| **Name** | **Description** | **Type** | **User Group** | **Issue ID** |
| --- | --- | --- | --- | --- |
| APIs: Updated .NET version to 4.6.2 | The .NET versions of the Client API and the Extensions API have been updated to 4.6.2. If any of your inhouse developed applications or extensions use an older version, they will need to be recompiled with 4.6.2 or later. This is a breaking change in those cases. | Improvement | API | 16345 |
| Client API: .NET Standard - Remove dependency on WindowsBase | We have removed WindowsBase related libraries from the ClientAPI in order to provide support for .NET Standard 2.0 version on native linux. You can specify which authentication method to be used. As before, you can use SystemWeaver authentication. You can also use network credential authentication (Windows username/password) on all platforms. Network authentication (Single sign-on) works when running the .NET Standard version in Windows. | Improvement | API | 16343, 16288 |
| Client API: Ability to use Synchronization Context | There are now 3 different ways to synchronize threads with the addition of SynchronizationContext:   * None * Dispatcher * SynchronizationContext   Dispatcher does not exist in .NET standard. Instead, there is SynchronizationContext. SynchronizationContext needs a context to work correctly which a wpf application will provide. For console application, use None unless you have access to a valid context.  Example:  SystemWeaverAPI.SWConnection.Instance.Login(SystemWeaverAPI.EventSynchronization.None;  An error message is thrown if you try to use Dispatcher in .NET standard. | Improvement | API | 16323 |
| Client API: Fix ThreadSynchronization.None | We had some cases where EventSynchronization.None did not have the correct behaviour.  Now, with this option, no listener thread will be created. Events will be detected and distributed when you call the api, on the calling thread, if the call results in a call to the server. Forcing a call to the server can be done with the Ping method. | Improvement | API | 16350 |
| Client API: Ping | A Ping call has now been added to SWConnection.  You previously had to do some not so trivial code to use this call. We recommend that if you are using Ping using the old workaround, please use this new method instead. | Improvement | API |  |
| Clone item: Support for comments in configuration | It is now possible to add comments in the **Clone item** view configuration xml.  <!--This is a comment./-->  <Configs>  <Config id="Id1">  <!--This is a comment./-->  <ViewSettings>  <Caption>Letter</Caption>  <RibbonGroup>Templates</RibbonGroup>  <Description>Hint shown in the ribbon</Description>  </ViewSettings>  <TopItemType>7PRA</TopItemType>  ...  </Config>  </Configs> | Bug fix | Architect | 15063 |
| Configurable graph: Introduction of port styles | It is now possible to define and apply custom styles to ports (InPort, OutPort, and InOutPort) in configurable graphs. This is done by using a new <PortStyle> element. A style can include one or more of the following attributes:   * Fill color * Border thickness * Border color * Border type * Port symbol (It is possible to use Alt-codes to insert a port symbol.)   An optional Shape sub-element can be used to apply one of the following hard-coded Autosar symbols to a PortStyle:   |  |  | | --- | --- | |  | <ClientServerRPort/> | |  | <ClientServerPPort/> | |  | <ClientServerPPortFilled/> |   <PortStyles>  <PortStyle name="p1" fillColor="Red" borderColor="Blue" borderType="dash" portSymbol="⯇"/>  <PortStyle name="p2" fillColor="Green" borderThickness="3" borderColor="Yellow">  <Shape>  <ClientServerPPort/>  </Shape>  </PortStyle>  </PortStyles> | Improvement | Architect | 16215 |
| Configurable grid: Action support part multiplicity single | Using the **Edit** mode in a configured grid, it is now possible to add and create parts with multiplicity Single (support for multiplicity List was introduced in R39). | Improvement | Architect | 16211 |
| Configurable grid: Actions for Empty list and parts | Actions in Grid handle the case of an empty objectName.  <ActionBindings>  <ActionBinding actionId="MyAction" caption="My Action">  <WithParam name="item" objectName="testCase"/>  </ActionBinding>  </ActionBindings>  <Actions>  <Action id="clearStatus">  <Parameter name="item" as="[Item]"/>  <If test="$item=[]">  ... do something...  </If>  </Action>  </Actions> | Improvement | Architect | 16283 |
| Icons: New type icons | The following new Type icons have been added. They can be used as image icons for item, part, and issue types, as well as for multi-instance configurable views, and plugins: | Improvement | Architect | 16316 |
| ItemCommand extension: Hide inactive button | It is now possible for extension writers to control the visibility of an ItemCommand button on the ribbon using a new version of item command:  [Guid("E3B01297-DCA2-4006-B9CE-062C2497286D")]  public interface IswItemCommand2 : RemObjects.Hydra.CrossPlatform.IHYCrossPlatformInterface  The Update method on IswItemCommand2 now takes a visible parameter  public void Update(IswItem focusedItem, out bool enabled, ref bool visible)  The old IswItemCommand has been marked obsolete and will be removed in a later release. This is a breaking change. | Improvement | Architect | 15475 |
| Path query: Find deprecated status for type | Using the Path Query Language, it is now possible to check if an ItemType, PartType or IssueType is deprecated or not. The result is true or false.  **Example**  Type.IsDeprecated | Improvement | Architect | 16208 |
| Path query: Support for retrieving attribute information | It is now possible to use the Path Query Language to retrieve all of the attributes of an item, issue, part, or node (if they have values).  The following properties can be used together with the *Attributes* method:   * Attribute name * Attribute SID * Attribute type * Attribute value   **Example**  Attributes.Select(SID) | Improvement | Architect | 16334, 16373 |
| swExport/Import: Renaming of tools | The swExport and swImport 64-bit tools have been renamed.   * swExport is now named swXmlExport * swImport is now named swXmlImport | Improvement | Architect | 16212 |
| XML Editor: Updated style and improved functions | The XML editor used for scripting throughout SystemWeaver has been updated. You will find the same right-click functionality as before....  .... as well as an improved **Find** (Ctrl+F)feature, and the introduction of **Undo** (Ctrl+Z) and **Redo** (Ctrl+Y) options. | Improvement | Architect | 16325, 16367, 16182 |
| XML Import improvements | Additional work has been done in the area of data import. As a result, there is improved support for database synchronization between SystemWeaver servers. The improvements enable multi-database synchronization as well. The features listed here were made available for XML exports in release R39. With R40, they now apply to XML imports as well.   * To synchronize databases, you now set a global prefix for XML exports and imports. This new feature enables you to keep multiple databases in sync. It is called **Export Import Prefix**, and is set in the swArchitect client on the **Refactorings** tab. A prefix value is required. If you are already exchanging data with another database, **contact SystemWeaver before implementing an Export Import Prefix** for recommendations. This is a breaking change.  * There is a new XML import configuration tag:   <ExcludedParts/>   * Improvements to the standalone swExport application include:   + Supports the use of the same XML Export definition as the XML Export view in the swExplorer.   + Can read the XML Export configuration file. Entering options via command line is still supported.   + Checks and writes to a log file. Errors are written to the log as well.   + If there is any error, the swXmlExport tool will abort the export operation.   + Meta Data Conversion is available now on the export side as well. | Improvement | Architect | 16167, 16159, 16291 |
| XML Import: DB Sync(Preview) | There is a new option in XML Export called **Db Sync(Preview)**. This is a feature still *in development*, which is going to be the default way of keeping data in different SystemWeaver databases in sync. It is supposed to have a slightly different approach to versioning to minimize the impact of updates to released data, which can happen in multiple use cases including changes to the export/import configurations. | Improvement | Architect | 16147 |
| Configurable grid: Expanded editability | Users can now perform the following operations in a grid when configuration supports it. The menu options are configurable.   * Update existing parts​  <Part id="pAB" match="." defObj="A"/>​​ * Remove existing parts ​<Part id="pAB" match="." remove="true"/> <Part id="pAB" match="." remove="1 = 1"/>​ ​ * Set single obj refs from create issue dialog​ ​ * Set Issue Status ​  <Issue status="Done">​ ​ * Set Assigned to​ <Issue assignedTo="username">  <Issue assignedTo="#{?CurrentUser.Name}">​ * Set Iteration ​ ​ <Issue iteration="R40-A">​ ​ * Access Name from ItemData/IssueData <Action id="DoIt"> <Parameter name="p1" as="ItemData"/> <Item id="A" itemType="SC" name="--#{?$p1.Name}--"/> </Action> ​ ​ ​​​ * Set Attribute (part, item, issue, node)​ <Attribute sid="ASTR1" value="hello"/>​ * Use DialogCreateIssue to create and add an issue <DialogCreateIssue project="" issueType=""/>​ * Set IssueObjRef​ <IssueObjRef refType="IR" issue="IS" obj="IT"/>​ * IssueData Type   **Examples:**  Add an issue  Batch creation of issues based on a variable, e.g., for all Test Cases with given status, e.g., "Not Ok":    Result:  Batch creation of items  Create multiple items, e.g., Test cases, Executables, Interface mappings with one click. | Improvement | Architect, User | 16258, 16235 |
| Documents: Command line for document exports | A new tool called swExport can be used to export SystemWeaver documents in PDF format. Using this tool, you can implement your own application solution for document export based on your organization's needs.  **Examples:**  Using network authentication:  swExport ServerName=sys7 ServerPort=1345 ItemID=/x04000000000AC88A FileName="C:\temp\requirementsDocument.pdf  Using username and password:  swExport ServerName=sys7 ServerPort=1345 ServerUserName=admin ServerPassword=wagadoff ItemID=x04000000000AC88A FileName="C:\temp\requirementsDocument.pdf"  **Result:**  **Known limitations in this first version**  The export does not currently include content originating from:   * Extensions (e.g., Jira2, ComponentGraph, FaultTreeGraph) * GenerateGraph element * YellowGraph element   **Note:** The tool previously known as swExport has been renamed to swXmlExport to clarify its use as an XML export tool. | Improvement | Architect, User | 15337 |
| Structure Tree grid: Support for viewing part type name | It is now possible to show **Part type name** in the structure tree grid making it unnecessary to switch to the Parts view to view it there. | Improvement | Architect, User | 16217 |
| Monitor service: Maintenance and enhancements | The SystemWeaver Server Monitor Service now includes a web-based Status Tool that enables you to stop and start individual server applications in the Server Monitor Service without disrupting the other server applications that are running in the service. This is useful if you have multiple SystemWeaver "server applications" running on the same server machine, e.g., in a test/QA environment. This is a breaking change for the monitor service.  The Monitor Service Status Tool can only be used on the server machine where the monitor service is running. It's use is optional. See [Configuring the Server Monitor Service Status Tool](https://support.systemweaver.se/en/support/solutions/articles/31000164668-configuring-the-server-monitor-service-status-tool/) for more information. There is a new, **required** *wcfServicePort* attribute for the <SystemWeaverMonitor> tag, and a new, **required** *id* attribute for each <Server> tag. | Improvement | IT Administrator | 16356, 16091 |
| Server: Log query for long get items | For long QGetItems searches, the server will now log queries that take longer than 1 second to run:  Example: | Improvement | IT Administrator | 16351 |
| Rest API: Support for network authentication | There is now support for using Windows network authentication with the SystemWeaver Rest API.  Example  curl -X POST https://localhost:9000/token -d "username=student&password=test&windowsauthentication=true&grant\_type=password" | Improvement | Rest API | 16267 |
| Compare View: Not working with signature attribute | This fixes an issue where the view threw an error when a Signature type attribute was included. | Bug fix | User | 16347 |
| Configurable graph: Incorrect table allocation with grandchild nodes | This fixes a problem with allocation of nested grandchild nodes in a GraphTable. | Bug fix | User | 16186 |
| Configurable graph: Introduction of manual layout | It is now possible to manually change the auto-layout of a graph, as needed, to bring more clarity. The option is configured using a new <ManualLayout/> options tag to activate the Edit mode for users. This new feature enables users to:   * Modify the placement of nodes * Modify the size of nodes * Save manual changes to the layout * Set different layout by parameter * Reset manual layout to default layout   The manual layout of a graph for an item type is identified by an *id*. Support for applying a manual layout to a specific parameter is also available. See *Manual Layout* in the application Help.  Without a parameter  <Graph>  <Options>  <ManualLayout value="true" id="system2Blayout"/>  </Options>  .....  </Graph>  With a parameter  <Graph>  <Options>  <ManualLayout value="true" id="systemModel#{?$ShowRequirements}"/>  </Options>  .....  </Graph>  When the option is configured by an Architect, users can click the **Edit** button to access the manual layout functionality:  **Examples**  Drag and drop node to the desired location  Resize a node  **Relayout, Save layout and Load layout**  At any time, changes to the layout can be saved by clicking **Save layout**. Changes are saved and will remain in place for the selected item type. If **Save layout** is not clicked, the changes will not be saved.  If there is a saved manual layout, and you have made changes that you do not wish to save, you can revert back to a saved layout by clicking **Load layout**.  To revert to the auto-layout at any time, click **Relayout**. If you wish to save the auto-layout and overwrite any saved manual layout, click **Save layout**.  A saved manual layout is applied and seen by all system users with access to the item.  Given the new manual layout option, we have implemented incremental layout whereby layout calculation is performed dynamicly. For informational purposes only, the graph caches are stored at: %LocalAppData%\Systemite\SystemWeaver\3.0\0. This address is a generic one and works on every Windows computer.  **Disclaimer:** This is a first version of the manual layout functionality. Changes may follow in later versions. Should problems arise with loading a graph, we recommend using the **Relayout** option to return to the auto-layout starting point. | Improvement | User | 16259, 16336, 14999 |
| Configurable graph: Neighborhood not showing all bus routing | This fixes an issue where the Neighborhood pane was not showing bus routing to all nodes in the main graph. | Bug fix | User | 16083 |
| Sequence Diagram: Improvements to Message label | Message names now have a white background making them easier to read on the edge.  In addition, long Message names will be truncated in order to fit between two lifelines. The complete name can be displayed using the hover tip. | Improvement | User | 16192 |
| Sequence Diagram: Various fixes | * A problem with missing highlighting of items in the structure tree when selecting objects in the sequence diagram graph has been fixed. * The right-click edit menu for Frames now appears on the Frame instead of on its parent. * The visualization of nested Fragments/Frames is corrected/improved. * Connecting lifeline. * Orphaned fragments (those that are nested and do not refer to any Message) are now placed properly in the graph. | Bug fix | User | 16151, 16191, 16187, 16198 |
| Server: Introduction of parallel reads | With R40, we have introduced multiple Read queues on the server. This means that multiple short reads can now successfully be performed against the server while a long read is in progress. Users doing short reads no longer have to wait until the long read is completed.  There is no change related to the Write queue. As before, only one (1) write executes at a time on the server. Note also that this is on the server side. There is no change to the Client API. | Improvement | User | 16332, 16200 |
| swExplorer: Active client no longer locks screen | When performing long operations that previously "locked" the client and screen, the client now remains responsive so you can continue working in your Windows session. The swExplorer client can be resized and moved.  In some views, the Minimize and Maximize buttons can be used. Support for them will be made available for more views in future releases. | Improvement | User | 16292 |
| swExplorer: Better progress/connection information | With R40, we have introduced a notification bar so users can see the server calls of a long operation while it is executing. This way, it's clear that the client is still performing the operation. In addition, there is now improved information regarding lost connection to server and reconnection attempts, as well as the inability to login to a server.  When applicable, the blue notification bar displays just below the menu ribbon.  **Long operations**  For long operations, such as loading a data-heavy structure or view which must retrieve data from the server, you will now see the server calls count moving as the operation progresses.  Together with the new server calls counter presented above, we have begun replacing the green progress bar, currently located in the views, with a "**Loading...**" message. With R40, we have introduced this in the **Complete status** view. The change will be implemented in additional views in future releases. The "Loading..." message, along with the notification banner, help to ensure users that their client is still working.  If you chose to close (X out) the application while a long operation is in progress, a **Close application** confirmation dialog will display where you can make your selection.  **Tip:** If you want to let your long operation run, open a second client instance to continue working.  **Lost connection**  When a connection between a client and the server becomes disconnected, e.g., due to internet or VPN outage, the notification bar will inform of the situation.  The blue Status bar at the bottom of the client displays the reconnect attempts as before.  You can use the existing **Reconnect** dialog to attempt to reconnect. As always, reconnection is not possible if the lost connection was due to the server being down, or if the client has been out of sync with the server for too long.  **Cannot reach server**  A new **Connection timed out** dialog informs the user when the client cannot connect to the specified server, and offers troubleshooting suggestions. | Improvement | User | 16335, 16358 |
| Validity check: Filter and column error | This is a fix for an error that occurred when filtering by a column and moving columns. | Bug fix | User | 16112 |
| Validity check: Various improvements | * The Validity check option will now be hidden when not configured for use. * The configuration now supports the use of comments in the XML. * When you sort by a column and then filter using a checkbox, the selected sort will remain in place. | Improvement | User, Architect | 16127, 16129, 16113, 16128, 16136 |