can choose from: 'Horizontal', 'Square', 'Hierarchy', 'Center aligned hierarchy', 'Vertical' and 'Default grid'. The property defaults to 'Center aligned hierarchy'.
Default display mode	Determines the display mode when the Graph View displays data value nodes. You can specify that a graph displays all nodes at their actual size, or adjust the zoom level to fit all nodes in the window, or choose between its height and width.
Display record label first	Determines in what order the record label, and additional node information displays.

4.4 'Reference data' domain

The nodes in the 'Reference data' domain store information referenced by graph templates.

'Graph use' table

This table declares the possible uses of a graph. In the current version, only the use 'Standard' is available. In further version, depending on the use, the graph will offer different services, such as to handle the meta-data.

Property	Definition
Code	Code of the 'graph use'. It is not possible to create user-defined 'Graph use'.
Name	Name of the 'graph use'
Description	Description of the 'graph use'

'Table layout' table

This table allows you to manually create configurations that determine how an individual node displays. For example, you can use an image to represent a table and use other images to represent each table record. These configurations override those set in a graph configuration.

Property	Definition
Graph configuration	The 'Graph configuration' to which this table layout applies.
Data space	The data space where the data set containing the table to use for this configuration is located.
Data set	The data set containing the table for this configuration.
Table	The table that this configuration applies to.
Field for record image	If records in the specified table contain fields that link to images, logos, shapes etc., you can specify that field here. The field must be URI type. Each table record displays as its referenced image. Note: images can not be larger than the property specified in the 'Image configuration' table.
Node layout	The node layout you want this table to display as.

'Node layout' table

The 'Node layout' allows you to configure the frame (shape, image) and the size of the nodes.

Records in this table contain the following properties:

Property	Definition
Code	A unique name without white spaces.
Name	Any naming convention can be used to define this property.
Node frame	Drop-down list containing the list of shapes available for node frames. These are predefined in the 'Node frame' table and contain basic shapes, as well as, a frame for images. If the node frame is set to "Image layout", the 'Image' field will be available and allow you to configure the corresponding image.
Node size	Drop-down list containing the list of available size configurations for nodes. These values are predefined in the 'Node size' node. You can click '+ Create' to open a new page to create a custom node size.
Color	Defines the color for this configuration. You can chose a color from the color picker.
Image	This drop-down list displays the images available for this configuration. These images are contained in the 'Image' table.
Description	A description for this configuration.

'Node size' table

The 'Node size' allows you to configure the size of the nodes.

Records in this table contain the following properties:

Property	Definition
Code	A unique name without white spaces.
Name	Any naming convention can be used to define this property.
Dim (pixel)	The node's dimension in pixels.
Height (pixel)	The node's height in pixels.
Width (pixel)	The node's width in pixels.
Description	A description for this configuration.

'Node frame' table

Records in this table are predefined and determine node shape. These values are used in the 'Node layout' table. Possible values are:

- Circle
- Ellipse
- Rectangle
- Square
- Star
- Image layout

Property	Definition
Code	A unique name without white spaces.
Name	Any naming convention can be used to define this property.
Description	A description for this configuration.

'Node distance' table

This table allows you to create and store predefined measurements that determine the distance between nodes. You can apply these settings when viewing a Custom data value graph in the default orientation.

Property	Definition
Code	A unique name without white spaces.
Name	Any naming convention can be used to define this property.
Distance (pixel)	The distance, in pixels, between nodes.
Description	A description for this configuration.

'Node transition' table

Records in this table specify the transition shape and contain the following properties :

Property	Definition
Code	A unique name without white spaces.
Name	Any naming convention can be used to define this property.
Type	Determines the transition's shape. You can specify a line, or arrow.
Color	Color of the transition
Text color	Specifies the transition's text color.
Max. label length	Sets a limit on the number of characters that display on a transition's label.
Dim (pixels)	Dimension of the transition in pixels
Width (pixels)	Width of the transition in pixels
Description	A description for this configuration.

'Data life cycle context' table

Records in this table specify the 'Data life cycle context' used for the graph configuration.

Property	Definition
Data space	Data space where the data set is configured
Data set	Data set use as context

'Image configuration' table

Allows you to point the add-on to an upload location. This location is used to store images uploaded in the 'Image' table. These images can be used to represent tables.

Property	Definition
Code	Enter a unique code with no white spaces.
Name	Enter name for this configuration.
Upload folder	Path to the folder where you want to store images. By default, this path is `ebx_home/gram_img`. If you want to change this location, the `JAVA_OPTS` environment variable must be updated by adding the following parameter: `JAVA_OPTS="-Dgram.image.configuration.path=<path to folder location>"`. Note that you might need a system administrator to perform this change. EBX® needs to have read, write and execute access in this location. Without sufficient permissions the Graph View may not have the ability to upload, delete or display images correctly.
Max. upload size (KB)	Limits the uploaded image size.
Time out (milliseconds)	Limits the image's allowed load time. If this value is exceeded, a colored rectangle displays instead of the image.

'Image' table

This table is where you manage your image uploads. Use the 'Services' drop-down menu to upload and delete images. If the 'Upload' option does not display, you have to create an 'Image configuration' first.

Property	Definition
Name	Image name you would like to use in add-on configuration settings.
File name	Actual image file name.

4.5 'Preference' domain

The 'User graph data' table contains records that save node layout and context. The 'Transition's label' table stores configurations saved for hierarchy graphs. You can export and import this data when you want to share and reuse a certain layout.

User graph data' table

The 'User graph data' node contains records that save node layout and context when you save using the 'Save layout'/ 'Save layout and context" options from the 'Graph services' menu.

Property	Definition
Graph configuration	The configuration for the saved graph.
Root table	The table that user runs service "Graph view" on.
Data set	The specific data set that user selects to save layout.
User profile	Allows users to access current layout and context depending on their profile.
Scale	The scale (zooming) value of current graph.
Selected table	The last selected table that user queries data on.
Value	The selected records that user queries on.
Node layout: The position, display state of nodes in graph.	
X position	X position
Y position	Y position
Data space	Data space corresponds to a node.
Data set	Data set corresponds to a node.
Table	Table corresponds to a node.
Is hidden	Determines node visibility.

'Transition's label' table

To display a recursive hierarchy, you can define a context by choosing a relationship and, an optional, qualifying field. The add-on uses the qualifying field to display node transition labels. When you choose to save this context, this table stores the values.

Property	Definition
Graph configuration	The configuration applied to the table that displayed as a hierarchy.
User profile	The user who ran and saved this configuration.
Parent table	The table on which this configuration was used.
Selected foreign key	The relationship the hierarchy was based on.
Selected field	The field chosen to qualify the relationship.

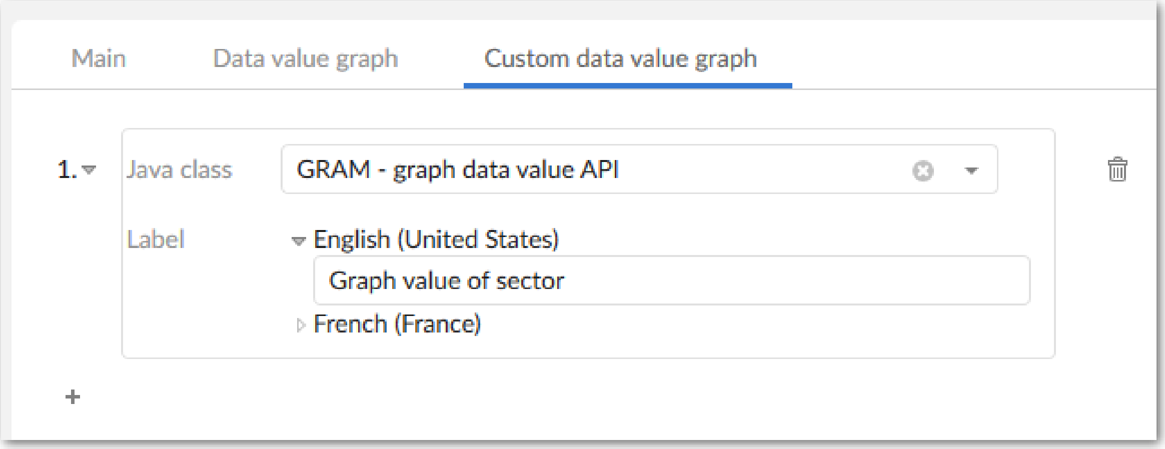
CHAPTER **5**

Creating a custom view

You can implement a customized graphical view of your data using the API. After creating a Java class, you can add it to a graph configuration in the 'Graph' table. Each graph configuration can have multiple Java classes. When you enter the Graph View, you can select one custom implementation to display your data. For more information, see the API documentation.

The following outlines the steps to take after creating your custom classes:

- Navigate to the 'Administration' → 'TIBCO EBX® Graph View Add-on configuration' → 'Graph configuration' → 'Graph' table.
- Open the graph configuration to which you want to add your custom classes and click on its 'Custom data value graph' tab.



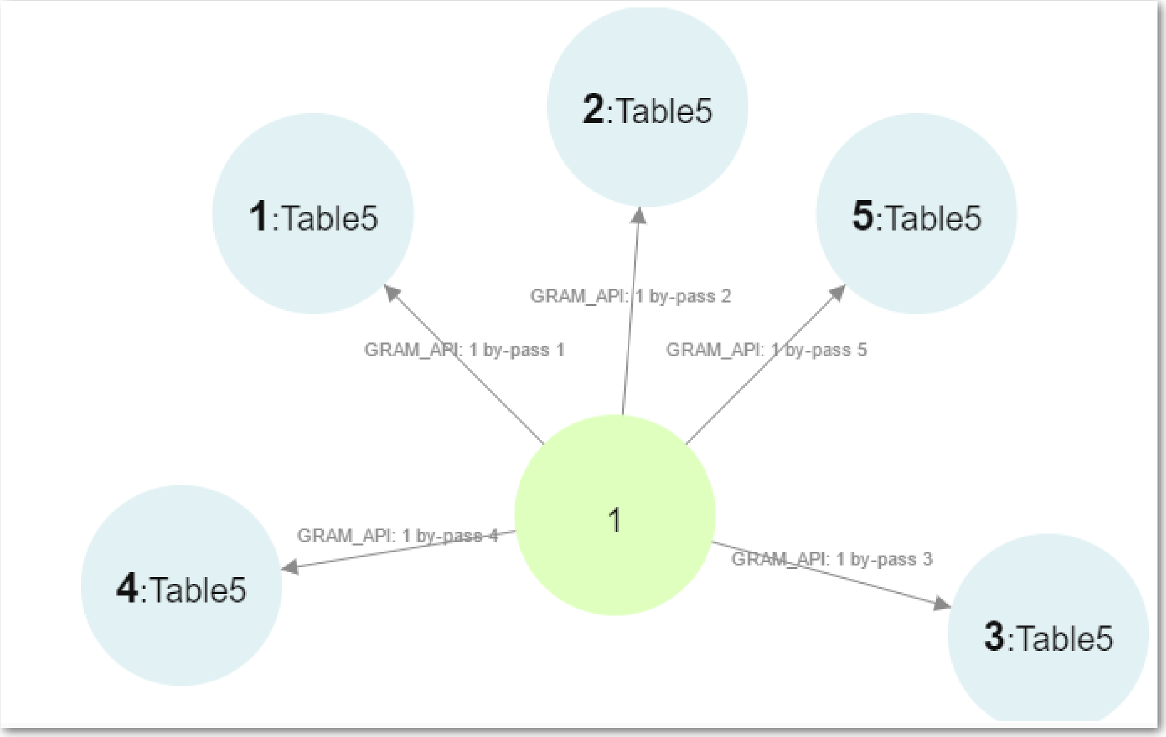
Java class: GRAM - graph data value API

Label: English (United States): Graph value of sector

French (France)

- Click the '+' icon to add a class and select the desired class from the 'Java class' drop-down list. In addition, you can add a label and define it in multiple languages. Repeat this step for any additional classes you want to add.

- Save and close. When you run the 'Graph View' service from a table the 'Custom data value graph' now displays as an option. After selecting this option, you can choose one of the classes you added and generate the graph.



CHAPTER **6**

Graph resolution algorithm

When you query data while in the Graph View, a resolution algorithm navigates the relationships between data to return the correct values for the query's context. The resolution algorithm follows relationships both forwards and backwards. Arrows connecting nodes indicate the relationship direction.

The algorithm gives forward resolution higher priority than backward resolution. When there are no more forward transitions to resolve, the algorithm follows reverse transitions. The following provides examples of each type of resolution using three nodes: A, B, and C. Both A and C hold foreign key relationships to B.

- Forward resolution: When you query node 'A', you have to select one or more records in node A that create a context. After you select, the algorithm follows the FK to table 'B' to return the values in table 'B' for the chosen context. When there are no more transitions to follow, the algorithm begins reverse resolution.
- Reverse resolution: From 'B', the algorithm resolves backwards to 'C' and retrieves the FK value that fulfills this context.

Resolution will not exceed the 'Max. Nb. of resolved records' property value. When the number of records resolved would be greater than this value, resolution stops and the node displays with an unknown number of related records (indicated by '*' in the node label).

CHAPTER 7

Known limitations

The known limitations are listed below:

- Recursive relation and multi-relations between nodes are not displayed.
- When two nodes have more than one relation, the resolution algorithm uses only one.
- The relations between data spaces and data sets are not displayed in the graph.

Release Notes

CHAPTER 8

Version 1.5.6

Released: April 2023

This chapter contains the following topics:

8.1 New features

This release contains no new features.

8.2 Changes in Functionality

In the user interface, the **Upload folder** field in the **Image configuration** table is now read-only. The default value is now `ebx_home/gram_img`. You can update this value using a Java environmental variable. For additional information, see *Graph view reference material* > *'Image configuration' table* *[p 58]* in the *User Guide*.

8.3 Changes to third-party libraries

This release contains no third-party library updates.

8.4 Closed issues

This release contains no closed issues.

8.5 Known issues

This release contains no known issues.

CHAPTER 9

All release notes

This chapter contains the following topics:

9.1 Version 1.5.6

Released: April 2023

New features

This release contains no new features.

Changes in Functionality

In the user interface, the **Upload folder** field in the **Image configuration** table is now read-only. The default value is now `ebx_home/gram_img`. You can update this value using a Java environmental variable. For additional information, see *Graph view reference material* > *'Image configuration' table* *[p 58]* in the *User Guide*.

Changes to third-party libraries

This release contains no third-party library updates.

Closed issues

This release contains no closed issues.

Known issues

This release contains no known issues.

9.2 Version 1.5.5

Released: December 2021

New features

This release contains no new features.

Changes in Functionality

This release contains no functionality changes.

Changes to third-party libraries

This release contains no third-party library updates.

Closed issues

This release contains the following closed issues:

- **[GRAM-1043]** Validate the **Image name** field inputs.
- **[GRAM-1045]** Validate the **Code** field inputs.

Known issues

This release contains no known issues.

9.3 Release Note 1.5.4

Release Date: September 18, 2020

Featured updates

- The add-on has been updated to support the OpenJDK8 and OpenJDK11 libraries.
- Some JAXB libraries were added to support JDK11.

9.4 Release Note 1.5.3

Release Date: June 23, 2020

Bug fixes

[GRAM-1026] An add-on description in French is not translated.

9.5 Release Note 1.5.2

Release Date: June 20, 2019

Featured update

The add-on has been updated to ensure compatibility with the TIBCO EBX® 5.9.4 release.

9.6 Release Note 1.5.1

Release Date: December 14, 2018

New features

This release contains the following updates:

- It is now possible to run the **Graph data value** service from a toolbar on the table row.
- The **Close** button is now only available if you run the **Graph data structure** service on perspective or indirectly from a workflow. Running the service directly from a workflow will not display the **Close** button.

9.7 Release Note 1.5.0

Release Date: October 26, 2018

Featured update

The EBX® Graph View Add-on has undergone significant updates to ensure compatibility with the EBX® 5.9.0 GA release.

9.8 Release Note 1.4.3

Release Date: May 19, 2017

New features

- The 'Open record' service can be hidden on a node in the custom graph API.

9.9 **Release Note 1.4.2**

Release Date: April 18, 2017

Bug fixes

- **[25497]** Unauthorized access to resources is possible.

Warnings

- **Customers are strongly advised to upgrade to the latest version which patches the security issue.**

9.10 **Release Note 1.4.1**

Release Date: March 31, 2017

Bug fixes

- **[25127]** Node labels do not display correctly in the recursive hierarchy graph.

9.11 **Release Note 1.4.0**

Release Date: March 20, 2017

New features

- A new option displays node labels in multiple lines.

9.12 **Release Note 1.3.0**

Release Date: January 23, 2017

New features

- An arrangement option to view nodes in a center aligned hierarchy has been added to graphs. This option only applies to graphs that display a hierarchical representation of nodes.
- You can now configure the number of displayed nodes when generating graphs and expanding nodes.
- The 'Display graph from current node' service was removed from graphs displaying a recursive hierarchy.
- It is now possible to configure the transition length for the custom graph API.
- The bidirectional data in 'Customize graph data value' is now subject to auto-correction.

Bug fixes

- **[22311]** The node label is incorrectly displayed when the total number of records exceeds the 'Max. record label size' value.
- **[22956]** Bidirectional transition in 'Custom graph data value' cannot be presented properly.

- **[23527]** Running the 'Custom graph data value' service displays a white page considering Java class undefined.
- **[23948]** Running the 'Display recursive hierarchy' service on a node that is not configured on 'Table' displays a white page.

9.13 **Release Note 1.2.3**

Release Date: August 4, 2016

New features

- Graph View performance has been improved by using the new SchemaNodeRelationships API from EBX® 5.7.1.
- The 'Hierarchy' arrangement option is now available on custom data value graphs. The default grid orientation for custom graphs can be configured.

9.14 **Release Note 1.2.2 fix 001**

Release Date: August 1, 2016

New features

- The 'Hierarchy' arrangement option is now available on custom data value graphs. The default grid orientation for custom graphs can be configured.

9.15 **Release Note 1.2.2**

Release Date: July 8, 2016

New features

- The 'Collapse/Expand linked nodes' services are renamed to 'Collapse/Expand child nodes' respectively. These services only affect nodes that they link to.

9.16 **Release Note 1.2.1**

Release Date: June 10, 2016

New features

- Child nodes can be collapsed, or expanded using a node's 'Collapse linked nodes' and 'Expand linked nodes' services, respectively.

Bug fixes

- **[20726]** The progress bar freezes when running the 'Display graph from current node' service from a data value graph.

9.17 Release Note 1.2.0

Release Date: May 19, 2016

New features

- The following services and options display graph nodes as a hierarchy:
  - The 'Display recursive hierarchy' service is available to display nodes as a hierarchy when a recursive relationship, or simple join table exist in your data structure. This service is available at the graph and table levels. On the table level, if no record is selected, the full hierarchy of graph will display.
  - The 'Hierarchy' option is a selection that is available when choosing a default node arrangement for a data value graph. You can also set this to be the default arrangement for a graph configuration.
- When a node has the 'Continue data query' service available, it is highlighted by a specific border color.
- The 'Hide unresolved nodes' checkbox is available to hide all nodes that remain unresolved after running a query. This can be set as the default behavior in a graph configuration.
- Options are now available when viewing a graph, and in its default configuration, to set the zoom level to actual size or one of the following options: Fit to window, Fit to width, or Fit to height.
- The 'Display the selected data value' service is now available to open a data value graph based on the table record(s) you select.
- The graph resolution algorithm is improved and allows you set the maximum number of backward steps to '1' or '2'.
- The API is exposed to allow you to create a custom implementation for data value graph display.

9.18 Release Note 1.1.3

Release Date: February 4, 2016

Bug fixes

- **[19254]** The 'Save layout' service under the 'Services' menu is missing a character when the language is set to French.
- **[19259]** The 'Graph View - Data path selection' screen title is not translated into French.
- **[19260]** The names and help contexts of the 'Upload image(s)', 'Delete images' services are not translated into French.
- **[19266]** The error message is not translated into French when a graph configuration uses nonexistent reference data.

9.19 Release Note 1.1.2

Release Date: August 24, 2015

New feature

- Previously, nodes displayed randomly the first time you opened a graph. This could be problematic and difficult to read due to the many intersections between node transitions. Now, the graph view automatically arranges nodes into rows and columns that make up a grid. The default layout is square in nature. However, you can specify two additional orientations - horizontal and vertical.

Bug fixes

- **[16998]** Result of 'Saving layout for selected data set(s)' form is inconsistent with EBX®.
- **[17003]** Color of the selected service from the list of services on node is inconsistent with EBX®.

9.20 **Release Note 1.1.1**

Release Date: June 10, 2015

Bug fixes

- **[16258]** Foreign keys are missing from a recursive node's submenu when they are in a group.
- **[16288]** A JavaScript error occurs after adding a Context containing cross script on an Item's 'Definition' tab.

9.21 **Release Note 1.1.0**

Release Date: April 9, 2015

New updates

- The 'Graph view' can now display table records graphically. The 'Display graph data value' service - available on a resolved table node - opens a new graph and displays the table's records and their relationships.
- You can improve graph performance using the property - 'Max. number of displayed nodes'. This property restricts the number of table, or record nodes displayed in a graph. Reducing the number of displayed nodes can improve performance.
- The 'Graph view' now has the ability to display and resolve recursive relationships. The [↑] symbol on a node's label indicates a recursive relationship.
- A new 'Continue data query' service allows you to resume a query from a resolved node without changing graph's context.
- Images can now be used in addition to the default shapes provided with the add-on to represent graph nodes.
- You can now manually configure each graph node's layout.
- New services are:
    - Display graph data value
    - Display the graph's recursive nodes
    - Continue data query

    - Display graph from current node

9.22 Release Note 1.0.0

Release Date: September 12, 2014

Features

The EBX® Graph View Add-on gives you a global view of your data models and data hierarchy.

- You can query tables from the graph representation. This allows you to quickly understand the relationships between tables.
- You can configure any graph view to adapt node layout transitions, colors and styles.
- Services are applied to a node are:
    - Data query
    - Resolve from context
    - Open context
    - Unselect context
    - Hide node
- Services are available at the 'Graph view' level:
    - Actual size
    - Center on root table
    - Show hidden nodes
    - Export image
    - Save layout (current data set)
    - Save layout (Selected data set(s))
    - Save layout and context
    - Restore layout and context