



Information Engineering Technology

# Rapide Release Notes



Release 4.0.0

# Contents

<b>Updated Pre-requisites and 3<sup>rd</sup> party software</b>	<b>5</b>
Java Runtime.....	5
RAP Version .....	5
<b>New Functionality</b>	<b>5</b>
Explicit override of Gen Comms Type for Coop .....	5
Add specific support for TUXEDO coop flows .....	5
New Rapide API method to set Custom Theme on the fly .....	5
Performance improvements when same AB called many times .....	5
Method to Refresh Specific Window/DialogBox .....	5
Overlay HTML properly configurable for Radio Buttons.....	5
Performance improvement in Trees with Images .....	6
New API Props/Methods for PStep Generation Date, Time and Rapide Version .....	6
Block Mode Transformation Tab Sequence.....	6
Application-Designed Timeout Facility .....	6
Protected fields option to not show error markers.....	6
Option to continue with references to unpackaged procedure steps.....	6
Support Set/Get Enabled & Caption on Menultems .....	6
Restrict number of tab sessions in a client browser.....	6
App Timeout facility to auto-flush specific events .....	7
Custom XML template folder.....	7
Support override selection character for group-view drop-downs.....	7
New Function Library Method replaceCharsAll.....	7
Improved performance for image handling.....	7
GainFocus support.....	7
Rotated text literals .....	7
New Method listBoxGetColumnPosition .....	7
Support for RightMouseButtonDown Added .....	7
Diagnostics added for tracking hangs/crashes better .....	7
ZZ Edit Pattern and Zero entry not to flag as auto-changed.....	8
New Methods to iterate URL parameters on Browser .....	8
Support MAKE on Radio Buttons .....	8
UNITTEST any PStep without an Entry Point on Browser .....	8
Assembling for Unit Test.....	8
Limited Support for LeftMouseButtonDown on Fields .....	8

Fonts with Underline & Strikethrough supported on DESKTOP .....	9
GENRT for Gen 8.6 Java Runtimes Updated – WKS86200 .....	9
New RFL method setOrientation .....	9
Value of ALL for RTL_LANGUAGES .....	9
Tokens to override Gen trace HOST and PORT .....	9
Method to set an image for a listbox column header .....	9
New tokens to set listbox header background and foreground colours .....	9
Application-specific sections in rapide.properties.....	9
New method postEventToPStepWithCommand .....	9
New option on setNextFlowWindowStyle for NEXTTRAN into a Frame.....	9
New option to package server import/export classes into separate jar .....	10
New User Exits invoked before a Server Call, and after a Server Abend .....	10
New User Exits invoked when Client logic fails.....	10
Option to directly start the System Console.....	10
Methods to get and set a Pane’s position .....	10
Ability to switch between a RowTemplate and Standard ListBox at runtime .....	10
Option to disable/hide listbox context menu items .....	10
Windows GDI Handle check more intelligent .....	10
Application start-up performance improvement .....	11
New ExpandBar feature to disable collapse .....	11
Studio Developer UI Preview Custom CSS .....	11
CSE Custom Theme auto-prefix based on control name .....	11
Force zOrder to BACK based on Control Name .....	11
Load into Parent Frame option available as a default .....	11
Make popup dialogs have no title bar by default .....	12
Dynamically change CSS Custom Themes on the fly .....	12
Multiple CSS theme files .....	12
New Edge Tracking method using Percentage of designed position .....	12
CSE Custom Theme auto-prefix rename tokens and support MultiLine Fields .....	12
Support Permitted Values from a Group View in Static DropDown Overlays .....	12
Mindfusion JDiagram diagramming control support.....	13
Dynamic Menu enhancements .....	13
Support GetWidth/Height/Top/Left with no shell.....	13
Scaling from tokens overrides Exit.....	13
GC Garbage Collection frequency more controllable .....	13
Include server wrapper classes in <appl>_SVRIA jar .....	13
New RFL method to provide the time difference of user’s browser .....	13
Auto-Suggest drop-down improvements for browser.....	13

New RFL method to renameFile ..... 14

## Updated Pre-requisites and 3<sup>rd</sup> party software

---

### Java Runtime

---

Rapide 4.0 requires Java 17 for the runtime environment for browser and desktop applications. Please consult the installation guide for further details.

### RAP Version

---

Rapide 4.0 uses the 3.26 release of the Eclipse RAP 3<sup>rd</sup> party software: .

## New Functionality

---

The following features have been added since Release 3.0.0. Note that most of the new features were released as service packs for Release 3.0.

### Explicit override of Gen Comms Type for Coop

---

A new token COMMS\_TYPE may be specified in rapide.properties to explicitly override the Gen comms type classes to use for Co-Operative calls to servers. Default is TCPIP, but other values can be ECI, EJBRMI, ITPIP, J2CSRmi, J2CS, MQS, TUX and WS.

### Add specific support for TUXEDO coop flows

---

Enhance Rapide runtime to support Tuxedo co-operative flows. Regenerate of GUI XML for clients will be required if using overrides for server pstep Tuxedo Service Names.

### New Rapide API method to set Custom Theme on the fly

---

Added method setControlCustomTheme to Rapide Function Library to allow BROWSER applications to change the theme of a specific control at run-time.

### Performance improvements when same AB called many times

---

Some performance improvements where a PStep calls the same Action Block multiple times during an execution of its' logic.

### Method to Refresh Specific Window/DialogBox

---

A new method called refresh is provided in the Rapide Function Library to allow requesting a refresh of the UI for a specific window or dialog box, which differs from the standard Gen REFRESH statement since that refreshes the UI for all windows and dialog box in the PStep. A special value of "<TOP>" may be passed to the new API method to have the top-most shell's UI refreshed (the shell must be a window or dialog box belonging to the current PStep, otherwise the request is ignored).

### Overlay HTML properly configurable for Radio Buttons

---

A new rapide.properties token, RADIO\_OVERFLOW\_CLIP, controls whether the HTML Overflow property is applied to radio button controls to enable the choice between clipping if the radio's design size is too small to contain the full text of its' prompt. Default is False, which means that the text will NOT be clipped.

## **Performance improvement in Trees with Images**

---

Significant performance improvements have been made for DESKTOP where Tree controls have images in subsequent columns.

## **New API Props/Methods for PStep Generation Date, Time and Rapide Version**

---

New Rapide API Properties are available to retrieve the generation date, time and Rapide generator Version for the calling PStep instance: `generationDate`; `generationTime`; `generationRapideVersion`. In addition, methods are provided to return the same date, but passing in the Member Name of any PStep in the application.

## **Block Mode Transformation Tab Sequence**

---

The first field with the 'place cursor here' property defined will have its GUI sequence defined to be the first field in the tab sequence. Remaining fields will be tabbed in sequence apart from the PF Key area which will have the lowest tab sequence.

## **Application-Designed Timeout Facility**

---

If an application is inactive for a period, it may be useful to support a timeout. This is already possible on BROWSER with appserver session timeout and RAP's native session timeout support. However, this gives little control in how the application times out and what happens at that point. Therefore, new functionality has been added which allows the application to register a timeout event which is triggered when reactivating after a period of inactivity. See the Rapide API documentation for further details, in particular, new methods `registerAppTimeoutEvent` and `rapide.properties` token `RAPIDE_APPLICATION_TIMEOUT`.

## **Protected fields option to not show error markers**

---

When a field has an error in its' input, such as not adhering to the attribute's permitted values, an error marker is tagged to the field and an appropriate tooltip is attached to indicate what the problem is. If the field is disabled/protected (not the same as read-only), the error marker is still displayed. A new option controlled by `rapide.properties` token `PROTECTED_CONTROLS_NEVER_SHOW_ERROR_MARKERS` can be set to True which will suppress these error markers for protected/disabled controls.

## **Option to continue with references to unpackaged procedure steps**

---

Normally it is desirable that Rapide generate/build fails early on if a pstep is being generated which references an unpackaged pstep (via flow or pstepUSE). However, in the case of testing, there may be situations where you want to ignore this failure and have the ability to build, assemble and test small portions of an application, even where some packaging is missing (if you are not going to test those areas). To support this, an environment variable can be set to instruct Rapide to skip such issues, `RAP_IGNORE_UNPKG=YES`.

The Rapide runtime accommodates this possibility, and any flows/calls to unpackaged psteps are trapped, with suitable error messages presented to the user.

## **Support Set/Get Enabled & Caption on MenuItems**

---

Support included for Set and Gen Enabled & Set and Gen Caption on MenuItems.

## **Restrict number of tab sessions in a client browser**

---

A new `rapide.properties` token, `MAX_SESSIONS_PER_CLIENT`, allows you to restrict the maximum number of tabs that can contain sessions pointing to the same Rapide application. This is because it has been found that different browsers have different limitations on resources and too many sessions can cause the browser to freeze, which also knocks on to the AppServer, thus affecting other users on different clients. In the GuardIEn

WebCR application, as an example, we found that IE11 could handle around 42 session tabs, Chrome and FireFox could only handle 6. Default limit remains unrestricted.

---

## **App Timeout facility to auto-flush specific events**

The Application Timeout facility has been further enhanced to allow automatic flushing (ignoring) of specific types of events in the event queue when a timeout occurs. For example, if the timeout is detected at the point where a flow is triggered, you probably want to process the timeout and NOT the flow, so this feature would allow you to throw away FLOW events in this situation. The feature is enabled with new rapide.properties token APP\_TIMEOUT\_FLUSH\_EVENT\_TYPES.

---

## **Custom XML template folder**

A new CustomXML folder is provided beneath the main Rapide software folder where you may copy and customise any of the template XML skeleton files, and if they exist there, these will be used in preference to the default versions of the XML files.

---

## **Support override selection character for group-view drop-downs**

Previously, it has been possible to define an override selection character for normal listboxes, so that a character other than '\*' is used in the group view data to identify the selected row(s). This feature is also now available for group-view-based Drop-Down controls too.

---

## **New Function Library Method replaceCharsAll**

Provided a new method in the Rapide Function Library to perform a 'replace all' where all occurrences of a string of characters will be replaced with another string.

---

## **Improved performance for image handling**

Startup performance of the application and also individual PSteps which contain images has been improved.

---

## **GainFocus support**

Previously, Rapide runtime would ignore GainFocus events. These are now supported on single and multi-line fields, although they always run in foreground so you cannot interact with the UI during the processing of the triggered event.

---

## **Rotated text literals**

A new property is available for text literals to rotate by 90, 180 or 270°. Supported only on DESKTOP targets.

---

## **New Method listBoxGetColumnPosition**

A new method is available in the Rapide Function Library, listBoxGetColumnPosition, to retrieve the current position of a column.

---

## **Support for RightMouseDown Added**

Added limited support for RightMouseDown for Listboxes and Single Line Fields. Note that this works well in DESKTOP, but should probably be avoided in BROWSER due to round-trip performance (same as most mouse-intensive events).

---

## **Diagnostics added for tracking hangs/crashes better**

New tokens are available in rapide.properties (MUTEX\_DANGER\_TIME and MUTEX\_DANGER\_GIVEUP) which can be enabled with IET Support's guidance to help in diagnosing occasional hangs/crashes in an application.

## **ZZ Edit Pattern and Zero entry not to flag as auto-changed**

---

If a field has a 'Z'-only edit pattern e.g. ZZZ.ZZ and a value of zero "0" is entered by the user, this is normally flagged through Rapide's auto-correct logic and pops up a warning message box if such an auto-correct is applied (according to rapide.properties tokens). This specific case where a "0" is reset to blank is now excluded from the pop-up message.

## **New Methods to iterate URL parameters on Browser**

---

Previously, the API method `urlGetParameter` was available to retrieve the value of a specific parameter as passed on a browser's URL e.g. "...?X=1&Y=2". This required you to know what the parameter names were in advance. Additional API methods are now available that allow you to iterate through all URL parameters retrieving names and values for each: `urlGetParmCount`; `urlGetParmNameAtIndex` and `urlGetParmValueAtIndex` (zero-based).

## **Support MAKE on Radio Buttons**

---

Rapide previously did not honour MAKE statements for Radio Buttons

## **UNITTEST any PStep without an Entry Point on Browser**

---

In a Rapide BROWSER application you normally have to define specific PSteps with Entry Points, and the application may only be started at those PSteps. However, for local/unit testing it may be useful to be able to go directly into a particular PStep that isn't normally an entry point. This is now possible by setting an environment variable `RAPIDE_ALLOW_UNITTEST=YES` (usually inside the `GdClientEnv.bat` scripts of Rapide), then reassembling the application. By doing this, an extra Entry Point called `UNITTEST` is included in the application that you may launch with a URL, specifying any PStep Member Name directly as the starting point, for example:

```
<appserver>:<port>/UNITTEST?APP=<appname>&PSTEPMBR=<mbrname>
```

The `UNITTEST` entry point will only be accessible if the application was assembled with the `RAPIDE_ALLOW_UNITTEST` option switched on at the time of the assemble, so you should ensure that this is not activated for your controlled test environments or production.

## **Assembling for Unit Test**

---

When assembling a Rapide application, the load modules have to be included in the application definition and all load module components generated and built before the application can be assembled. If you are performing local workstation testing on a subset, you may not have the complete application downloaded, have the necessary update access to allow the definition of the Rapide application, or you may simply want to quickly test a procedure step without wanting to generate, build and assemble the entire application.

To support unit testing, you can tick the *Assemble for Unit Test* checkbox. This then allows you to select individual load modules for an assemble into a test application called 'unittest'. You must have set the environment variable `RAPIDE_ALLOW_UNITTEST=YES` (see below) and be assembling for a BROWSER application.

When performing unit testing, you do not need to define entry points and can launch any procedure step.

## **Limited Support for LeftMouseDown on Fields**

---

Limited support has been added to the Rapide runtime for `LeftMouseDown` events defined on entry fields. These are not recommended though, especially when the field has other events defined.

## **Fonts with Underline & Strikethrough supported on DESKTOP**

---

Implemented support in SWT/DESKTOP targets (and STUDIO/ECLIPSE) for fonts with Underline and strikethrough styles. This is still not possible within BROWSER or MOBILE targets.

## **GENRT for Gen 8.6 Java Runtimes Updated – WKS86200**

---

The Gen 8.6 Java Runtimes shipped with Rapide (genrt.jar) have been updated to reflect changes in WKS86200, which include support for WildFly and JBossEAP as AppServers. This has been tested with Rapided connecting to Gen EJBs deployed within WildFly.

## **New RFL method setOrientation**

---

New method, setOrientation, for changing an individual control's RTL or LTR orientation at run-time.

## **Value of ALL for RTL\_LANGUAGES**

---

A new value of ALL or <ALL> can be set for token RTL\_LANGUAGES causing all locates to present as RTL orientation.

## **Tokens to override Gen trace HOST and PORT**

---

When AB/PStep logic is generated with Gen Trace, normally the Gen Java runtimes will attempt to connect to a Gen Diagram Trace Utility at localhost:4567. New tokens, GEN\_TRACE\_HOST and GEN\_TRACE\_PORT, are now provided in rapide.properties to allow you to override the DTU that will be used to connect to, making it easier (and more explicit) to connect to a different workstation than the one where the Rapide client logic is executing.

## **Method to set an image for a listbox column header**

---

A new RFL method is provided to allow you to set an image for a listbox column heading, listBoxSetColumnHeadingImage.

## **New tokens to set listbox header background and foreground colours**

---

Provided new rapide.properties tokens to set the background and foreground colours for all listbox headers in the application - LISTBOX\_HEADER\_BG\_COLOUR and LISTBOX\_HEADER\_FG\_COLOUR.

## **Application-specific sections in rapide.properties**

---

By placing tokens in the rapide.properties between markers, it is now possible for these tokens to only apply to the specified application name. For example:

```
<application name="testapp">  
TOKEN=VALUE  
TOKEN=VALUE  
</application>
```

## **New method postEventToPStepWithCommand**

---

A new RapideFunctionLibrary method is available, similar to the existing postEventToPStep, but allowing a COMMAND value to be sent on the triggered event too.

## **New option on setNextFlowWindowStyle for NEXTTRAN into a Frame**

---

If NEXTTRAN is detected, it is now possible to force the new PStep into a specific Frame as a child by calling setNextFlowWindowStyle beforehand to "CONTROL:<frameControlName>". This is designed primarily for Multi-Instance Tabs.

## **New option to package server import/export classes into separate jar**

For a co-operative server flow to be implemented, the server import and export view classes are generated and by default packaged into the client load module .jar. However, if a server is invoked by multiple clients, either directly, or via a used action block, all client load modules that include the server ia/oa classes need to be relinked. A new option is available to package the server ia/oa classes into a separate <application>\_SVRIA jar file so that they are only included in a single jar. This jar is re-built automatically when assembling the application. To enable this functionality, tick the 'Package Server IA/OA into separate JAR' checkbox in the Options dialog. Note that if you enable this function for existing applications, you will need to re-build existing client load modules to remove the server classes from their JARs. This option is not available when using a shared folder to build from multiple models. If you are using GuardIEn to build Rapide applications, the option is defined using the RAPSVRIA target property.

## **New User Exits invoked before a Server Call, and after a Server Abend**

New methods in UserExits are called before every server call (*determineIfServerCallAllowed*) and another exit whenever an error occurs during a server call - could be a comms failure or a server abend (*serverCallAbended*).

## **New User Exits invoked when Client logic fails**

A new method in UserExits is called when client AB logic fails (*clientLogicAbended*). Details of the calling pstep instance and the exception are provided, and you can decide whether to modify the displayed message, or even suppress the pop-up message altogether.

## **Option to directly start the System Console**

A new rapide.properties token, ALLOW\_SYS\_CONSOLE\_DIRECT=True, can be set which permits starting a browser application with a special URL/EntryPoint of SYS\_CONSOLE. This launches the System Console without the need to start a real session for the application and you can view all the other active sessions from there.

## **Methods to get and set a Pane's position**

New methods *getPanePosition* and *setPanePosition* are provided in the RapideFunctionLibrary allowing you to query the current position of a PStep's Pane, and to set it. Setting it will come into effect next time the PStep is started (not immediately). The position requested by *setPanePosition* is stored as a local preference so is remembered between sessions.

## **Ability to switch between a RowTemplate and Standard ListBox at runtime**

New RFL methods are available that allow a developer to switch the presentation of a ListBox between a RowTemplate and a standard ListBox (with column headings etc.). The method *toggleRowTemplate* is the primary method available for this, which may be complimented with methods *listboxGetRowHeight*, *listboxSetRowHeight* and *listboxStoreColumnsNow*. See the API documentation for further details.

## **Option to disable/hide listbox context menu items**

A new token DISABLE\_LISTBOX\_CONTEXT\_MENU is provided in rapide.properties so you can either hide the right-click menu altogether (value True) or you can concatenate codes to disable specific items in the context menu: R=Reset; O=OpenAsReport; D=Deselect; S=SelectAll; C=ChooseColumns.

## **Windows GDI Handle check more intelligent**

For Rapide applications running on Windows as SWT or as Workbench (Gen Studio), the check for GDI Handles reaching Windows system limits has been changed so rather than defining a threshold using rapide.properties token CHECK\_GDI\_HANDLES, this is now ignored and the actual relevant value for the limit per application is retrieved from the Windows Registry (HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows

NT\CurrentVersion\Windows\GDIProcessHandleQuota, and the threshold for warning set at 90% of that value instead). Note that there is a 64bit and 32bit (in SysWOW32 branch) copy of this property.

## **Application start-up performance improvement**

---

Improvement to Rapide application startup time - scanning for GUI PStep definitions in the application jars ignores jars that are named such that they cannot be Gen load modules i.e. they must be <=8 characters in length and contain only uppercase characters or digits. This may offer a noticeable improvement even in a small application.

## **New ExpandBar feature to disable collapse**

---

An ExpandBar control can now be defined as never collapsible. This will allow the style of that GroupBox to look the same as an ExpandBar but without actually being able to collapse it – useful for consistency of UI. This option is provided as a new value for the 'Start Contracted' property, of 'Disabled'. Support for BROWSER only at runtime.

## **Studio Developer UI Preview Custom CSS**

---

If the model rapide folder contains a custom css folder, when using the UI Preview feature of the Studio Developer UI Designer in Rapide mode, the custom CSS is now included in the preview.

## **CSE Custom Theme auto-prefix based on control name**

---

Rather than having to set a Custom Theme on all your controls in the UI Design, there are new rapide.properties tokens to automatically add a suffix to the Custom Theme at runtime based on the control name. Prefix Masks may be specified for most control types. New tokens are:

CUSTTHEME\_SUFFIX\_LIT (Literals)

CUSTTHEME\_SUFFIX\_FLD (Fields)

CUSTTHTME\_SUFFIX\_BUT (Buttons)

CUSTTHTME\_SUFFIX\_LST (ListBoxes/Trees)

CUSTTHTME\_SUFFIX\_RAD (Radio Buttons)

CUSTTHTME\_SUFFIX\_GRP (Group Boxes and ExpanBars)

CUSTTHTME\_SUFFIX\_CHK (CheckBoxes)

CUSTTHTME\_SUFFIX\_DRP (DropDowns)

Tokens may be repeated as many times as you need if multiple prefix masks are required.

For example:

CUSTTHTME\_SUFFIX\_FLD=fld:mytheme

...any Field control whose Name starts with 'fld' will have 'mytheme' appended to the Custom Theme applied.

## **Force zOrder to BACK based on Control Name**

---

If you want to force all controls whose control Name starts with a mask, use the new rapide.properties token ZORDER\_FORCE\_BACK\_PREFIX. This token can be repeated as many times as you need.

## **Load into Parent Frame option available as a default**

---

Some application designs have a parent Frame control high up in the hierarchy and use the 'Load into Parent Frame' property to have flowed-to psteps loaded into the same Frame. However, to get this to work you have to set every primary window to have the 'Load into Parent Frame' property switched on. A new token in rapide.properties is now provided to make this a default option so you do not have to set it for every window in the UI Design:

ALWAYS\_LOAD\_INTO\_PARENT\_FRAME=True

An additional value is available for this which also forces secondary Dialog Boxes to be loaded into the parent Frame, if there is one:

ALWAYS\_LOAD\_INTO\_PARENT\_FRAME=All

If you used these options, then the property defined on the individual window designs is not important.

---

## Make popup dialogs have no title bar by default

---

A pop-up window or dialog box may be presented with no title bar by setting the 'no title bar' property on the individual window in the UI Design. If you want this to be default behaviour, saving you having to set the property individually on every window / dialog, you may use the rapide.properties token:

2NDARY\_DLGBX\_DEFAULT\_NO\_TITLEBAR=True

This also applies to all window types, including Primary, not just Secondary Dialogs.

---

## Dynamically change CSS Custom Themes on the fly

---

Provided a new RFL property "SessionThemeSuffix" (GET and PUT) to set a Custom Theme Suffix to be added to the end of every control that has a theme defined. Allows you to switch between themes/styles programatically e.g. orange theme, green theme etc.

In addition, if you want this theme suffix to be added to ALL control types (not just those that already have a Custom Theme applied), a rapide.properties token SESSION\_THEME\_SUFFIX\_SCOPE may be set to ALL.

---

## Multiple CSS theme files

---

Technically not new, but now documented – it is possible to reference multiple CSS theme files so you can break down your customisations and place them in more conveniently sized CSS files. Place any such files alongside your modified rapideTheme.css in the \css folder of your model. They can be named however you like, but you must then modify the skeleton java file Configuration5.java in the Rapide software folder to list all of your css files, for example:

```
try {
    application.addStyleSheet(RWT.DEFAULT_THEME_ID, "css/rapideTheme.css");
    application.addStyleSheet(RWT.DEFAULT_THEME_ID, "css/rapideTheme_LT.css");
    application.addStyleSheet(RWT.DEFAULT_THEME_ID, "css/rapideTheme_LT_ONE.css");
    application.addStyleSheet(RWT.DEFAULT_THEME_ID, "css/rapideTheme_LT_TWO.css");
    application.addStyleSheet(RWT.DEFAULT_THEME_ID, "css/rapideTheme_LT_THREE.css");
} ...
```

---

## New Edge Tracking method using Percentage of designed position

---

Rapide runtime now supports the new Edge Tracking method of Percentage, which allows you to move each edge of a control (left, right, top and bottom) as a proportion of its' original designed position within the parent window. This means it is now possible to share space on the window proportionally between controls as the window grows/shrinks. Note that percentage tracking is only available against the <Window Edge>.

---

## CSE Custom Theme auto-prefix rename tokens and support MultiLine Fields

---

Tokens in rapide.properties for theme suffix were spelt incorrectly (was CUSTTHTME in some cases), renamed to CUSTTHEME. Also added support for Multi Line Fields using the CUSTTHEME\_SUFFIX\_FLD token, same as normal fields.

---

## Support Permitted Values from a Group View in Static DropDown Overlays

---

Previously, a listbox overlay could be enhanced as a Dynamic Permitted Value DropDown and Rapide could be instructed to get the list of PVs from a LOCAL Group View. This was NOT supported for static Overlays, but now is. Also support for explicitly listed PVs is provided, with corresponding documentation to explain the format required in the UI Designer.

## **Mindfusion JDiagram diagramming control support**

---

Initial support for Mindfusion's diagramming Java control has been implemented for WORKBENCH (Studio Developer) and SWT DESKTOP targets. This is intended initially for use by Studio Developer rather than for general consumption).

## **Dynamic Menu enhancements**

---

New RFL method (*dynamicTreeExpandUpOrLeftInstead*) to instruct a Dynamic Menu to popup to the left or above the menu starting position instead of right/below. This supports placing of the menu on the right or bottom of the UI design. Also fixed an issue in the Debug Console where the control type for Dynamic Menus was being displayed as 'Unknown Type' and any Custom Theme was not included.

## **Support GetWidth/Height/Top/Left with no shell**

---

Dot methods for getting Left, Top, Width and Height did not support a Window/DlgBox unless it was actually a physical Shell (window) at runtime. This meant it did not work for PSteps loaded into Frames or for Studio Editors/Views.

## **Scaling from tokens overrides Exit**

---

Scaling of X, Y and Fonts can be adjusted with a User Exit method already, but now matching tokens in `rapide.properties` will override anything returned by the Exit in preference. Tokens are:

`SCALE_FEATURE_FRONT_SIZE`

`SCALE_FEATURE_X_AXIS`

`SCALE_FEATURE_Y_AXIS`

## **GC Garbage Collection frequency more controllable**

---

A new token in `rapide.properties`, `GC_FREQUENCY`, can be used to determine how often a force/hint for the JVM to perform a GC (Garbage Collection) is done.

Value is in seconds. The check is done whenever a PStep is started, and will not occur more often than this number of seconds, but may occur less often. A value of zero is default, which means never request any additional GC - leave GC up to the JVM to manage (same as previous behaviour).

## **Include server wrapper classes in <appl>\_SVRIA jar**

---

A new option for Rapide Generation allows you to include the server 'wrapper' classes in the `<application>_SVRIA.jar`. These classes are needed when the servers are called using web services. When using the toolset plugin or GuardIEn CSE, the server managers will need to be generated separately with `java/ejb` as a target. When using Rapide from Studio Developer, the classes are generated when the 'Code' checkbox is selected for the 'server wrapper classes' row.

## **New RFL method to provide the time difference of user's browser**

---

Provided a new RFL method, *getClientLocalTimeOffset*, to return the time difference (in minutes) between the client's local time and UTC. Available for RAP/Browser only.

## **Auto-Suggest drop-down improvements for browser**

---

Changed events, and keyboard interaction with Auto-Suggest DropDowns did not work intuitively. `Ctrl+Space` did not work under Browser to force suggestion drop-down to appear.

## **New RFL method to renameFile**

---

Provided a new RFL method, *renameFile*, to attempt a simple file system rename of a specified file, to run on the machine where the Rapide application is running.