TIBCO Business Studio™ iProcess to BPM Conversion

Version 4.3.2

May 2022

Contents

Migration Considerations

Existing iProcess customers may want to consider migrating their iProcess applications to the TIBCO ActiveMatrix® BPM platform as a starting point for their development. Whilst the approach of migrating rather than considering the benefits afforded by the additional functionality provided by TIBCO ActiveMatrix BPM has its drawbacks it is still a legitimate starting point in some cases.

It should be noted that there are several aspects that might need to be considered when 're-platforming' an application from iProcess Engine to ActiveMatrix BPM, including:

- Re-platforming Process Models
- Re-platforming the external systems built upon those models and the various iProcess Engine APIs
- Interfacing to any existing EAI layer
- Dealing with in-flight process instances
- Dealing with historical data
- Migrating user accounts

After migration we recommend that the application is reviewed to make sure that it is the functional equivalent of the original.

A fully automated migration is not practical though it is desirable to make the migration of existing iProcess business processes as automated as possible. Migration considerations can be categorized as follows:

- There are many common features and constructs of iProcess business processes that have a generic equivalence in BPM and as such can be automatically converted with minimal or no human intervention.
- There are some features and constructs of iProcess business processes that do not have a generic equivalence in BPM but may have some recognizable meaning and equivalence in a specific iProcess business process example.
- There are some features and constructs of iProcess business processes that have no generic and no customer-specific equivalence in BPM.

The iProcess To BPM Process Conversion feature therefore focuses on:

- Automatically converting aspects of business processes that have generic equivalence in iProcess and BPM.
- Provision for customer extensibility of the conversion feature. The customer will have the ability to convert constructs that follow certain recognized patterns in their own business processes that otherwise could not be migrated in a more generic situation.

Converting Processes Using the iProcess To BPM Import Wizard

This import wizard allows the import and conversion of iProcess Modeler XPDL(s) from the file system into a BPM project in TIBCO Business Studio XPDL format suitable for the BPM destination.

For requirements and recommendations around exporting the iProcess Modeler XPDL, see Restrictions.

Procedure

1. Select **Import** > **Business Process Management** > **Import and Convert iProcess to BPM XPDL(s)**. You can also launch the import wizard by right-clicking the Process Packages special folder and using the context menu **Import** > **Import And Convert iProcess to BPM XPDL(s)**.

2. Browse to the iProcess XPDL(s) you want to import and browse to select the folder where you want to locate the XPDL(s) and click **Next**.

3. Import Validation validates the files. Provided the validation is successful you can now click **Finish** to begin the import and conversion process.

4. The conversion process starts. When the conversion is complete, the XPDL files are saved under a process package folder. The import creates a separate new process package for each process in the incoming XPDL, named after the process.

    - You can abort the conversion at any time before it completes using the abort button. It is not possible to abort once the save of processes starts (to provide consistency that either all converted processes are saved or none).

    - Where a process has already been imported into the workspace (for example, there is a BPM destination process with the same name), the existing workspace process is preserved and the incoming process is ignored. References to that process (for example, sub-process call steps) are switched to the existing workspace process.

See Restrictions for issues you should be aware of when importing processes.

iProcess to BPM Conversions

During the conversion process, the iProcess XPDL is first converted into TIBCO Business Studio XPDL for the iProcess Destination.

Then in a second stage the Studio iProcess XPDL is (automatically) migrated to the BPM destination during which various conversions are made to migrate between equivalent process features in the iProcess and BPM destinations.

The following conversion extensions are implemented in this release:

- The conversion moves Participants from Process to Package level.
- The Destination Environment for your process is set to BPM (all iProcess destinations that are enabled for the process are removed and replaced by the BPM destination).
- The conversion converts expressions and scripts from iProcessScript grammar to TIBCO Business Studio ActiveMatrix BPM JavaScript.
- CUSTAUDIT(…) iProcess Script Expression
- Arrays: there are several important differences between array handling in iProcess and ActiveMatrix BPM which are covered in this conversion.
- Unsupported iProcessScript SPECIALCHARS function will be replaced by a literal string copy of its original parameter.
- Convert GETPROCESSNAME() to String Literal
- CALCTIME().
- Standalone script objects will be replaced by using CALL(SCRIPT) or SCRIPT(SCRIPT, params …) expressions from within another script.
- The conversion converts standalone receive tasks to Message Event Handlers.
- Where the iProcess procedure has declared public start-steps, the conversion creates an additional message start event for each named public start step and directly link it to the named start step.
- EAI Withdraw/Delayed Release: in ActiveMatrix BPM, Delayed Release is two separate activities, a Service Task that performs the action on external system followed by a Receive Task that holds the flow until it is triggered via web service invocation.
- Event cannot have more than one incoming flow.
- Leading digit process content naming restrictions for WSDL Generation: where the artifact name has leading digits, the names derived from it will be prefixed with an underscore.
- Statically named iProcess user/group user step addressees will be replaced by BPM participant RQL queries that reference organization model resources with specific names.
- EAI DB (SQL): iProcess to Studio iProcess XPDL already converts EAI DB step into Database Service Task. Some small modifications are made in this conversion.
- iProcess EAI Mail step will be converted to the ActiveMatrix BPM Email service task.
- Transaction control steps: there is no direct equivalent for this activity type in ActiveMatrix BPM so they are converted to a task type **None** activity.
- iProcess dynamic sub-procedure steps and ActiveMatrix BPM dynamic sub-process invocation tasks are similar, therefore most of the conversion is performed in the initial iProcess to Studio iProcess XPDL conversion. This conversion sets ActiveMatrix BPM dynamic sub-process tasks to multi-instance-parallel, they have the "Allow Unqualified Sub-Process Identification" property set by default, and the multi-instance 'loop expression' (number of instances in the case of parallel) is automatically set to RuntimeIdentifierArrayField.size().
- Duplicate activity names in a process are not permtted in BPM, so when converted will be suffixed with an incrementing integer.

- Set Unique Parameter Label.

- SW_fields: · `SW_` and `IDX_` system fields will be **removed** (from Studio iProcess XPDL generated by initial import) and scripted use of a specific subset of system fields will be converted to ActiveMatrix BPM JavaScript equivalents.

- The conversion removes iProcess extensions.

- Out of bound gateway names should not truncate gateway labels whose length exceeds 58 chars.

- Handle date to date and time to time field assignment semantic differences (BPM is not copy by value, so needs to create new date from right hand side so that fields don't reference same object internally).

iProcess to BPM Conversion Extensions

It is possible to extend the standard iProcess to BPM conversions in order provide conversion for application specific process constructs that otherwise could not be handled by the standard conversion.

For more information see Getting Started: Creating Custom Conversion Extensions , and the TIBCO Business Studio user help for *Business Studio Customer API Feature Documentation*.

The iProcess BPM conversion API JavaDoc and Extension Point documentation can be seen when creating a contribution to the `com.tibco.xpd.customer.api.iProcessBpmConversion` extension point from your own Eclipse `plugin.xml`).

Converting TIBCO Business Studio iProcess XPDL to TIBCO Business Studio BPM XPDL

This wizard converts TIBCO Business Studio XPDL(s) (selected/ contained in the selected TIBCO Business Studio project) already configured for the iProcess destination into TIBCO Business Studio XPDL format suitable for the BPM destination.

This conversion operates on XPDL files already in a TIBCO Business Studio iProcess workspace project, and the user specifies a separate TIBCO Business Studio BPM project as the target for the converted files. If the selected XPDL files reference processes or process-interfaces in other XPDL files, then those XPDL files will be automatically added to the set of files to convert.

Procedure

1. From the individual iProcess project, right-click on the iProcess Project, or process package special folder of an iProcess project, or multiple XPDL files of the same iProcess project, select **Convert iProcess XPDL(s) for BPM Destination**.

   > An iProcess project here means a project with the iProcess destination enabled.

   > When activated on a project with the iProcess destination, if the project does not contain any valid iProcess XPDL, it will show a message dialog stating "No Valid iProcess XPDL found for Conversion".

   This invokes the Convert Studio iProcess XPDL(s) to Studio BPM XPDL(s) wizard.

2. Browse to select a target folder for the XPDL(s) (which cannot be the same as the source folder). Click **Next**.
   The validation starts.

3. Click **Finish** when the validation has completed and if you do not need to resolve any issues.
   XPDLs and any referenced WSDL are copied to the target BPM project.

Getting Started: Creating Custom Conversion Extensions

In order to contribute additional conversions to the **Import/Convert iProcess to BPM XPDL(s)** feature you will need to perform a number of tasks using an Eclipse (Indigo or later) installation and a clean workspace (separate from TIBCO Business Studio and TIBCO Business Studio workspace).

Custom Conversion extensions can also be used to contribute `LifeCycleListener`, which enables the user to perform tasks at specific stages of the conversion by providing an extension with a `lifecyclelistener` contribution and implementing what is to be done as a subclass of `AbstractIProcessToBPMLifeCycleListener` under appropriate methods like `importAndTransformComplete()`, `packageSeparationComplete()`, `conversionComplete()`.

Procedure

1. Setup a JRE /JDK definition (**Window** > **Preferences** > **Installed JRE's**) that is appropriate for your installed TIBCO Business Studio.

   - For example: A `JDK 1.7 64-bit` installation.

   - Set the **Default VM Arguments** to: `-XX:MaxPermSize=256m -XX:+UnlockDiagnosticVMOptions -XX:+UnsyncloadClass -XX:+UseParNewGC -Xms512m -Xmx600m`

2. Create a target platform (**Window** > **Preferences** > **Target Platform** > **Add**) to reference the TIBCO Business Studio installation.

   a) **Target Definition**: Select **Empty Target Definition.**
   b) (**Next**): Select **Target Content**: **[Add...] Installation**.
   c) (**Next**): Select **Add Installation:** Select the following folder in your TIBCO Business Studio installation: *STUDIO_HOME*`\studio\3.6\eclipse`
   d) Click **Finish**.
   e) Back in **Target Platform** select your new target platform.

3. Create your own Eclipse `Plug-in` Project.

4. In the project's `MANIFEST.MF` editor, add a dependency on the TIBCO Business Studio Customer API plug-in.

5. In the project's `plugin.xml` editor contribute an extension to the iProcess to BPM Conversion extension point.

   a) **Editor's Extensions tab**: Click **Add...** and select **com.tibco.xpd.customer.api. iProcessBpmConversion**

      From the selection dialog (or after you have added contribution) you can access the **TIBCO Business Studio - BPM Edition,TIBCO iProcess to TIBCO ActiveMatrix BPM conversion feature** extension point description (this contains a full description of the extension point and its use).

   b) After adding the main contribution, add either **Converter** or **LifeCycleListener** element(s) as required (right-click **contribution** > **New** > **Converter | LifeCycleListener**) .

   c) Set the properties of contribution elements (on the right-hand side) including creating a contributed class of the appropriate type for the element.

      For complete description of Converter and LifeCycleListener contribution class API's see **Business Studio** > **User Help** > **Business Studio Customer API Feature Documentation** > **AbstractIProcessToBPMContribution | AbstractIProcessToBPMLifeCycle Listener**.

6. Develop your conversion code.

7. Create a Run | Debug Configuration.

   a) Create an **Eclipse Application** configuration.

b) Set **Program to Run** to `com.tibco.xpd.branding.prod.`
c) **Run.**

Debugging iProcess Import/Convert

You can use the **TIBCO BPM Debug Preference** page to configure the levels at which XPDL files are output to the debug file folder during import/conversion. When this configuration is enabled, copies of the XPDL models are created in a temporary sub-folder (`iProcessConversionDebug`) within the TIBCO Business Studio Workspace. It is switched **off** by default.

Procedure

1. Select **Window** > **Preferences** > **TIBCO BPM Debug** > **iProcess Import/Convert Debug Mode**.

2. Select from the following options to activate conversion debug mode:

   - **No Debug**: default option.
   - **Basic** (preserve files at basic conversion stages).
   - **Enhanced** (preserve files after each conversion stage).

Equivalence Table: iProcess to BPM

This table covers equivalence between iProcess and BPM objects.

iProcess	BPM
Project Objects	
None	Project
Package	A package will be created for each iProcess procedure even where there were multiple procedures in each iProcess XPDL package.
Procedure	Process
I/O Parameter Template	Process Interface
Sub-Procedure (Process referenced by an reusable sub-procedure call or a process that has formal parameters)	Sub-Process
All participants appear in the Users or Roles column in iProcess. An additional administrative task in the iProcess Engine is to create the required users and groups.	Package level participant
Field.	Data Field (Package and Process)
I/O parameter and field. The mandatory flag on an process formal parameter is equivalent to the Required column in iProcess. This column is on the **Procedure** > **IO Parameters** menu in iProcess when defining a sub-procedure. Using the mandatory flag means that the sub-procedure requires this parameter to always be supplied.	Process Formal Parameter
Activities and Tasks	
Sub-Procedure call	Reusable Sub-Process
Enterprise Application Integration (EAI) step with the appropriate type set.	Service Tasks
EAI Script step	Script Task
Complex router	Task Type (None)
iProcess Event	Receive Task

iProcess	BPM
User Step	User Task
User step participant, depending on addressee type	User Task Participant Field = Performer Data Field, all other types = Org Model Query Participant (with query fetch org model resource with same name as addressee e.g. "resource(name = "CallOperators"))
Formflow Form URL	User Task Form URL
Field instance on iProcess form or Formflow parameter.	User Task Parameter
Script object	User Task Script
EAI Mail	Email service task
EAI DB step	Database-Stored Procedure service task
Transaction Control Steps	No direct equivalent for this activity type in BPM: the difference between iProcess and BPM runtimes that affect this behavior is that in iProcess, contiguous EAI steps are run in a single database transaction and therefore the Commit step would perform an 'earlier than expected commit' between these whereas in BPM a commit is performed after each and every task and event. • Commit steps: are converted to a Task type **None** activity. • Abort steps: are converted to a Task type **None** activity with an associated **FIXME** text annotation describing that there is no equivalent of abort step in BPM.
Delayed Release/EAI Withdraw	Delayed release is handled as two separate activities; a **Service Task** that performs the action on external system followed by a **Receive Task** that holds the flow until it is triggered via web service invocation.
Events	
Start/Stop	Start/End Event
In-flow event "Standalone Event handler"	Receive Task Message Event
Deadline expression	Timer Event Script

iProcess	BPM
Deadline on iProcess step Timer event on equivalent task boundary with equivalent withdraw / continue on timeout setting.	Timer Event (on Task boundary)
Withdraw link	Signal Event pair (throw and catch on task boundary)
Gateways	
Complex Router	Task type none (small green)
Condition with the Expression defined by the condition on the conditional Sequence Flow:	Gateway with one input, one conditional output, and optionally one default output
Wait	Parallel Gateway (multiple unconditional input)
Connecting Objects	
Deadlines Link	Sequence Flows from a Timer Event (attached to task boundary)
Diagram Annotation	Text Annotation
Swimlanes	
Lanes in single automatically created Pool	Lane
Naming	
Step Name	Activity/Task Name
Description	Label
Extended Description	Label
Leading Digit Process Content Naming Restrictions For WSDL Generation	iProcess procedure content names are generally reasonably restricted to be within a subset of permitted character set for the equivalent artifact in BPM. Where the artifact name has leading digits, the WSDL content NCNames derived from it will be prefixed with underscore.
Arrays	

iProcess	BPM
Array[....] (with explicit index specification using square bracket notation)	Two new JavaScript methods: • `ScriptUtil.setArrayElement(arrayField, index, value)` which allows an existing array element (**or the next additional element** at end of current list) to be set. • `ScriptUtil.getArrayElement(arrayField, index)` which allows an existing element to be read from the list where null is returned if a non-existent element index is specified. In ActiveMatrix BPM, arrays are implemented as JavaScript list objects. Therefore elements must be added contiguously starting from index = 0 (this was not necessary in iProcess).
Arrays with implicit index specification (via SW_GEN_IDX or IDX_xxx index field)	Not supported: these will result in script problem markers that must be addressed manually.
Scripting	
Standalone Script Object	Calls to standalone scripts using `CALL("SCRIPT")` have the content of the referenced script inserted in place of the call. Script calls using `SCRIPT("SCRIPT", param, ...)` method cannot be used because of the parameter variables. A commented out version of these is inserted.
Unsupported iProcessScript `SPECIALCHARS` Function	JavaScript string with special character notation (\n = newline etc)
SW_ fields:	BPM Javascript Equivalent:
`SW_NA`	null
`SW_TIME`	`DateTimeUtil.createTime()`
`SW_DATE`	`DateTImeUtil.createDate()`
`SW_PRONAME`	`Process.getName()`
`SW_CP_VALUE`	`Process.priority`
`SW_CASENUM`	`Process.getId()`
All others	References to other `SW_` system values in scripts will require additional human intervention as `getId()` returns string (so depends how `SW_CASENUM` was used) and hence there maybe problem markers to resolve.

Equivalence Table: What Will Not Be Converted

This topic described what cannot be converted. These will generally result in problem markers that will need to be addressed manually.

Arrays with implicit indexing via `SW_GEN_IDX` or `IDX_xxx` index fields.
Conditional step Deadlines
Conditional delayed release configuration
Abort steps
When process instance is cancelled or terminated, EAI withdraw action service will not be executed.
Some flow related scriptable capabilities such as `GOTO` step, `SETSTEPSTATUS`, `CASESTART`, `EVENT` and so on are not supported in BPM.
Commit steps

Preparation for Conversion: Recommendations

TIBCO recommend that you spend time preparing for conversion to ensure that you make the conversion as effective as possible. The following is a list of issues you should be aware of.

iProcess/BPM Objects	Required Actions to Convert
Standalone Event Steps /Message Event Handlers	For correlation data: • Incoming message activities (except start activities) require correlation data mappings to identify the target process instance for the incoming message. • It will not be possible for the base conversion to ascertain which parameters should be used for correlation. • It may be possible for iProcess customer extension to derive this and add correlation data and the data mappings appropriately.
(iPE) CUSTAUDIT(…) iProcess Script Expression	Although the *OriginalAuditId* may not be meaningful to the user, its inclusion in the expression makes it possible for an iProcess customer to contribute a conversion extension to replace the number with a more meaningful message from their known custom audit message set.
(iPE) Transaction Control Steps	For tasks created from Abort steps, the user needs to consider how to handle parts of their application that require a transaction across multiple-activities as there is no equivalent in BPM where process flow state is committed after each activity.
[EAI] Scripts	To deal with any constructs with no direct equivalence (or for which automatic conversion to similar features in BPM is not possible).
EAI Mail	If custom SMTP configuration was used in iProcess applications then the equivalent ActiveMatrix BPM shared resources will need to be created / late bound.
iProcess EAI DB step/Database-Stored Procedure service task	A default database system participant will be created for the task, but requires human / conversion extension intervention to configure that participant.

iProcess/BPM Objects	Required Actions to Convert
Standalone Script Object	`SCRIPT(SCRIPT, params …)` expressions will be in-lined and commented out. Human intervention will be required to fix and uncomment these scripts (matching passed parameter data to parameters in the commented out script expressions).
Arrays	Use of arrays in implicit ways requires human intervention. • This is highlighted at design time by problem markers on the converted JavaScript. • This is because `Array = ElementValue;` is not permitted. • And also because the no data field will be created for the `SW_GEN_IDX` and `IDX_ARRAY` iProcess fields. Explicit assignment of array elements in non-contiguous, zero based manner require human intervention. • A runtime exception results in any attempt to perform `setArrayElement()` where the provided index is greater than index of the next element after the current last element in the list.
Delayed Release	Human intervention is required to design appropriate trigger for the continuation of the flow after the service task.
User Task Addressee Handling	Provided that the organization model contains the required statically defined users and groups, then no human intervention should be required.

Restrictions

You should be aware of the following restrictions when migrating processes from iProcess to BPM.

- iProcess Modeler export has a setting (**Save Referenced Subprocedures**) that you can select from **Procedure management mode client** > **Window** > **Options** that allows you to export a Referenced Subprocedure when the main procedure is exported. We recommend that you use this setting when exporting, to achieve the best result when converting and importing into TIBCO Business Studio BPM processes.
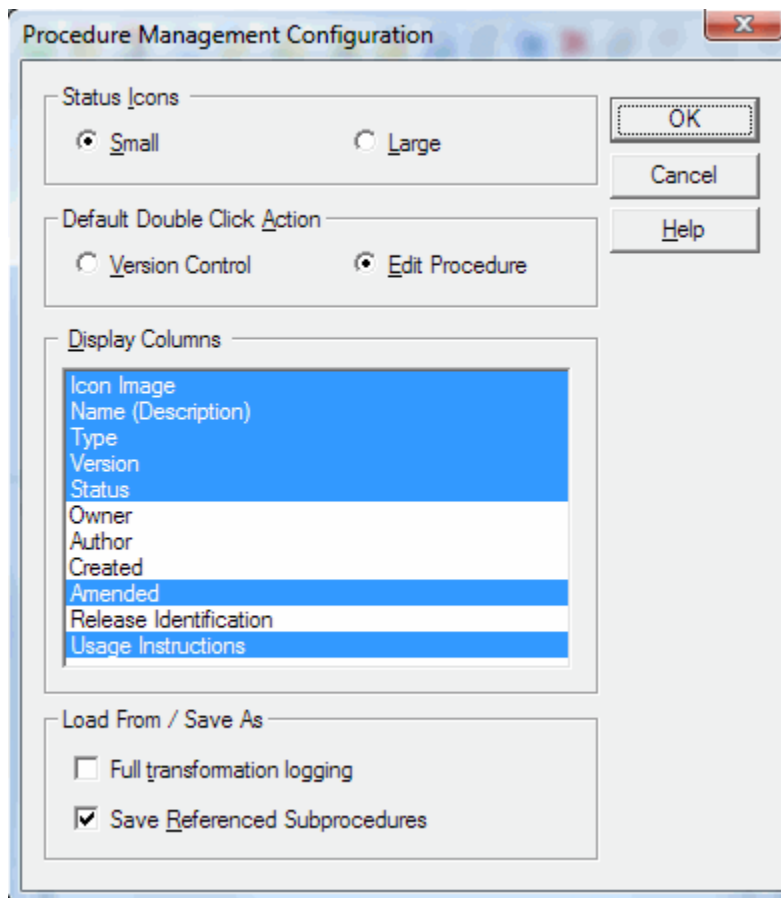
- When you perform an import, you should import XPDL files that have been exported at the same time from the same server to ensure consistency of sub-processes.

- You should ensure that referenced sub-processes and sub-process I/O templates are exported with the processes and sub-processes that reference them. Failure to do so will cause broken references to process interfaces in TIBCO Business Studio (even if the I/O template is subsequently imported). Recent versions of TIBCO iProcess Windows Client support an option to ensure that this is so:

  The IProcess Modeler Client saves the referenced I/O templates along with the referencing Procedure/ Subprocedure, whereas to save all the Subprocedures based on the I/O templates, a preference is provided as shown below:
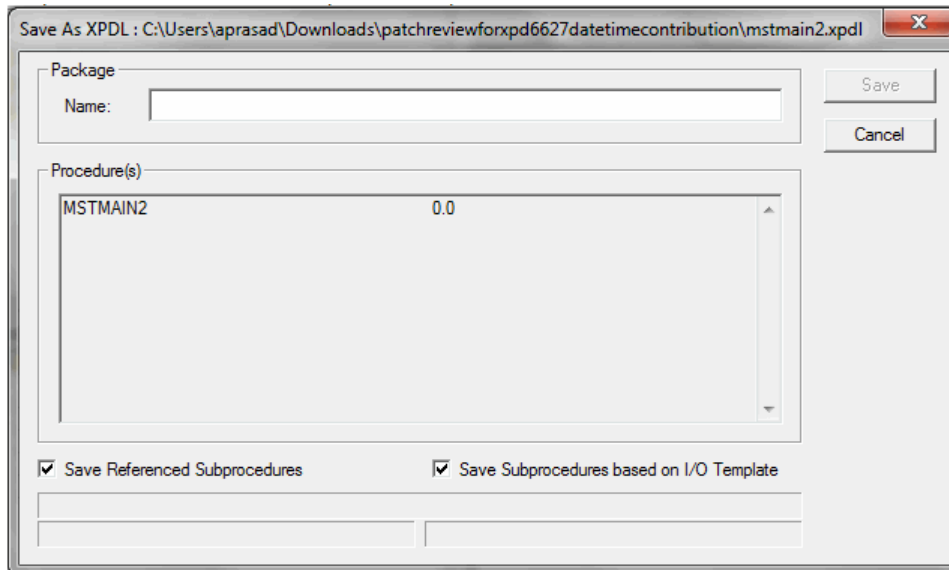
If this is not available, then ensure you select referenced I/O templates explicitly when exporting referencing processes from iProcess.

It is not necessary to use the **Save Subprocedures Based on I/O Template** for correct functioning. However the presence of the option indicates that you are using an iProcess client that automatically exports referenced I/O templates as well as sub-procedures.

- After converting an iProcess project XPDL to a BPM project XPDL, problem markers with quickfixes may be shown because iProcess XPDLs and BPM XPDLs have the same process IDs.

    - Close iProcess Project and clean build the BPM Project to remove the errors.

    - Use Quick fix to recreate object IDs

- You may get an error if the EAI web-service task did not originally statically contain the WSDL. Copy the WSDL into a new Service Descriptors folder in the TIBCO Business Studio project and re-select from service task if necessary.

TIBCO Documentation and Support Services

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website, mainly in HTML and PDF formats.

The website is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The following documentation for TIBCO Business Studio is available on the TIBCO Business Studio Product Documentation page:

- TIBCO Business Studio™ Release Notes
- TIBCO Business Studio™ Concepts
- TIBCO Business Studio™ Modeling User's Guide
- TIBCO Business Studio™ - Analyst Edition User's Guide
- TIBCO Business Studio™ - BPM Implementation
- TIBCO Business Studio™ Forms User's Guide
- TIBCO Business Studio™ Simulation User's Guide
- TIBCO Business Studio™ Customization
- TIBCO Business Studio™ - Analyst Edition Installation
- TIBCO Business Studio™ - BPM Edition Installation
- TIBCO Business Studio™ iProcess to BPM Conversion

How to Contact TIBCO Support

Get an overview of TIBCO Support. You can contact TIBCO Support in the following ways:

- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the TIBCO Support website.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to TIBCO Support website. If you do not have a user name, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the TIBCO Ideas Portal. For a free registration, go to TIBCO Community.

Legal and Third-Party Notices