

webMethods Integration Server 10.2 Readme

April 2018

This is an innovation release. Innovation releases have a much shorter support cycle than standard releases. To learn more, please visit <http://empower.softwareag.com/innovationrelease/>."

This file contains important information you must read before using webMethods Integration Server 10.2. You can find user documentation on the [Documentation website](#) or the [TECHcommunity website](#). At those locations, you can also find suite-related security and globalization information.

Included in this file is information about functionality that has been added, removed, deprecated, or changed for this product. Deprecated functionality continues to work and is supported by Software AG, but may be removed in a future release. Software AG recommends against using deprecated functionality in new projects.

1.0	Critical Information	1
2.0	Known Issues	2
3.0	Usage Notes	10
4.0	Fixes Included in Each Release	14
5.0	Other Resolved Issues	20
6.0	Documentation Changes	100
7.0	Terminology Changes	101
8.0	Added, Removed, Deprecated, or Changed Items	101
9.0	Added, Removed, Deprecated, or Changed Built-In Services	125
10.0	Added, Removed, Deprecated, or Changed Parameters	136
11.0	Added, Removed, Deprecated, or Changed APIs	154
12.0	Copyright Information	159
13.0	Support	160

1.0 Critical Information

This section lists any critical issues for the current release that were known when this readme was published. For critical information found later, go to the Knowledge Center on the [Empower website](#).

- PIE-50502
Migrating Integration Server 10.1 to 10.2 version fails with a NullPointerException.
Migrating Integration Server 10.1 in which a functional alias is pointing to MySQL Community Edition or Enterprise Edition database to 10.2 fails with a NullPointerException.
There is no workaround for this issue.

2.0 Known Issues

This section lists any issues for the current release that were known when this readme was published. For known issues found later, go to the Knowledge Center on the [Empower website](#).

- PIE-50294
Integration Server Administrator does not display existing active WebSocket server sessions after a particular session disconnects.
In Integration Server, when one or more WebSocket server sessions are running and a user disconnects a particular session, Integration Server Administrator does not display the remaining active WebSocket server sessions.
There is no workaround for this issue.
- PIE-50699
WebSocket response headers contain server version information in the responses.
There is no workaround for this issue.
- PIE-50624
Changes to Lockout Duration do not reflect unless Integration Server is restarted.
In Integration Server Administrator, changing and saving the value specified for Lockout Duration field in Security > User Management > Account Locking Settings > Edit screen, does not reflect unless you restart Integration Server.
To work around this issue, restart Integration Server.
- PIE-50355
Integration Server returns unexpected response for a REST API.
When adding two REST V2 resources to a REST API descriptor that have same URL template but different HTTP method, Integration Server returns an unexpected response during runtime; however finds correct mapping service.
To work around this issue, do not add REST V2 resources with the same URL templates.
- PIE-50354
Integration Server fails to generate a valid Swagger file.
When creating a REST API descriptor from a Swagger document, Integration Server generates an invalid Swagger document. Integration Server fails to set the correct value of isRequired attribute for a REST parameter that should be according to the Swagger document and always sets the default value to False.
There is no workaround for this issue.
- PIE-50616

Integration Server becomes unresponsive when creating a REST API descriptor for a service or for a REST V2 resource.

Integration Server becomes unresponsive while creating a REST API descriptor for a resource that is associated with a service having documents with cyclic references. Integration Server does not create the REST API descriptor in this situation.

There is no workaround for this issue.

- PIE-50515

When reloading a package on which shutdown services for other packages depend, package reload fails with the error "[ISS.0028.0017E] Shutdown service <serviceName> was not found in <packageName> package".

There is no workaround for this issue.

- PIE-50362

Integration Server fails to add a new REST V2 resource to an empty REST API descriptor.

In Designer, if a REST V2 resource associated with a REST API descriptor is deleted and then a new REST V2 resource is added to the REST API descriptor, Integration Server throws a NullPointerException.

There is currently no workaround for this issue.

- PIE-50738

Integration Server throws unexpected response while invoking an URL from the REST API descriptor.

When creating a REST API descriptor by importing a Swagger file that contains a path and requires a document as input, Integration Server throws an unexpected response as '_generatedInput', even if the path is invoked with proper input.

There is currently no workaround for this issue.

- PIE-49688

Mediator virtual services with large MTOM attachments consume a seemingly excessive amount of CPU time.

Integration Server converts a large MTOM attachment for a Mediator virtual service to an HTTP chunked format instead of simply streaming the MTOM attachment. The conversion to HTTP chunked format results in the virtual service consuming a large amount of CPU time. Additionally, when Mediator virtual services use a GZIP encoding, the HTTP chunking results in the sending of an extra HTTP 200 response.

There is no workaround for this issue.

- PIE-49912

Filtering the list of scheduled tasks does not work as expected for some searches.

In Integration Server Administrator, when using the wild-card character '*', filtering the list of scheduled tasks works only with '*' entered prior and at the end of the search criteria.

There is no workaround for this issue.

- PIE-49747

Unable to set the Description parameter to blank for a scheduled task using public services.

For a scheduled task, setting the Description parameter to blank fails using the following services:

pub.scheduler:updateOneTimeTask, pub.scheduler:updateRepeatingTask, and pub.scheduler:updateComplexTask. When set to blank, the parameter retains the previous description upon save.

There is no workaround for this issue.

- PIE-49599
Integration Server returns an extraneous HTTP 200 response after sending a response using chunked transfer encoding in combination with GZIP.
If a client that accepts GZIP responses invokes an operation in a provider web service descriptor or a Mediator virtual service and the response is sent using chunked transfer encoding, Integration Server sends an extraneous HTTP 200 response.
There is no workaround for this issue.
- PIE-50359
Integration Server clears the response code from the REST API descriptor.
When adding a new response code to a resource in REST API descriptor and then moving the resource from one package to another, Integration Server removes all the new response codes from REST API descriptor.
There is no workaround for this issue.
- PIE-50364
Integration Server does not rename a REST V2 resource.
When renaming a REST V2 resource by changing the case, Integration Server does not update the reference to that REST V2 resource.
There is no workaround for this issue.
- PIE-50360
Integration Server saves invalid path in the REST API descriptor.
When saving the basepath in a REST API descriptor, Integration Server fails to validate the basepath and saves it, thereby throwing an error at runtime.
There is no workaround for this issue.
- PIE-50361
Integration Server throws an exception while adding resources to REST API descriptor.
When adding two REST V2 resources with the same name but different in case, Integration Server throws an exception.
There is no workaround for this issue.
- PIE-48001
The Swagger tab for a REST API descriptor does not get updated on deleting a Document input parameter from a flow service associated with the operation of the REST V2 resource that is included in the descriptor.
On deleting a Document input parameter from a flow service that is associated with an operation of a REST V2 resource, the Swagger tab for the corresponding REST API descriptor does not reflect the particular change even after manually refreshing the REST API descriptor.
There is no workaround for this issue.

- PIE-47159
 Variables referred to in map set values and selected for refactoring are not previewed.
 If more than one variable is referred to in a map set value, selecting one of them for refactoring does not list the particular variable in the Refactor Variable preview window.
 There is no workaround for this issue.
- PIE-47850
 The field labels in some dialog boxes opened from Integration Server Administrator are not localized.
 There is no workaround for this issue.
- PIE-50675
 Duplicate UUIDs when invoking pub.publish:publish inside a LOOP or a REPEAT.
 Beginning with Integration Server 10.2, the `_uuid` field can be set in the document envelope when publishing a document to Universal Messaging using the `pub.publish:publish` service. Integration Server generates and assigns a UUID to the `_env/uuid` field if one is not specified. However if the `pub.publish:publish` service is invoked multiple times, such as in a LOOP step or a REPEAT step, and the UUID generated by Integration Server is not dropped from the service output, subsequent invocation of `pub.publish:publish` use the same UUID as the initial invocation. This results in published documents with duplicate UUIDs.
 To work around this issue, drop `_env/uuid` from the `pub.publish:publish` service output.
- PIE-8533 (was 1-1Z6J9O)
 Integration Server does not shut down if an audit logging queue contains records waiting to be written to a destination.
 To work around this issue, wait for the records in the queue to be written to the destination.
- WSC-515
 Known issue in the JVM Cipher implementation for certain JVM builds may cause decryption failures in Integration Server and other products that use the outbound password manager provided in Integration Server.
 A change in the `PBEWithHmacSHA256AndAES_256` implementation causes Strings encrypted with the cipher in Open JDK builds 1.8.0_64 and before do not decrypt in later builds, beginning around build number 1.8.0_71. If your Integration Server runs on a JVM from Open JDK builds 1.8.0_64 up to, but not including 1.8.0_71, and then upgrade to Open JDK build 1.8.0_71 and after, passwords stored in the Integration Server outbound password manager cannot be decrypted.
 To avoid this issue, change the encryptor before upgrading the JVM. That is, store the password using the same handle but a different encryption code such as `EntrustPbePlus`.
 For information about changing the encryption method for the outbound password file and resetting outbound passwords, see the “Master Passwords and Outbound Passwords” chapter in the *webMethods Integration Server Administrator’s Guide*. You can also change an outbound password using the `pub.security.outboundPasswords:updatePassword` service which is documented in the *webMethods Integration Server Built-In Services Reference*.
 A description of this Open JDK defect can be found here:
<https://bugs.openjdk.java.net/browse/JDK-8162690>
 Note that the Open JDK defect description has some ambiguity regarding the build number in

which this issue was introduced. The issue is observed in build number 1.8.0_74 but may have been introduced in earlier builds. Builds 1.8.0_64 through 1.8.0_74 were not tested by the defect reporter. The WSC-515 description cites build 1.8.0_71 as the Open JDK defect identities 71 as the "Introduced in version" value.

- PIE-25824
Cache-related failures on HP-UX.
When Overflow To Disk is enabled for a local cache, Integration Server returns multiple cache-related exceptions in the log at start up. For example,
"java.lang.IllegalArgumentException: Initial table allocation failed. Initial Table Size (slots) : 64
Allocation Will Require : 1KB Table Page Source :
com.terracottatech.offheapstore.disk.paging.MappedPageSource@56cac09
at com.terracottatech.offheapstore.OffHeapHashMap.<init>(OffHeapHashMap.java:204)
at com.terracottatech.offheapstore.AbstractLockedOffHeapHashMap.<init>(AbstractLockedOffHeapHashMap.java:58)
at com.terracottatech.offheapstore.AbstractOffHeapClockCache.<init>(AbstractOffHeapClockCache.java:61)
at com.terracottatech.offheapstore.disk.persistent.AbstractPersistentOffHeapCache.<init>(AbstractPersistentOffHeapCache.java:30)
at com.terracottatech.offheapstore.disk.persistent.PersistentReadWriteLockedOffHeapClockCache.<init>(PersistentReadWriteLockedOffHeapClockCache.java:26)
at net.sf.ehcache.store.offheap.factories.EhcachePersistentSegmentFactory\$EhcachePersistentSegment.<init>(EhcachePersistentSegmentFactory.java:67)
at net.sf.ehcache.store.offheap.factories.EhcachePersistentSegmentFactory.newInstance(EhcachePersistentSegmentFactory.java:50)
at net.sf.ehcache.store.offheap.factories.EhcachePersistentSegmentFactory.newInstance(EhcachePersistentSegmentFactory.java:27)
at com.terracottatech.offheapstore.concurrent.AbstractConcurrentOffHeapMap.<init>(AbstractConcurrentOffHeapMap.java:98)
at com.terracottatech.offheapstore.disk.persistent.AbstractPersistentConcurrentOffHeapMap.<init>(AbstractPersistentConcurrentOffHeapMap.java:42)
at net.sf.ehcache.store.offheap.disk.EhcachePersistentConcurrentOffHeapClockCache.<init>(EhcachePersistentConcurrentOffHeapClockCache.java:52)
at net.sf.ehcache.store.offheap.disk.OffHeapDiskStoreFactory.createBackingMap(OffHeapDiskStoreFactory.java:213)
at net.sf.ehcache.store.offheap.disk.OffHeapDiskStoreFactory.create(OffHeapDiskStoreFactory.java:63)
at net.sf.ehcache.EnterpriseFeaturesManager.createNonPersistentStore(EnterpriseFeaturesManager.java:211)...."
This issue occurs only on HP-UX 11.
To work around this issue, update wrapper.conf file located under profiles/IS_<instanceName>/configuration directory. Modify wrapper.java.command parameter in wrapper.conf file to point to <SAG_DIR>/jvm/jvm/jre/bin/IA64W/java_q4p
- PIE-8045 (was 1-1Y1BZ7)
Installing a package that contains a schema with a target namespace that is the same as an existing

schema on the Integration Server may result in two sets of definitions or declarations for the same components.

If you only need the schema definitions or declarations contained in one schema, delete the other schema. However, if you need definitions from both schemas, there is no workaround for this issue.

- PIE-45939
State options are not localized in Settings > Messaging > JMS Settings > View JMS Trigger Management > Edit Global JMS Trigger Controls > Individual Standard JMS Trigger Controls screen.
In Integration Server Administrator, the following options for State field in Settings > Messaging > JMS Settings > View JMS Trigger Management > Edit Global JMS Trigger Controls > Individual Standard JMS Trigger Controls > edit all screen are not localized:
Enabled, Disabled, Suspended
There is no workaround for this issue.
- PIE-47546
Creating a REST API descriptor from a Swagger document fails if the Swagger definitions do not contain at least one property.
Creating a REST API descriptor from a Swagger document fails if the Swagger document contains definitions which do not have at least one property.
There is no workaround for this issue.
- PIE-50313
Integration Server fails to remove the references to a REST V2 resource.
When a REST V2 resource is deleted from a REST API descriptor, Integration Server fails to remove the reference to that REST V2 resource.
There is no workaround for this issue.
- PIE-50370
Integration Server fails to refresh the REST API descriptor.
When creating a REST API descriptor by importing a Swagger file and the REST API descriptor is renamed, Integration Server throws an error while refreshing the REST API descriptor.
There is no workaround for this issue.
- PIE-50352
When creating a REST API descriptor from a Swagger definition, Integration Server represents a file type as a String instead of an Object.
There is no workaround for this issue.
- PIE-8185 (was 1-1YBYQD)
Web service connector ends with the error [ISC.0082.9034] Field is absent, field must exist.
If the output signature of a service used as an operation in a provider web service descriptor (WSD) contains a field that has a namespace URI without a prefix, Integration Server adds a prefix when generating a WSDL document for the provider WSD. In the consumer WSD created from the WSDL, the web service connector that corresponds to the operation (IS service) specifies a prefix for the field in the service output. However, the web service provider does not include a prefix with the field in the response. As a result, the contents of the SOAP response cannot be mapped to the web

service connector output and the web service connector ends with the error [ISC.0082.9034] Field is absent, field must exist.

To avoid this issue, if a service will be exposed as a web service, always associate a prefix with a namespace URI for fields in the service signature.

- PIE-8494 (was 1-1Z342R)

The xsi:nil attribute in an element does not convert properly when generating an IS document from an XML document.

If an XML document has an element containing only xsi:nil as an attribute and an IS document is generated from that XML document using the pub.xml.xmlNodeToDocument service, the xsi:nil attribute is generated as an @xsi:nil field for the element in the resulting IS document. This occurs even if the element with the xsi:nil attribute has a simple type string; however, the document type that is created from the XML schema (which is used by the XML document) has a string field for the xsi:nil element instead of the IS document. There is a type difference between the generated document and the document type.

To work around this issue, manually edit the generated IS document to remove the @xsi:nil attribute and then convert the IS document to string field.

- PIE-16451

WSDL generated for a web service descriptor with a service signature, header document type, or a fault document type containing derived document types does not contain the schema definitions for the derived document types.

To work around this issue, create a WSDL with the schema definitions for the derived document types and then create a WSDL first web service descriptor.

- PIE-18649

When creating a WSDL first provider web service descriptor, Integration Server does not preserve the original service name from the WSDL document.

When Integration Server generates a WSDL document for the provider web service descriptor, the service name will not match the service name in the source WSDL document.

There is currently no workaround for this issue.

- PIE-19157

IMAP email listener does not start.

This issue occurs when an IMAP e-mail port is configured to receive requests from an e-mail server that uses NTLM for authentication. With this configuration, the following error is returned when the port is enabled:

```
"Failed to start EmailListener:imap: <UserName>@<HostName>: [ISS.0070.9003] Enable failed:  
Could not log into account <UserName>@<HostName>"
```

To resolve this issue, do one of the following:

If you want to disable NTLM authentication, follow these steps:

1. Open custom_wrapper.conf located under <Software AG_directory>/profiles/IS_default/configuration directory.
2. Add the following property:
wrapper.java.additional.n=-Dmail.imap.auth.ntlm.disable=true
where n is the next unused sequential number in the file.

3. Restart Integration Server.

Note that this behavior is consistent with the past releases of Integration Server.

If you want to enable NTLM authentication, follow these steps:

1. Download jcifs-1.3.15.jar file from <http://jcifs.samba.org/src/> into the <IntegrationServer_directory>/instances/<instanceName>/lib/jars or <IntegrationServer_directory>/lib/jars directory.
2. Restart Integration Server.

▪ PIE-22556

Java service throws java.lang.reflect.InvocationTargetException when attempting to use jars from <JRE_directory>/lib/ext directory, such as classes in the com.sun.crypto.provider package.

If you plan to use jars from the <JRE_directory>/lib/ext directory, you can avoid this issue by modifying the config.ini as follows:

1. Open the config.ini file located in <Software AG_directory>/profiles/IS_<instanceName>/configuration.
2. Add the following line:
osgi.parentClassLoader=app
3. Restart Integration Server.

▪ PIE-32205

Integration Server does not provide Java-based NTLM (Windows NT LAN Manager) support for proxy servers that support NTLM authentication. You can only use the NTLM authentication support in Integration Server to allow clients to access resources on web servers that support NTLM authentication, such as Microsoft Internet Information Server (IIS).

▪ PIE-46867

Editing the "Maximum Entries in Cache" field in Integration Server does not display updated value.

When a user edits the value in "Maximum Entries in Cache" field under Distributed Cache configuration section in Integration Server does not display the new values.

There is no workaround for this issue.

▪ PIE-50348

Integration Server does not add the comments to the generated document.

When importing a Swagger file, if a definition in the Swagger file contains comments, Integration Server fails to add those comments to the corresponding comment section of the generated document.

There is no workaround for this issue.

▪ PIE-50621

In Integration Server Administrator, help content for WebSocket Server Endpoints screen is unavailable.

There is no workaround for this issue.

▪ PIE-50571

The text "API Gateway Transaction Logger" is not translated.

In Integration Server Administrator, on the Settings > Logging > View API Gateway Transaction

Logger Details screen, the text “API Gateway Transaction Logger” is not translated into the targeted language.

There is no workaround for this issue.

- PIE-50359
Integration Server clears the response code from the REST API descriptor.
When adding a new response code to a resource in REST API descriptor and then moving the resource from one package to another, Integration Server removes all the new response codes from REST API descriptor.
There is no workaround for this issue.
- PIE-50528
An invalid message appears while migrating Integration Server from an earlier version to the latest version.
In Linux, while migrating from an earlier version to the latest version of Integration Server, following message appears in the console:

```
../.././install/bin/setenv.sh
```


However, this message is erroneous and can be ignored.
There is no workaround for this issue.

3.0 Usage Notes

This section provides any additional information you need to work with the current release of this product.

- webMethods Broker is deprecated.
- PIE-4086 (was 1-1MDBR7)
In the ClearCase dynamic view, reverting a node after performing a checkout, edit, save, or rename command causes the node to disappear.
When an uncheckout command is issued in ClearCase dynamic view, there is a delay of few seconds before the file is copied to the view. During the delay, the node disappears from Designer. This node reappears in Designer only after the view is refreshed.
This issue occurs only for direct installation of Integration Server in the Dynamic view for certain types of nodes, such as flat file schema. The issue does not occur when working directly in SnapShot view or when working with a folder that is mapped to a SnapShot or Dynamic view.
- PIE-16497
Integration Server does not generate the *doctype field for IS document types generated from derived document types in a schema, when:
 - Deriving a complex type from an empty complex type by extension.
 - Deriving a complex type from a simple type by extension.
- PIE-42419
As of Integration Server 10.0, the Allow List for any new port with an Access Mode of “Deny by Default” now includes the `wm.server.csrfguard:getCSRFSecretToken` service and the

`wm.server.csrfguard:isCSRFGuardEnabled` service. When migrating to Integration Server 10.0 or higher from a version lower than 10.0, for any existing ports with an Access Mode of “Deny by Default”, you must manually update the Allow List to include the `wm.server.csrfguard:getCSRFSecretToken` and `wm.server.csrfguard:isCSRFGuardEnabled` services.

- **PIE-48484**
A web service request that specifies a content type of `application/xml` succeeds in versions of Integration Server prior to version 9.12. However, the web service request fails in Integration Server version 9.12. Prior to Integration Server 9.12, Integration Server did not provide out of the box support to directly handle the content type `application/xml`. If a content handler was not specified for `application/xml`, Integration Server used the default content handler. The default handling is the same as the handling of `text/html`, resulting in XML being placed in the pipeline as an `InputStream`. In version 9.12, Integration Server added a content handler named `ContentHandler_XML` to support the content type `application/xml`, resulting in the content being placed in the pipeline as an XML node. This behavior change causes the web service request to fail in version 9.12 because the web service expects an `InputStream`, not an XML node. To address this issue and avoid having to change the service to accept an XML node instead of an `InputStream`, you can use the `watt.server.content.type.mappings` server configuration parameter to map the `application/xml` content type to the `text/xml` content type. Integration Server will use the `text/xml` content handler when receiving a request that specifies a content type of `application/xml`. Specify the following:
`watt.server.content.type.mappings=application/xml text/xml`
- If your solution includes publishable document types with an encoding type of protocol buffers, you need to edit, save, and synchronize the publishable document types to Universal Messaging while the queues of subscribing triggers are empty. This ensures that subscribing triggers will not fail with a `NullPointerException` if you later choose to edit the publishable document type while queues of subscribing triggers contain published documents that conform to earlier iterations of the publishable document type.
- Integration Server 10.0 introduces the ability to enable or disable follow the master behavior on a per messaging connection alias basis. Prior to Integration Server 10.0, follow the master was enabled globally for `webMethods` messaging. The only way to disable follow the master behavior was to modify the `custom_wrapper.conf` file to include the `wrapper.java.additional.n=DFollowTheMaster=false` where `n` was the next available `wrapper.java.additional` number. Do not use the `DFollowTheMaster` parameter to control the follow the master behavior for Integration Server. This parameter setting will override the follow the master behavior set in Universal Messaging connection aliases and JMS connection aliases.
- When securing web services using policies based on `WS-SecurityPolicy`, you cannot alter an inbound message before the security processing executes or alter an outbound message after the security processing completes. For inbound messages, Integration Server always performs the security processing first upon receiving the message. As a result, Integration Server cannot invoke custom handlers before the security processing of an inbound message. For outbound messages, Integration Server always performs the security processing last, right before it sends the message. As a result, Integration Server cannot invoke custom handlers after the security processing of an outbound message.

- Integration Server uses Xerces Java parser version J-2.11.0. Limitations for this version are listed at <http://xerces.apache.org/xerces2-j/xml-schema.html>.
- If you want to use WS-SecurityPolicy to secure a web service and also want to use MTOM streaming, be aware that if the fields to be streamed are being signed and/or encrypted, Integration Server cannot use MTOM streaming because Integration Server needs to keep the entire message in memory to sign and/or encrypt the message.
- Added support for session-based authentication.
 When a client invokes a stateful service, Integration Server creates a session, and includes the session ID in its response to the client. If the client returns to invoke a service during the same session, Integration Server should use this session ID to identify the client, but this is not happening. Instead, Integration Server is reauthenticating the client for each service invocation. This behavior is similar to that performed for stateless services. When a client connects to a stateless service, the session is not maintained, and Integration Server's response does not include a session ID, so Integration Server must reauthenticate the client during the next service invocation. With this update, if a client returns to invoke a stateful service during the same session, Integration Server uses the session ID to identify the client, rather than performing authentication for each service invocation. This behavior allows the Integration Server to perform session-based authentication for stateful services, that is, authentication that lasts for the duration of the client's session with Integration Server.
 For best performance, use stateful services if your Integration Server receives requests from repeating clients. The client can connect to Integration Server, be authenticated just once, and then issue many service invocations during the same session. Use stateless services if clients typically send a single invocation request to Integration Server at a time. Using a stateless service prevents the creation of sessions that will sit unused, taking up resources in Integration Server.
 Note: This change does not require any changes to your existing implementation.
- Web services security implemented using WS-Security facility in Integration Server does not support partial message operations (Sign/Encrypt). Integration Server allows only the body of the SOAP message to be signed and encrypted.
- Do not modify the following file unless instructed to do so by Software AG:
 <IntegrationServer_directory>\instances\<instanceName>\config\wss\axis2.xml
 Changes to this file may result in an unstable configuration. Software AG will not support issues that arise as a result of changes to this file that were not authorized by Software AG.
- Software AG does not support the deployment of custom handlers or modules via placement of an Axis Module (*.mar) file in the following directory:
 <IntegrationServer_directory>\instances\<instanceName>\config\wss\modules
 Unexpected behavior that arises due to the manual deployment of mar files directly to this location is the responsibility of the user and will not be addressed by Software AG.
- Software AG does not support the deployment of web services via placement of an Axis Archive (*.aar) file in the following directory:
 <IntegrationServer_directory>\instances\<instanceName>\config\wss\services

Unexpected behavior that arises due to the manual deployment of aar files directly to this location is the responsibility of the user and will not be addressed by Software AG.

- In Integration Server version 10.1, the default behavior of the enhanced XML parser now prohibits the support of DTDs and resolution of external entity references. For the `pub.xml:loadEnhancedXMLNode` and `pub.xml:XMLStringToEnhancedXMLNode` services the `inputProcessing\supportDTD` input parameter and the `inputProcessing\isSupportingExternalEntities` now have a default value of false. If an existing solution relies in the previous default behavior (`inputProcessing\supportDTD` and `inputProcessing\isSupportinExternalEntities` were set to true), after migrating to Integration Server 10.1 or higher, you must update invocations of `pub.xml:loadEnhancedXMLNode` and `pub.xml:XMLStringToEnhancedXMLNode` to set `inputProcessing\supportDTD` and `\inputProcessing\isSupportinExternalEntities` to true.
- Now, when you start Integration Server, Integration Server receives configuration settings (for example, the size of the Java heap) from the `wrapper.conf` and `custom_wrapper.conf` files located in the `Software AG_directory\profiles\IS_instance_name\configuration` directory. Integration Server no longer obtains settings from `setenv.bat/sh` or `server.bat/sh`. If you need to modify the default property settings for Integration Server, you can override the settings using the `custom_wrapper.conf` file. The following table shows the settings formerly set in the `setenv.bat/sh` file that are now set using properties in the `custom_wrapper.conf` file:

This setting in setenv.bat/sh...	Is replaced with the following property in custom_wrapper.conf...
APPENDCLASSES/ APPEND_SYSTEM_CLASSPATH	wrapper.java.additional.203= Dwatt.server.append.classes=
JAVA_CUSTOM_OPTS	wrapper.java.additional.n
JAVA_MAX_DIRECT_SIZE	wrapper.java.additional.n=XX:MaxDirectMemorySize=
JAVA_MAX_MEM	wrapper.java.maxmemory
JAVA_MAX_PERM_SIZE	wrapper.java.additional.n=XX:MaxPermSize=
JAVA_MIN_MEM	wrapper.java.initmemory
PREPENDCLASSES/ PREPEND_SYSTEM_CLASSPATH	wrapper.java.additional.202= Dwatt.server.prepend.classes=

The following table shows settings you can change that were formerly in the `setenv.bat/sh` file, but are now located in other files or removed:

Setting	New location (if applicable)
DEBUG_ENABLED	<code>Software AG_directory\profiles\IS_instance_name\bin\startDebugMode.bat/sh</code>
JAVA_PROFILER_OPTS/ PROFILER_ENABLED	Removed.

Java location	wrapper.java.command in <i>Software AG_directory\profiles\IS_instance_name\configuration\wrapper.conf</i>
JMX_ENABLED	Removed. JMX monitoring is enabled by default and cannot be disabled.
JMX_PORT	The port property in <i>Software AG_directory\profiles\IS_instance_name\configuration\com.software.jmx.connector.pid-port.properties</i>

The startup.bat/sh and shutdown.bat/sh scripts contained in the *Integration Server_directory\instances\instance_name\bin* and *Integration Server_directory\bin* directories are deprecated. You should use the scripts contained in the *Software AG_directory\profiles\IS_instance_name\bin* directory to start and stop Integration Server. If you will manage Integration Server through Command Central, you *must* use the scripts located in the *Software AG_directory\profiles\IS_instance_name\bin* directory.

The installSvc.bat file located in *Integration Server_directory\instances\instance_name\support\win32* directory is also deprecated. You should use the service.bat file from the *Software AG_directory\profiles\IS_instance_name\bin* directory to register or remove an Integration Server instance as a Windows service.

For complete instructions for using any of the features affected by these changes, see *webMethods Integration Server Administrator's Guide* and *Working with the webMethods Product Suite and the Java Service Wrapper*.

4.0 Fixes Included in Each Release

This section lists the latest fix level that has been included in each release for each product component. A release is listed in this section only if changes occurred in that release. Go to the Knowledge Center on the [Empower website](#) for detailed information about fixes.

Release 10.2

- IS_8.2_SP2_Core_Fix24
- IS_9.0_SP1_Core_Fix21
- IS_9.5_SP1_Core_Fix18
- IS_9.6_Core_Fix18
- IS_9.7_Core_Fix22
- IS_9.8_Core_Fix18
- IS_9.9_Core_Fix22
- IS_9.10_Core_Fix11

- IS_9.12_Core_Fix14
- IS_9.12_SPM_Fix4
- IS_10.0_Core_Fix3
- IS_10.0_WmCloud_Fix3
- IS_10.1_Core_Fix3
- IS_10.1_SPM_Fix1
- MIG_9.12_MigrationFramework_Fix3
- MIG_10.1_MigrationFramework_Fix1
- WFF_9.7_Fix6
- WFF_9.9_Fix4
- WFF_9.12_Fix3

Release 10.1

- IS_8.2_SP2_Core_Fix23
- IS_9.0_SP1_Core_Fix20
- IS_9.5_SP1_Core_Fix17
- IS_9.6_Core_Fix16
- IS_9.7_Core_Fix19
- IS_9.8_Core_Fix15
- IS_9.8_Tanuki_Fix1
- IS_9.9_Core_Fix15
- IS_9.9_SPM_Fix4
- IS_9.10_Core_Fix9
- IS_9.10_SPM_Fix2
- IS_9.10_Tanuki_Fix
- IS_9.12_Core_Fix7
- IS_9.12_SPM_Fix2

- IS_9.12_WmCloud_Fix2
- IS_10.0_Core_Fix1
- IS_10.0_WmCloud_Fix2
- MIG_9.12_MigrationFramework_Fix1
- WAR_9.7_Fix3
- WAR_9.8_Fix4
- WAR_9.9_Fix2
- WAR_9.10_Fix3
- WAR_9.12_Fix1
- WFF_9.6_Fix5
- WFF_9.9_Fix3
- WFF_9.10_Fix1
- WFF_9.12_Fix1

Release 10.0

- IS_8.2_SP2_Core_Fix22
- IS_9.0_SP1_Core_Fix18
- IS_9.5_SP1_Core_Fix14
- IS_9.6_Core_Fix13
- IS_9.7_Core_Fix14
- IS_9.7_SPM_Fix3
- IS_9.7_SubVersion_Fix1
- IS_9.7_Tanuki_Fix1
- IS_9.7_VCS_Fix1
- IS_9.7_WmCloud_Fix4
- IS_9.8_Core_Fix11
- IS_9.8_SPM_Fix2

- IS_9.8_VCS_Fix1
- IS_9.8_WmCloud_Fix1
- IS_9.9_Core_Fix9
- IS_9.9_SPM_Fix3
- IS_9.9_Tanuki_Fix1
- IS_9.9_VCS_Fix1
- IS_9.9_WmCloud_Fix1
- IS_9.10_Core_Fix6
- IS_9.10_SPM_Fix1
- IS_9.10_VCS_Fix1
- IS_9.10_WmCloud_Fix1
- IS_9.12_Core_Fix4
- IS_9.12_WmCloud_Fix1
- WAR_9.6_Fix5

Release 9.12

- IS_8.0_SP1_Core_Fix32
- IS_8.2_SP2_Core_Fix20
- IS_9.0_SP1_Core_Fix14
- IS_9.5_SP1_Core_Fix11
- IS_9.7_Core_Fix8
- IS_9.7_Tomcat6_Fix2
- IS_9.8_Core_Fix5
- IS_9.9_Core_Fix2
- IS_9.9_SPM_Fix2
- IS_9.10_Core_Fix1
- IS_9.10_OData_Fix1

- WAR_9.7_Fix2
- WAR_9.8_Fix3
- WAR_9.10_Fix2
- WFF_8.2_SP2_Fix6
- WFF_9.5_SP1_Fix4
- WFF_9.6_Fix4
- WFF_9.8_Fix1
- WFF_9.9_Fix1

Release 9.10

- IS_7.1.3_Core_Fix28
- IS_8.2_SP2_Core_Fix19
- IS_9.0_SP1_Core_Fix13
- IS_9.5_SP1_Core_Fix10
- IS_9.5_SP1_Portal_Fix3
- IS_9.6_Core_Fix9
- IS_9.7_Core_Fix5
- IS_9.7_Portal_Fix4
- IS_9.8_SPM_Fix1
- IS_9.9_Core_Fix1
- IS_9.9_SPM_Fix1
- WAR_9.5_SP1_Fix6
- WAR_9.8_Fix2
- WFF_8.2_SP2_Fix5
- WFF_9.0_SP1_Fix2
- WFF_9.5_SP1_Fix3
- WFF_9.6_Fix1

- WFF_9.7_Fix4

Release 9.9

- IS_8.2_SP2_Core_Fix17
- IS_9.0_SP1_Core_Fix10
- IS_9.0_SP1_WmCloud_Fix1
- IS_9.5_SP1_Core_Fix8
- IS_9.5_SP1_WmCloud_Fix1
- IS_9.6_Core_Fix7
- IS_9.6_MobileSupport_Fix1
- IS_9.6_SubVersion_Fix1
- IS_9.6_Tomcat6_Fix3
- IS_9.7_Core_Fix4
- IS_9.7_Migration_Fix1
- IS_9.7_SPM_Fix2
- IS_9.8_Core_Fix2
- WAR_9.5_SP1_Fix5
- WAR_9.7_Fix1

Release 9.8

- IS_7.1.2_Core_Fix47
- IS_7.1.3_Core_Fix27
- IS_8.2_SP2_Tanuki_Fix1
- IS_8.2_SP2_Tomcat6_Fix7
- IS_9.0_SP1_Core_Fix8
- IS_9.0_SP1_Tanuki_Fix1
- IS_9.5_SP1_Migration_Fix1

- IS_9.5_SP1_Portal_Fix2
- IS_9.5_SP1_Tanuki_Fix1
- IS_9.5_SP1_Tomcat6_Fix1
- IS_9.6_Core_Fix4
- IS_9.6_Migration_Fix2
- IS_9.6_Portal_Fix1
- IS_9.6_Tanuki_Fix1
- IS_9.6_Tomcat6_Fix1
- IS_9.7_AssetPublisher_Fix1
- IS_9.7_Core_Fix2
- IS_9.7_Portal_Fix2
- IS_9.7_Tomcat6_Fix1
- IS_9.7_WmCloud_Fix1
- WAR_8.2_SP2_Fix6
- WAR_9.0_SP1_Fix2
- WAR_9.5_SP1_Fix4
- WAR_9.6_Fix3
- WFF_9.0_SP1_Fix2
- WFF_9.5_SP1_Fix3
- WFF_9.6_Fix1

5.0 Other Resolved Issues

This section lists the issues that were resolved in each release but were not part of the fixes listed in the previous section. A release is listed in this section only if changes occurred in that release.

Release 10.2

- PIEAR-1052
In webMethods Adapter for SAP, the connection is not enabled, as the script file was not loaded.

The issue is resolved . Now the connection is enabled and ART retains the backward compatibility of the adapters which are referring to the java scripts with the extension .js.txt.

- **PIEAR-1056**

In webMethods Adapter for Runtime, an error occurs when you view the listeners in Integration Server Administrator .

This issue occurs when the listener is not loaded in the Integration Server which is referred by one of the notifications.

This issue is now resolved.

Release 10.1

- **PIE-45041**

The Integration Server server log contains the active Session ID even though masking of Session IDs is enabled.

The Integration Server server log contains the active Session ID when the maskSession ID is set to “true” and logging level is set to “Trace”. The active Session ID is mentioned even though masking of Session IDs is enabled.

This issue is now resolved.

- **PIE-45064**

Moving or renaming folders containing flow services results in broken references.

When a folder containing a flow service is moved or renamed, any document references present in the flow service are not correctly updated. This results in the flow service containing broken references.

The issue is now resolved.

- **PIE-45087**

Designer does not honor the protocol buffer encoding type for a publishable document type created in an earlier version of Integration Server.

When using Designer 9.10 or higher to view or edit a publishable document type created on an Integration Server version 9.10 or earlier, Designer may display the IData as the Encoding type even though the assigned encoding type is protocol buffers. Additionally, if the Encoding type is changed to Protocol buffers, Designer reverts the Encoding type to IData upon save.

This issue is now resolved.

- **PIE-45125**

Java services that reference Ehcache classes directly cannot be compiled.

This issue is resolved.

- **PIE-45131**

Issues with configuring a restV2 resource.

When configuring a restV2 resource, Integration Server does not display any error on specifying a URL format containing only dynamic parameters for a REST resource. This is incorrect because a URL format must include at least one static parameter.

Integration Server throws an exception on specifying a URL format with only dynamic parameters.

- PIE-45172

From Command Central, Integration Server instance creation using a YAML template and specifying the dbtype in lower case prevents any update to Audit Logging Destination. When using Command Central to create an Integration Server instance using a YAML template and specifying the dbtype in lowercase prevents any attempt to change the Audit Logging Destination to 'Database' for an Audit Logger in the Edit Session Logger Details page (Settings> Logging> Edit Session Logger Details). Additionally, the following error occurs on changing the Session Logger: RESTART is required due to inability to update in Integration Server. This issue is resolved. Now, the dbtype is case insensitive in the YAML template, allowing the choice of "Database" as the destination when the ISCoreAudit Functional Pool is set to a valid JDBC pool.
- PIE-45177

When using Command Central to create an Integration Server instance using a YAML template, specifying the dbtype in lower case causes the functional description to be unavailable. When creating an Integration Server instance with Command Central, using YAML template and specifying the dbtype in lowercase causes the functional description of functional alias to be unavailable for viewing or editing on the Settings > JDBC Pools > Functional Definitions page. This issue is resolved. Now, the dbtype is case insensitive in the YAML template and hence, the function description appears correctly.
- PIE-45320

Unable to add IPv6 address in Integration Server to access a port. While updating the IP access of a port in the Integration Server Administrator on the Security > Ports > IP Access screen, Integration Server does not allow a user to add an Ipv6 address in the allow/deny lists of IP address. This issue is now resolved. Integration Server now allows an IPv6 address in the allow/deny lists.
- PIE-45366

Searching for Integration Server elements results in a NullPointerException. Searching for Integration Server elements using the Service Development perspective of Designer in a package for which the manifest.v3 file has the system_package attribute missing results in a NullPointerException. This issue is now resolved.
- PIE-45543

Executing the is_container script with the createLeanDockerfile option creates a Docker image that does not include the WmCloud package. This issue is now resolved.
- PIE-45588

When using a stand-alone Java client created for Integration Server, the client ignores client certificates when establishing an outbound HTTP connection using JSSE. When a stand-alone Java client uses the Integration Server method `com.wm.app.b2b.client.BaseContext.setSSLCertificates()` to provide client certificates for the purpose of establishing an SSL connection using JSSE, the client did not provide the destination HTTP

server with the client certificates needed to establish the connection.

This issue is now resolved.

- PIE-45797
Java service compilation fails with an error if the service uses the Common Directory Services API. If a Java service uses the Common Directory Service API, compilation of the Java service fails with the following error:
class file for com.webmethods.sc.LocalizedException not found
This issue is now resolved.
- PIE-46024
Integration Server now supports migration when source and target version are same. The migration utility can now be used for migration where the source and target versions of Integration Server are the same. This can be useful for data center or machine moves.
- PIE-46407
Unable to change the logging level for Integration Server by using composite templates in Command Central.
When using a composite template in Command Central to change the logging level for an Integration Server, Command Central throws a ConfigurationValidationException and the logging level does not change.
This issue is now resolved.
- PIE-46654
The Integration Server migration utility does not migrate the quiesce port settings.
This issue is now resolved.
- PIEAR-1002
In webMethods Adapter Runtime 10.1, when you create an adapter connection, the Pool Increment Size value in connection management property cannot be greater than Maximum Pool Size value. This issue is resolved. Pool Increment Size is now validated with Maximum Pool Size value.
- PIEAR-1004
In webMethods Adapter Runtime 10.1, the connection pool reset does not happen when adapter connection exception occurs in the design time while creating or configuring the adapter services.
This issue is now resolved.
- PIEAR-1012
In webMethods Adapter Runtime 10.1, when you navigate to the Connections screen from the About screen, the left navigation pane of the Connections screen still shows the information related to About screen.
This issue is now resolved.

Release 10.0

- WFF-259
An exception causes file polling ports to automatically disable.

When processing large number of files, the file polling port is automatically disabled and the following exception is displayed in Integration Server Administrator.

"java.lang.IllegalArgumentException: Comparison method violates its general contract!"

The disabled port can only be enabled by reloading the package.

This issue occurs because of the change in the sorting algorithm in JDK 1.7.

This issue is resolved.

- WFF-261

In webMethods Integration Server, the pub.flatFile:convertToString service eliminates the values or fields that contain only white spaces which results in incorrect outputs.

This issue is resolved.

Release 9.12

- PIE-39870 (IS_9.0_SP1_Core_Fix14, IS_9.5_SP1_Core_Fix11, IS_9.9_Core_Fix2, IS_9.10_Core_Fix1)
Service fails to run automatically after the cache expires when the Prefetch parameter is set to True. When the Prefetch parameter is set to True and the cache expires, the service fails to run automatically. The following exception is displayed in the server logs. "[ISS.0086.9249] Missing Parameter:num1" .

This issue is resolved.

- PIE-39177 (IS_9.0_SP1_Core_Fix14, IS_9.9_Core_Fix2)

Integration Server logs contain duplicate entries after a port is enabled and disabled.

When a port is enabled and disabled, Integration Server creates two entries for each action in the server logs. This issue occurs if the logging level is set to Debug. Integration Server logs the same information in the Debug log and the Info log which causes duplicates.

This issue is resolved. Now, if the logging level is set to Debug Integration Server logs information related to enabling and disabling of ports in the Info log.

- PIE-36924 (IS_9.5_SP1_Core_Fix11)

An additional prefix is included in an XML document input when restored from the pipeline.

Integration Server includes an additional prefix dx: at the root of an XML document instance.

This issue occurs when the XML document is updated as specified in the following sequence of steps:

(1) The XML document is provided as input to an operation in a provider web service descriptor.

(2) The document is saved to a pipeline.

(3) The pipeline is restored.

The issue is now resolved.

- PIE-39114 (IS_9.7_Core_Fix8, IS_9.9_Core_Fix2)

Discrepancy in session count on Server > Statistics page and Server > Statistics > Session page.

In Integration Server Administrator, the current total sessions displayed on the Server > Statistics page does not match the number of rows for Current Sessions displayed on the Server > Statistics > Sessions page.

This issue is now resolved.

- **PIEAR-746 (IS_9.7_Core_Fix8)**
 The threads waiting for the connection and the threads releasing the connection from the connectionPool are blocked in the ConnectionPool.
 When the connection creation thread acquires a lock on the ConnectionPool, the thread goes to the wait state and the state of this thread does not change. This prevents the connection from being released.
 To resolve this issue, Integration Sever now includes the following server configuration parameter:
`watt.server.jca.connectionPool.createConnection.interrupt.waitTime` Specifies the wait time, measured in milliseconds, that elapses before Integration Server interrupts a connection creation thread that is in a wait state. There is no default value. You must restart Integration Sever for changes to this parameter to take effect.
- **PIE-39355 (IS_9.7_Core_Fix8)**
 Integration Server does not indicate when it is disconnected from a cluster.
 Integration Server does not log an error message when it is disconnected from a cluster, preventing automatic detection of the situation.
 Now, Integration Server logs the following error message when it is disconnected from the cluster.
 [ISS.0033.151] The cluster is now not operational.
 Additionally, Integration Server logs the following error message when it rejoins the cluster.
 [ISS.0033.152] The cluster is now operational.
- **PIE-34518 (IS_9.8_Core_Fix5)**
 Addition of CDATA block support to outbound SOAP processing.
 When processing an outbound SOAP message, Integration Server ignores CDATA delimiters in String fields.
 If a String field contained text in a CDATA block, Integration Server treats the text as regular text instead of character text and url-encodes special characters in the delimiters and in the text block.
 Integration Server now provides CDATA block support for processing of outbound SOAP messages only when Integration Server hosts the web service provider. When a service used as an operation in a web service provider returns String values containing CDATA blocks, when encoding the IData into a SOAP message, Integration Server places the CDATA text in the outbound SOAP message in a CDATA section and does not url-encode special characters in the delimiters or text block.
 Keep the following information in mind:
 - A CDATA block begins with `<![CDATA[` and ends with `]]>` - Multiple CDATA blocks may be used in a single String value.
 - CDATA blocks may not overlap or be nested.
 Note: When Integration Server is acting as the web service client, Integration Server does not provide CDATA block support when processing outbound SOAP messages. If a String value containing the request is passed to the web service connector and the string contains CDATA, the contents of CDATA block will be treated as regular text and special characters in the delimiters and text block will be url-encoded in the outbound SOAP request.
- **PIE-35132 (IS_9.8_Core_Fix5)**
 Changes to Integration Server to address the security vulnerabilities identified during internal security testing.

This fix resolves the security issues found during internal security testing.

- PIE-38427 (IS_9.8_Core_Fix5)

A NullPointerException occurs when modifying or saving a flow service if dependency checking features are disabled.

A NullPointerException occurs in Software AG Designer when modifying or saving a flow service if the server configuration parameter `watt.server.ns.dependencyManager` is set to false.

This issue is resolved.

- PIE-38649 (IS_9.8_Core_Fix5)

Host name of an email port is displayed incorrectly in the View Email Client Details page in Integration Server Administrator.

The host name of an email port is incorrect and contains garbled characters when the email port is viewed in the Security > Ports > View Email Client Details page in Integration Server Administrator.

This issue is resolved. The host name is now displayed correctly in the Ports > View Email Client Details page in Integration Server Administrator.

- PIE-38920 (IS_9.8_Core_Fix5)

Certification of Integration Server with MySQL version 5.6.

Integration Server 9.8 is now certified for use with MySQL version 5.6

- PIE-39009 (IS_9.9_Core_Fix2)

Messages are rolled back to Universal Messaging incorrectly if the webMethods messaging trigger is configured to Suspend and Retry Later.

If a webMethods messaging trigger specifies Suspend and Retry Later for On Retry Failure, Integration Server rolls back a message to Universal Messaging if a transient error causes the trigger to fail. However, instead of rolling back a single message, Integration Server rolls back all of the received but unacknowledged messages for the trigger. For concurrent triggers, this can cause messages to be reprocessed. For serial triggers, this can cause messages to be lost.

Now, when a webMethods messaging trigger specifies Suspend and Retry Later and a transient error prevents the trigger from executing successfully, Integration Server rolls back a single message at a time to Universal Messaging. Additionally, Integration Server clears the trigger queue on the Integration Server immediately after suspending the trigger. Clearing the queue removes any messages that the trigger received but did not process. These messages will be redelivered.

- PIE-39287 (IS_9.9_Core_Fix2)

Integration Server displays the predefined Client Prefix instead of the user-defined Client Prefix after a restart.

After restarting Integration Server, the user-defined Client Prefix for a new Broker Connection Alias is replaced with the predefined Client Prefix that Integration Server creates.

This issue is resolved. Now, Integration Server displays the user-defined Client Prefix correctly after a restart.

- PIE-39295 (IS_9.9_Core_Fix2)

An attempt to create a web service descriptor from a WSDL document fails with a NullPointerException if the WSDL document contains multiple services but the services are not associated with all of the binding and portType definitions in the WSDL document.

Creating a consumer web service descriptor or a WSDL first provider web service descriptor from a WSDL document fails with a `NullPointerException` if the WSDL document defines multiple services and every service is not associated with every binding and portType defined in the WSDL document.

This issue is resolved. Each web service descriptor generated from a service in a WSDL document exposes only the binding and portTypes associated with the service.

- PIE-40157 (IS_9.10_Core_Fix1)

Correction to a spelling error in Queue Provider field.

On the Settings > Logging > Edit <loggerName> Details page, Integration Server Administrator lists Universal Messaging instead of Universal Messaging as one of the Queue Provider values.

The spelling error is now corrected.

- PIE-37272

Deadlock among threads in XA recovery store prevents XA transactions from completing.

The XA recovery store contains information about each XA transaction, including the transaction ID (XID), the global state of the transaction at each point in the transaction, and the state of each resources participating in the transaction. During an XA transaction, a deadlock between threads accessing the XA recovery store occurred because an exception occurred after a thread acquired a lock. The exception prevented the release of the lock.

This issue is now resolved. In addition to resolving the issue described above, Integration Server now includes server configuration parameters to control recovery logging.

`watt.server.transaction.xastore.performXALogging`

Specifies whether or not Integration Server writes transaction information to the XA recovery store.

Set to true to instruct Integration Server to log information about the state and progress of each XA transaction. Set to false to instruct Integration Server to skip logging XA transaction information.

The default is true.

Important! If you set `watt.server.transaction.xastore.performXALogging` to false, Integration Server does not log any information to the XA recovery story while processing a transaction, making transaction recovery impossible. If you want Integration Server to automatically resolve incomplete transactions or you want to manually resolve incomplete transactions, Integration Server must perform XA logging.

You must restart Integration Server for changes to this parameter to take effect.

`watt.server.transaction.xastore.maxTxnPerFile`

Specifies the maximum number of unique XA transactions in an XA recovery log file. When the XA recovery log file reaches the maximum number of transactions, Integration Server creates a new file. The default is 2000 transactions.

Consider increasing the maximum number of unique XA transactions for the XA recovery log file if there are more than 2000 active XA transactions and Integration Server exhibits a performance delay due to input/output. Increasing the number of unique XA transactions allowed per file decreases the number of files used for the XA recovery log, which, in turn, may result in fewer files for Integration Server to search when performing XA recovery.

Decreasing the number of unique XA transactions stored in file may help during transaction recovery as it might decrease the time to read and consolidate open transactions.

- PIE-37566

Executing anonymous services creates persistent sessions.

Some anonymous services in the WmRoot package create persistent sessions. Repeated execution of these services can consume all of Integration Server's licensed sessions.

This has been fixed. Anonymous services in WmRoot are now stateless which means that Integration Server creates sessions to execute these services and discards the sessions immediately

- PIE-38112

Integration Server does not restart after improper shut down.

If Integration Server does not shut down properly, which can be caused by a JVM crash or a machine crash, configuration files such as `acls*.cnf` become corrupted or are reduced to a size of zero. In turn, this causes a restart of Integration Server to fail. To recover, data must be restored manually from a backup directory. This issue results from how Integration Server persists data into the file system. Prior to this fix, Integration Server tried to overwrite the existing file in the file system. If Integration Server shut down improperly right after the file was recreated but before the new content was written to the file system, the file content became corrupted or the file was reduced to a size of zero.

With this fix, Integration Server changes how it saves changes to configuration files. When saving changes to configuration files, Integration Server first saves the configuration changes in a temporary file in the `IntegrationServer_directory/instances/instanceName/config/work` directory. After saving the changes in a temporary file, Integration Server moves the temporary file to the actual configuration file. This ensures that if unexpected behavior occurs before the configuration changes are initially saved to the temporary file, only the temporary file is impacted. The actual configuration file is not corrupted. At start up, if Integration Server detects files in in the `IntegrationServer_directory/instances/instanceName/config/work`, it suggests that changes to a configuration file were not saved to the temporary file and that, therefore, the changes were not made to the actual configuration file. Use the contents of the `IntegrationServer_directory/instances/instanceName/config/work` directory to determine which configuration files were in the process of being changed. Decide whether or not you want to redo the changes to the configuration files. Delete the files under the directory and redo the configuration change if you so choose.

- PIE-38470

The `xsd:any` element and attributes are not generated for a web service descriptor when the `Allow Unspecified Fields` property is true.

The WSDL document for a web service descriptor does not include the `xsd:any` element and attributes when the web service descriptor uses a `Document` variable for which the `Allow unspecified fields` property is set to true. This issue occurs even when the server configuration parameter `watt.core.schema.createSchema.omitXSDAny` is set to false.

To address this issue, the web service descriptor now includes a property, `Omit xsd:any` from WSDL. When set to false, the WSDL document generated by Integration Server includes an `xsd:any` element. When set to true, the WSDL document does not include the `xsd:any` element. The default value is true. For changes to this property to take effect, after saving the change to the property, either refresh the web service descriptor or reload the package that contains the web service descriptor.

- PIE-38658

When a client sends multiple HTTP PUT requests to Integration Server, every second request fails with a 400 Bad Request response.

Now, alternating HTTP PUT requests from the same client will not fail. Integration Server handles multiple HTTP PUT request from the same client correctly.

- PIE-38686

GET request mandatorily adds a default value to the Content-Type header.

If Content-Type header is not specified in a GET request, the `pub.client.http` service adds a default Content-Type: "application/x-www-form-urlencoded". According to HTTP 1.1, it is not mandatory to specify the Content-Type header. If the Content-Type header is not specified, the `pub.client.http` service should not add a default Content-Type.

Now, if no Content-Type header is specified, `pub.client.http` service does not add a default Content-Type to the request. However, if you pass the data using 'args' or 'table' keys, default Content-Type: "application/x-www-form-urlencoded" is added.

- PIE-38751

In Integration Server, sometimes publishing a guaranteed document to Universal Messaging server fails due to transaction failure.

This issue occurs when the Universal Messaging server does not respond to the transaction within the specified `EvenTimeout`.

This issue is resolved. A new server configuration property,

`watt.server.um.producer.transaction.commitRetryCount` is introduced in the server configuration file. This property specifies the number of retry attempts Integration Server makes to publish the message to the Universal Messaging server. The maximum number of retries is 9. If you try to assign a value greater than 9, Integration Server automatically sets the value of the property to 9.

The default is 0. When setting a retry value, you must ensure that the total transaction time does not exceed the `MaxTransactionTime`. The total transaction time is calculated by multiplying the total number of attempts with the `EvenTimeout`. For example, if the retry value is set to 9, and the `EvenTimeout` is set to 60s, the total transaction time is $60,000(9+1) = 600s$.

- PIE-38932

Integration Server loads a package more than once during deployment.

During deployment of multiple packages at the same time, Integration Server reloads the packages more than once.

Now, when a deployment operation includes multiple packages, Integration Server loads each package only once.

- PIE-39017

In a clustered environment, Integration Server sometimes does not create child tasks when a new server is added to the cluster or when an existing server is restarted.

When a task is scheduled to run on all servers in a clustered environment, Integration Server creates a parent task and a child task for each server in the cluster. When a new server is added to the cluster or when an existing server in the cluster is restarted, Integration Server creates a corresponding child task upon server restart. However, Integration Server sometimes does not create the child task for the newly added server or for the server that was restarted. As a result, the complete information for all servers in the cluster is not available on the Scheduler screen.

This issue is resolved.

- **PIE-39164**
SSL handshake fails as no client certificate is sent when a new connection is opened through proxy. When Integration Server tries to open a connection to web server that requires client certificate through proxy, the SSL handshake using JSSE fails as no client certificate is sent. This happens only when the `keyStoreAlias` and `keyAlias` parameters are set using `pub.security.keystore:setKeyAndChain` service and not when they are set using the Integration Server Administrator (Security > Certificates). This issue occurred because the client certificate attachment was missing in the request.
This issue is now resolved. Now, client certificate is sent to web server when a connection is opened through proxy.
- **PIE-39178**
Repository-based deployment fails when deploying locally publishable documents on the target Integration Server.
When deploying locally publishable documents on the target Integration Server using a repository-based deployment, Integration Server cannot find a messaging connection alias for the locally publishable document. This causes the deployment to fail. When the publishable document type is set to locally only, there is no messaging connection alias attached to it.
This issue is resolved. Now, Integration Server does not check for a messaging connection alias during deployment when the publishable document type in Integration Server is set to locally only.
- **PIE-39210**
Integration Server experiences memory leak due to orphaned entries of type `com.wm.app.b2b.server.ostore.FileCache$InvokeCounter` located in `com.wm.app.b2b.server.ostore.FileCache`.
This issue is now resolved.
- **PIE-39268**
Enhancement to Integration Server to use a file for specifying allowed cipher suites.
The following server configuration properties identify the list of cipher suites for use with inbound and outbound SSL connections.
`watt.net.ssl.client.cipherSuiteList`
`watt.net.ssl.server.cipherSuiteList`
`watt.net.jsse.client.enabledCipherSuiteList`
`watt.net.jsse.server.enabledCipherSuiteList`
Prior to this enhancement, the properties accepted a comma-separated list of cipher suites. However, specifying a long list of cipher suites can be cumbersome. To make it easier to specify a long list of cipher suites, Integration Server now allows specifying a file as the value for the cipher suite server configuration properties. In the file, specify each cipher suite on a different line. For each cipher suite server configuration property for which you want to specify a file instead of a list of cipher suites, specify the following as the value of the property: `file:directoryName\filename` For example: `watt.net.jsse.server.enabledCipherSuiteList=file:c:\ssl\ciphers.txt`.
Note: You can set the value of a cipher suite server configuration property to a comma-separated list, default, or the absolute path to a file. You cannot specify a combination of these. Integration

Server loads the file and its list of supported cipher suites at start up. Changes to the contents of the file that are made after Integration Server starts will not take effect until the next time Integration Server starts. Integration Server throws the following exception if it cannot find the specified file: ISS.0070.9048E Integration Server cannot find the file {0} specified for the server configuration parameter {1}.

Cause The specified server configuration parameter uses a file to identify the allowed cipher suites; however, Integration Server cannot find the file.

Action Make sure the server configuration parameter specifies the correct location of an existing file.

- PIE-39278

Message not found for messageKey 68.29 and 68.30.

If an email port has "Log out after each mail check=Yes" , Integration Server fails to retrieve the error messages for messageKey 68.28 and messageKey 68.30.

Following messages appear in the server log:

For IMAP: ISP.0068.0029I] Message not found for messageKey 68.29. For POP3: ISP.0068.0029I] Message not found for messageKey 68.30.

This issue is now resolved. Now, correct messages are displayed for messageKey 68.28 and messageKey 68.30.

- PIE-39298

Error in getting SFTP server host key, if a key exchange algorithm is not supported by SFTP server. Jsch has a default key exchange algorithm order.

During the handshake, jsch checks the client key exchange algorithms one by one with SFTP server key-exchange algorithms. The first matching algorithm is used as the key-exchange algorithm between SFTP client and SFTP server. However, Jsch does not support 2048-bit keys for diffie-hellman-group-exchange-sha256 and diffie-hellman-group-exchange-sha1 key exchange algorithms in Java 1.7 and earlier versions. Consequently, if the Integration Server runs with Java 1.7 or earlier version and SFTP server expects 2048-bit keys for these algorithms, then the handshake between SFTP server and SFTP client fails.

To avoid this issue, the order of these key exchange can be changed using the watt.ssh.jsch.kex parameter so that, any other matching key exchange algorithm can be selected as the key exchange algorithm between the SFTP client and SFTP server.

To address this issue, Integration Server now include a server configuration parameter to change the order of the key exchange algorithm.

watt.ssh.jsch.kex Specifies the order of the key exchange algorithm for Jsch. The specified order overrides the default key exchange algorithms order supported by Jsch.

For example: watt.ssh.jsch.kex=diffie-hellman-group-exchange-sha1, diffie-hellman-group1-sha1,diffie-hellman-group14-sha1 If the SFTP server expects 2048-bit keys for diffie-hellman-group-exchange-sha1, then the order of this algorithm can be changed so that, any other matching algorithm can be selected as key exchange algorithm between the SFTP client and SFTP server.

watt.ssh.jsch.kex=diffie-hellman-group1-sha1, diffie-hellman-group14-sha1, diffie-hellman-group-exchange-sha1 Integration Server must be restarted for changes to this parameter to take effect.

- PIE-39357

Integration Server fails to retrieve the SFTP Get Host Key during the SFTP Server Alias creation.

SFTP Server Alias creation fails with 'Auth Cancel' exception, as Integration Server fails to retrieve the SFTP Get Host Key.

The issue is now resolved.

- PIE-39380
Integration Server issues an Access Denied error message during the logon process and logs an authentication exception in the server log.
Integration Server issues the following error message during the logon process and logs the exception in the server log:
Access to Integration Server is denied because of the following authentication exception:
LoginModule "com.wm.app.b2b.server.auth.jaas.X509LoginModule" unavailable.
The issue is now resolved.
- PIE-39574
Web service connector does not capture all the Set-Cookie HTTP response headers sent back by the web service provider.
When a web service connector sends a request to a web service provider that sends back multiple Set-Cookie response headers, the connector captures only one Set-Cookie response header.
This issue is resolved. Now, if the web service provider sends back multiple Set-Cookie HTTP response headers, the connector returns all of the headers in its output.
- PIE-39585
Missing fields in an IS document type generated from an XML Schema Definition (XSD) that has complex type definitions referring to group elements defined later in the schema.
When an Integration Server document type is generated from an XSD where the complex type definitions contain references to group elements defined later in the schema, the generated document type does not contain the fields corresponding to such references.
This issue is now resolved.
- PIE-39701
Integration Server migration utility reports an exception during migration of Trading Networks package.
During migration of Integration Server, if the source Integration Server contains a Trading Networks package, the migration utility throws an exception if JDBC pool information was not selected during migration of configuration files and JDBC pool information was specified during installation of Trading Networks.
Trading Networks data and configuration file Migrator : Unable to locate the database configuration information. File
<IntegrationServer_installDirectory>\bin\migrate\..\..\instances\default\config\jdbc\pool\<poolName>.xml does not exist. Where poolName refers to the pool name specified for the Trading Networks functional alias.
This issue is now resolved. Integration Server migration utility correctly migrates Trading Networks data.
- PIE-39735
Synchronization of publishable documents with the messaging provider fails when using a startup

service to run `pub.publish:syncToProvider`.

When the `pub.publish:syncToProvider` service runs as part of a startup service, Integration Server reloads the `webMethods` messaging triggers but fails to update the state of the triggers.

This issue is resolved.

- **PIE-39762**
REST API descriptor treats parameter of type `byte[]` as `byte`.
When creating a REST API descriptor, if a selected REST resource contains a service with `byte[]` in the service signature, the REST API descriptor treats the parameter as a `byte` instead of `byte[]`.
This issue is now resolved.
- **PIE-39819**
An error occurs when you create or edit a file polling port if the path to the monitoring directory contains a backslash `'\'`.
When you create or edit a file polling port and the path to the monitoring directory contains a backslash `'\'`, the following error message is displayed. "No listener defined with the specified key: FilePollingListener: c: mp" This issue occurs because the Integration Server Administrator uses the wrong `encodingType`.
This issue is resolved.
- **PIE-39825**
A synchronous or asynchronous publish and wait does not receive a reply.
If a publishable document type is empty, that is, it does not contain any fields besides the `_env` field, and values are mapped to fields the `_env` field before publishing the document using `pub.publish:publishAndWait`, Integration Server does not receive a reply to the published request. This occurs for synchronous and asynchronous publish and wait requests.
This issue is now resolved.
- **PIE-39839**
Universal Messaging does not reuse the session on Integration Server even though `watt.server.trigger.reuseSession` is set to `true`.
When the server configuration property, `watt.server.trigger.reuseSession` is set to `true` and Broker is the messaging provider, Broker reuses the session on Integration Server. Whereas, Universal Messaging as the messaging provider does not reuse the session.
This issue is resolved.
- **PIE-39847**
JDBC connection pools are blocked after the database restarts.
After the database used by a JDBC pool restarts, Integration Server does not release connections from the pool because of an unexpected exception thrown by the JDBC driver.
Now, Integration Server exception handling prevents the exception thrown by the JDBC driver from blocking the release of connections from the JDBC pool.
- **PIE-39848**
After migrating to Integration Server 9.5 or later, Integration Server displays an exception when calling a web service and using MTOM streaming.
After migrating 2 to Integration Server 9.5 or later, sending a web service request that uses MTOM

streaming for which more than one chunk is sent, the following exception occurs: "Exception --> org.apache.axis2.AxisFault: Connection reset by peer: socket write error"
This issue is now resolved.

- PIE-39853
Unable to view assigned input values for services in Integration Server Administrator when the server configuration property `watt.server.disableXSSFilter=true`.
When creating a scheduled task, assigning input values using the "Assign Inputs" button displays an empty input field even though the assigned value is saved in the input field. Since the XSS filter is disabled by setting the server configuration property `watt.server.disableXSSFilter=true`, Integration Server is unable to encrypt quotation marks (") in the input values. Therefore, the assigned input values are not displayed in the input field in Integration Server Administrator. This issue is resolved. Integration Server is now able to encrypt quotation marks (") in input values when the server configuration property `watt.server.disableXSSFilter=true`.
- PIE-39876
Swagger document generated by Integration Server does not represent REST resources with required list parameters correctly.
If a REST API descriptor contains REST resources that refer to a list parameter, such as String list, Document list, Document Reference list, and Object list, and the list parameter is required, the definitions section of the generated Swagger document does not indicate that the parameter is required.
This issue is now resolved.
- PIE-39877
If a REST API descriptor contains a REST resource that refers to a parameter of type String table, the generated Swagger document does not indicate the type of the parameter as "array".
This issue is now resolved.
- PIE-39938
Reliable messaging sequences on Integration Server are timed out when the inactivity timeout interval is set to a value of -1.
This issue is now resolved. The reliable messaging sequences do not get timed out when the inactivity timeout interval is set to -1 in the Integration Server Administrator (Settings > Web Services > Reliable Messaging > Edit Configuration).
- PIE-39967
Service error statistics counter causes memory leak.
An internal mechanism used to report the last 60 seconds worth of service failures has a memory leak. The memory leak occurs when a steady stream of exceptions, at least one every 60 seconds, are thrown without interruption.
To resolve the memory leak, Integration Server clears the service error statistics counter when new entries are added.
- PIE-40014
Selected Authentication Type does not get populated under HTTP Transport Properties in consumer web service descriptor endpoint alias.

When creating a consumer web service descriptor endpoint alias, the selected Authentication Type is not populated under HTTP Transport Properties.

The issue is now resolved.

- PIE-40027
Integration Server logs the wrong message when it cannot find the encoder "esapi".
When Integration Server cannot find the encoder "esapi", it logs the following wrong error message: [ISC.0072.0023E] Couldnt initialize encoder "{0}".
Whereas, Integration Server should log the following message:
ISC.0072.0023E] Failed to initialize the bundle for encoder 'esapi'.
This issue is resolved now.
- PIE-40048
In the Service Usage screen, some of the values are displayed as instead of the actual value.
In the Service Usage screen of Integration Server Administrator, some of the values in the Count and Last Run columns are displayed as instead of the actual value. This issue occurs because of encoding issues in Integration Server.
This issue is resolved. Now, the values in the Count and Last Run columns are displayed correctly.
- PIE-40109
The pub.xml:getNextXMLNode service incorrectly returns a NullPointerException when the NodeIterator does not have XML nodes to return.
This issue is resolved. The pub.xml:getNextXMLNode service returns null when there are no XML nodes to return.
- PIE-40126
Results window does not show parameter names when using Natural CALC subprogram.
From the Natural CALC subprogram, the IDL and IS connection are created. When using Integration Server and Adapter 9.9, and running the example CALC flow service in Software Designer from the Service Development perspective, the results window does not show the parameter names. The field name abbreviator (as used for XmlData) truncates field names that start or end with #.
With this fix, the Field name abbreviator no longer truncates field names that start or end with #.
- PIE-40139
Integration Server becomes unresponsive due to an OutOfMemoryError caused by lack of PermGen space. A memory leak in the package classloader may result in a java.lang.OutOfMemoryError: PermGen space, which causes Integration Server to become unresponsive.
The memory leak in the package classloader is now repaired.
- PIE-40141
The pub.event.routing* services are not working properly in Integration Server .
The pub.event.routing* services of send, subscribe, and unsubscribe are not working properly. For example, the send service throws the following ServiceException: Could not run 'send'
com.wm.app.b2b.server.ServiceException: com.wm.app.b2b.server.ServiceException: Event Routing service not available.

This issue is resolved now.

- PIE-40183
Different error messages for primary and regular HTTP ports.
When Integration Server receives a client request for a web service that does not exist, it returns different error messages depending on whether the request was received through the primary port or an HTTP port.
Now, Integration Server returns the same error message regardless of the port through which it receives the request.
- PIE-40189
Missing schema definition for the envelope (`_env`) field in the WSDL document when a service containing reference to a publishable document type is exposed as a web service.
When a service containing reference to a publishable document type is exposed as a web service, the resulting WSDL document does not contain the schema definition for the envelope (`_env`) field.
This issue is now resolved.
- PIE-40194
A webMethods messaging trigger that receives messages from a Universal Messaging cluster becomes disconnected and does not reconnect.
When a Universal Messaging connection alias that specifies a cluster of Universal Messaging servers and the network between the Universal Messaging servers fails, one or more webMethods messaging triggers that use the alias might disconnect and then not reconnect. This might happen even if the trigger can successfully connect to one of the Universal Messaging servers but not the other servers in the cluster.
Now, if a webMethods messaging trigger becomes disconnected from a cluster of Universal Messaging servers, the trigger reconnects only when the cluster has a quorum. In addition to resolving the issue described above, Integration Server now redirects the Universal Messaging client log entries from the default stdout to the Integration Server's log directory, `Integration Server_directory/instances/instanceName/logs/umClient.log`. The `umClient.log` file can be used when debugging Universal Messaging issues.
- PIE-40195
Integration Server does not validate a Kerberos ticket sent by an Internet browser.
If Integration Server receives a request on a port configured to accept Kerberos tickets, Integration Server returns a 401 status code with a "Negotiate" header challenge to the Internet browser. Upon receiving the Negotiate header challenge, a properly configured Internet browser generates a Kerberos ticket and sends the request, along with the Kerberos ticket, to Integration Server.
Integration Server should validate the Kerberos ticket and either allow or deny the client request based on the ACL settings. However, Integration Server could not process the Kerberos ticket because a Kerberos ticket sent by an Internet browser is structurally different from tickets generated by a Java client. Because Integration Server could not process the ticket, Integration Server could not validate the Kerberos ticket. Instead, Integration Server returned a 401 status code and prompted the client for username and password credentials.
Integration Server can now process a Kerberos ticket generated by an Internet browser.

- PIE-40202
 JDBC connection pool for ISInternal functional alias is blocked.
 Integration Server does not release connections from the JDBC connection pool used by the ISInternal functional alias because of an unexpected exception thrown by the JDBC driver.
 Now, Integration Server exception handling prevents the exception thrown by the JDBC driver from blocking the release of connections from the JDBC connection pool.
- PIE-40212
 Enhancement for using a public cache, including a distributed cache, for service results caching. Previously, Integration Server stored cached service results in the ServiceResults system cache which is part of the SoftwareAG.IS.Services system cache manager.
 With this enhancement, a local or distributed public cache can now be used for service results caching. By using a distributed cache as the service results cache, multiple Integration Servers can access cached service results.
 Keep the following information in mind when using a public cache for service results caching:

 - The cache must be configured such that elements do not expire once they are placed in the cache. That is, the Eternal check box must be selected for the cache. The duration of cached service results depends on the Cache expire property value for the service and the `watt.sever.cache.flushMins` server configuration parameter value.
 - When using a distributed cache as a service results cache, make sure that all the Integration Servers that share the service results cache have the same packages.
 - When returning cached results for a service, by default, Integration server returns a reference to the cached results instead of the actual value of the cached results. If you want Integration Server to return the actual value instead of a reference, make sure that the Copy on Read and Copy on Write check boxes are selected for the public cache.
 - If you want to be able to use Integration Server Administrator or the `pub.cache.serviceResults:listCache` service to view the cached elements for a service, make sure that the service results cache is searchable and that the cache allows automatic indexing for keys. That is, for the service results cache, the Searchable check box must be selected and the Key check box next to Allow Automatic Indexing must be selected.
 - When a public cache used for service results caching is disabled or a public cache manager that contains the cache used for service results caching is shut down, Integration Server does not cache or retrieve service results. Instead of using cached results, Integration Server executes the service.
 - When a public cache used for service results caching is enabled or the public cache manager that contains the public cache used for service results is started, Integration Server re-initializes the cache.

For detailed information about creating a public cache, public cache manager, and a distributed cache, see the webMethods Integration Server Administrator's Guide.
 Note: In Integration Server Administrator, the Server > Service Usage page displays statistics about service results. For a distributed cache, this statistics corresponds to the current Integration Server instance only. Integration Server Administrator does not provide aggregated statistics for all the Integration Servers using the same distributed cache for service results caching.
 To configure Integration Server to use a public cache for service results caching, you specify the name of the cache and cache manager in the server configuration parameters `watt.server.serviceResults.cache` and `watt.server.serviceResults.cacheManger`, respectively.

watt.server.serviceResults.cache

Specifies the name of the public cache to use for service results caching. If no value is specified for the watt.server.serviceResults.cache parameter, Integration Server uses the ServiceResults cache in the cache manger specified in the watt.server.serviceResults.cacheManager parameter as the service results cache. If the cache manager in the watt.server.serviceResults.cacheManager parameter does not contain a cache named ServiceResults, Integration Server throws an error at start up and does not cache service results. You must restart Integration Server for changes to this parameter to take effect.

watt.server.serviceResults.cacheManager

Specifies the name of the public cache manager that contains the public cache to use for service results caching. If no value is specified for this parameter, Integration Server uses the SoftwareAG.IS.Services system cache manager as the service results cache manager. If the SoftwareAG.IS.Services system cache does not contain a cache with a name that matches the value of the watt.server.serviceResults cache, Integration Server throws an error at startup and does not cache service results. You must restart Integration Server for changes to this parameter to take effect.

Note: To use the ServiceResults system cache located in the SoftwareAG.IS.Services system cache manager as the service results cache, do not specify a value for watt.server.serviceResults.cache or watt.server.serviceResults.cacheManager.

- PIE-40246
The pub.client:soapClient service does not use the private key and certificate chain set using the pub.security.keystore:setKeyAndChain.
This issue is now resolved.
- PIE-40259
Using the delete icon to delete an OpenID Provider on the Security > OpenID page does not delete the OpenID Provider.
In the Security > OpenID page of Integration Server Administrator, clicking the delete icon in the Delete Provider column invokes an unknown service, resulting in the display of the following message:
[ISC.0072.9001] Unknown service: wm.server.security.openid:deleteProvider
Furthermore, the OpenID Provider is not deleted.
Now, clicking the delete icon in the Delete Provider column on the Security > OpenID page results in Integration Server deleting the OpenID Provider.
- PIE-40261
Use JSSE option is missing on external port if the internal port is set to HTTP.
On Integration Server Administrator, when you view a port for which the Enterprise Gateway External Port is set to HTTPS and Enterprise Gateway Registration Port is set to HTTP, the "Use JSSE" option does not appear. The "Use JSSE" option appears only if the Enterprise Gateway Registration Port is set to HTTPS.
This issue is now resolved.
- PIE-40275
Enhancement to allow the default locale for pub.date* services to be the server locale instead of the

client locale.

When executing a pub.date* service with a locale input parameter for which no locale is specified Integration Server uses the locale from the session used by the client that invoked the service.

Integration Server now includes a server configuration parameter that you can use to specify that the default locale for pub.date* services should be the server locale.

watt.server.session.locale.ignore

Specifies whether the default locale for the pub.date* services is the server locale or the locale from the session used by the client that invoked the service. When set to true, when executing a pub.date* service for which a locale input parameter value is not specified, Integration Server uses the server locale as the value of the locale parameter. When set to false, when executing a pub.date* service for which a locale input parameter value is not specified, Integration Server uses the locale from the session used by the client that invoked the service. The default is false.

You do not need to restart Integration Server for changes to this parameter to take effect.

- PIE-40300

Under heavy load, Integration Server might generate duplicate contextIDs for audit log messages resulting in an AuditLogManagerException.

Under heavy load, Integration Server might generate duplicate contextIDs for audit log messages, causing Integration Server to write the following AuditLogManagerException to the server.log:

[ISS.0095.9998] AuditLogManager Exception:

com.webmethods.sc.auditing.api.logger.WmAuditLoggerException: Null or duplicate context given to logger 'Error Logger'

Now Integration Server generates unique contextIDs even under heavy load.

- PIE-40324

Integration Server throws an error when an incoming message contains a pre-populated contextID and audit contextID.

Integration Server write the following error to the server.log when an incoming message contains a pre-populated context ID and audit contextID.

[BAA.0000.0061] This context has already been used in the current context stack.

Integration Server now handles a message with a pre-populated contextIDs and audit contextID and no longer logs the above error.

- PIE-40362

Unable to publish a package to multiple subscribers.

When publishing a package to multiple subscribers, clicking the 'Send Release' button in Integration Server Administrator prompts you to select a subscriber even though subscribers are selected.

This issue is resolved.

- PIE-40363

UserInfo service does not receive error information.

If an OpenID Provider's UserInfo Endpoint returns an error, information about the error should be passed to the userInfoError input variable in the UserInfo service registered for the OpenID Provider. This is not happening.

This issue is now resolved.

Note: Previously, the `userInfoError` input variable was named `error`.

- **PIE-40372**
Reply messages are sometimes not received with asynchronous request-reply using `webMethods` messaging and Universal Messaging as the messaging provider.
In an asynchronous request-reply for `webMethods` messaging, the `pub.publish:publishAndWait` service is first invoked followed by the `pub.publish:waitForReply`. When the messaging provider is Universal Messaging, some reply messages are not received if Integration Server receives the reply message before invocation of the `pub.publish:waitForReply` service.
This issue is now resolved.

- **PIE-40408**
Enhancements for managing a service results cache. Integration Server now includes built-in services that you can use to manage a service results cache, including:
`pub.cache.serviceResults:resetServerCache` resets the entire cache.
`pub.cache.serviceResults:resetServiceCache` resets the cache for a particular service.
`pub.cache.serviceResults:listServiceCache` lists the cached results for a particular service.
In addition to the above services, Integration Server Administrator now includes a way to view the cached service results for a particular service. To use Integration Server Administrator to view the cached service results, do the following:
 1. Open Integration Server Administrator if it is not already open.
 2. In the Packages menu of the Navigation Panel, click Management.
 3. In the Packages list, click the package containing the service for which you want to view cached service results.
 4. Click Browse Services in `packageName`.
 5. In the Services list, click the name of the service for which you want to view the cached service results.
 6. On the Packages > Management > `packageName` > Services > `serviceName` page, click List Service Cache. Integration Server displays a list of the cached results for the service. For each cached element, Integration Server displays the key for the cached element, the cached service inputs, and the associated cached service outputs.Notes:
 - You can also access the Packages > Management > `packageName` > Services > `serviceName` page in the following way: Under Server, click Service Usage. On the Server > Service Usage page, click the name of the service for which you want to list cached results.
 - To return a list of cached result for a service, the services results must be searchable and that the cache allows automatic indexing for keys. That is, for the service results cache, the Searchable check box must be selected and the Key check box next to Allow Automatic Indexing must be selected. Note that exposing the contents of the service results cache may represent a security risk.
 - You can view cached service results for a public service results cache only. You cannot view cached service results in the ServiceResults system cache.Below is more information about each of the new built-in services.
`pub.cache.serviceResults:listServiceCache WmPublic`.

Returns a list of the cached service results for a particular service.

Input Parameters

serviceName String. Name of the service for which you want to view the cached service results.

Output Parameters elements Document List. Conditional. A document list (IData) containing the cached service results for the specified service. The pub.cache.serviceResults:listServiceCache service returns the elements output only if the service results cache contains cached results for the specified service. For each set of cached service results, the

pub.cache.serviceResults:listServiceCache returns the following information:

key String. Optional. Key for the cached element containing the service results.

input. Document. Optional. An IData containing the key/value pairs for the cached service input.

output Document. Optional. An IData containing the key/value pairs for the cached service output

Usage Notes:

- If serviceName specifies a service that does not exist or a service for which Integration Server does not cache service results, Integration Server returns an empty elements output parameter.
- You can use the pub.cache.serviceResults:listServiceCache with a public service results cache only. You cannot use the pub.cache.serviceResults:listServiceCache with the ServiceResults system cache.
- For the pub.cache.serviceResults:listServiceCache service to return results, make sure that the service results cache is searchable and that the cache allows automatic indexing for keys. That is, for the service results cache, the Searchable check box must be selected and the Key check box next to Allow Automatic Indexing must be selected.
- Exposing the contents of the service results cache via the listCacheService may represent a security risk.

pub.cache.serviceResults:resetServerCache

WmPublic. Resets the cache for all services in the service results cache, resulting in the removal of all cached service results for all services from the service results cache.

Input Parameters None.

Output Parameters message String. Message indicating whether or not Integration Server cleared the service results cache successfully.

- "Cache reset successfully" indicates that Integration Server cleared the entire service results cache successfully.

- "Error resetting cache!" indicates that Integration Server did not clear the entire service results cache successfully. Review the sever log and error log to determine why Integration Server did not clear the service results cache successfully.

pub.cache.serviceResults:resetServiceCache

WmPublic. Resets the cache for specific service, resulting in the removal of cached service results for the service.

Input Parameters serviceName String. Fully qualified name of the service in the format folder.subfolder:serviceName for which you want to remove cached service results.

Output Parameters message String. Message indicating whether or not Integration Server cleared the service results cache for the specified service successfully.

- "Cache reset successfully" indicates that Integration Server cleared the cached service results for the specified service successfully.

- "Error resetting cache!" indicates that Integration Server did not clear the cached service results for the specified service successfully. Review the sever log and error log to determine why Integration Server did not clear the service results cache successfully.

Note: To avoid revealing whether or not a service exists, if `serviceName` specifies a service that does not exist, the message output parameter returns the message "Cache reset successfully"

- PIE-40422
Integration Server grants access to a user account disabled by the external user manager. Integration Server caches information for users who provide valid credentials. If the user later provides invalid credentials, Integration Server denies access but does not remove the cached information about the user. Consequently, if the user later provides valid credentials, Integration Server grants access even if an external user manager disabled the user account. Now, when a user provides invalid credentials, Integration Server removes the user information from the cache.
- PIE-40454
The `pub.storage:get` service fails in a cluster. Integration Server improves the performance of the `pub.storage*` services by caching the `DATASTORE_ID` values from the `IS_DATASTORE` table. However, in a cluster, a cached `DATASTORE_ID` value might become stale due to actions taken by other Integration Servers in the cluster. A `pub.storage*` service fails when it attempts to use a stale `DATASTORE_ID`. Now, when a `pub.storage*` service attempts to use a stale `DATASTORE_ID`, Integration Server replaces it with the current `DATASTORE_ID`.
- PIE-40467
Corrections for cosmetic issues in Integration Server Administrator. Integration Server Administrator displays some cosmetic issues, such as empty cells, deeply nested tables, and missing graphics, that are now repaired.
- PIE-40480
When restarting Integration Server fix after applying a fix, expected changes are not made to the embedded database. If the `ISInternal` functional alias uses the embedded, internal database and application of an Integration Server fix made changes made to the database structure, Integration Server should perform some database updates upon restart of Integration Server. However, this was not happening. This issue is now resolved.
- PIE-40504
A `NullPointerException` occurs when Integration Server enters quiesce mode and the JMS subsystem has not been initialized. This issue is resolved. Integration Server can now enter quiesce mode even if the JMS subsystem has not been initialized.
- PIE-40505
Memory leak when invoking `pub.jms:reply` Integration Server did not correctly release JMS Session and MessageProducer objects when invoking the `pub.jms:reply` service, resulting in a memory leak. The issue is now resolved.
- PIE-40516
In a cluster of Integration Servers, the scheduled task for which the target node is set to "Any Node" and "Repeat After Completion" executes the same service simultaneously on different cluster nodes when the Terracotta Server Array becomes unavailable.

This issue is now resolved. Now, in a cluster of Integration Servers, a scheduled task with target node set to "Any Node" and "Repeat After Completion", executes the service only on one of the cluster nodes at a time when the Terracotta Server Array becomes unavailable.

- **PIE-40517**
Specifying a value of true for the `watt.net.ssl.client.useJSSE` server configuration property does not use JSSE for all of the outbound HTTPS connections from Integration Server.
After applying a fix or upgrading to a release that includes PIE-38599, Integration Server no longer honors the value of `watt.net.ssl.client.useJSSE` for outbound HTTPS connections. This occurs because PIE-38599 changed the default value of the `useJSSE` input parameter in the `pub.client:http` service to "no" from a null value. A value of "yes" or "no" for the `useJSSE` input parameter overrides the value of the `watt.net.ssl.client.useJSSE` server configuration property. A null value for the `useJSSE` input parameter causes the service to defer to the value of the `watt.net.ssl.client.useJSSE` server configuration property.
Now, the default value of the `useJSSE` input parameter for the `pub.client:http` service is null, which allows the `watt.net.ssl.client.useJSSE` server configuration property to determine the default behavior for outbound HTTPS connections.
- **PIE-40548**
Performance degradation observed in Integration Server.
Integration Server exhibits performance degradation in the GA versions 9.8, 9.9, and 9.10. Integration Server versions 9.6 and 9.7 exhibit a similar performance degradation after applying the latest fix.
The cause of the apparent performance degradation is now resolved.
- **PIE-40561**
During the migration of Integration Server from 9.0 to a later release, the migration utility also migrates the server configuration parameters `watt.core.schema.validateIncomingXSDUsingXerces` and `watt.server.wsdl.validateWSDLSchemaUsingXerces`.
The migration utility need not migrate these server configuration parameters during the migration of Integration Server from 9.0 to a later release because the functionalities provided by the two parameters are replaced in releases starting with 9.0.
 - The 'Validate schemas using Xerces' option in the New Web Service Descriptor wizard replaces the functionality of the `watt.core.schema.validateIncomingXSDUsingXerces` parameter.
 - The value of the 'Validate schemas using Xerces' option is stored as the web service descriptor property 'validateSchemaUsingXerces' (which replaces the functionality of the `watt.server.wsdl.validateWSDLSchemaUsingXerces` parameter).The issue is resolved. The specified server configuration parameters are removed from Integration Server starting with release 9.0.
- **PIE-40569**
Integration Server logs a message about JDBC connection pool waiting threads threshold count being exceeding erroneously.
Integration Server should log the following message only when a thread is waiting for the JDBC connection pool to have an available connection.
[ISS.0096.0009I] JDBC Connection pool waiting threads threshold exceeded, <number> threads waiting for connection for pool <JDBCPoolName> Integration Server should not log the above message if the JDBC pool is empty or the JDBC driver threw an exception when Integration Server

attempts to create a connection. In these cases, there is not actually a waiting thread. This issue is now resolved.

- PIE-40576
SFTP transfers fail due to timestamp parsing failure.
A change in the Log Timestamp Format (Settings > Logging > View Server Logger Details > Log Timestamp Format) changes the timestamp format returned by `pub.client.sftp:ls` service. This causes the SFTP transfers to fail due to the timestamp parsing failure.
To address this issue, Integration Server now includes a server configuration parameter which specifies the timestamp format for SFTP client public services.
`watt.server.sftp.dateStampFmt`
Specifies the format of date to be used in the SFTP client public services.
To specify the date format to use, you can use any format that is supported by the Java class `java.text.SimpleDateFormat`. For example, to display the date with the format 08-12-02 14:44:33:1235, specify `dd-MM-yy HH:mm:ss:SSSS`. The default format for `watt.server.sftp.dateStampFmt` property is `yyyy-MM-dd HH:mm:ss z`.
It is not necessary to restart Integration Server after you change the value of this parameter.
- PIE-40578
Reloading the Packages > Management page in Integration Server Administrator results in a re-execution of the last command on the page.
The Packages > Management page in Integration Server Administrator encodes the last action as part of the URL. Web browsers remember the last action as part of the history, resulting in the re-execution of the last action when reloading the frame.
To resolve this issue, for HTML5-compatible browsers, Integration Server alters the browser history so that that the URL for the Packages > Management page does not include the last action as part of the URL. As a result, when reloading the frame, the last action will not be re-executed.
- PIE-40629
A concurrent `webMethods` messaging trigger may not utilize all available threads.
When a concurrent `webMethods` messaging trigger uses the maximum available threads for message processing, the next request for a thread must wait for a thread to be returned to the thread pool before another message can be processed. Integration Server returns a thread to the thread pool once message processing completes. When a thread becomes available, the waiting request should begin executing. However, instead of the request using a thread as soon as it becomes available, the request waits until the number of threads used by the trigger reaches 0 (zero). This results in periods of time when the trigger is not using all of the available threads.
Note: A trigger uses the maximum available threads when the number of messages being processed by the trigger equals the `Max execution threads` property value. For example, if `Max execution threads` is set to 10, the trigger uses the maximum available threads when it is processing 10 messages.
Now, a concurrent `webMethods` messaging trigger will not wait for the number of the threads used by the trigger reaches zero before processing a waiting request. Instead, the trigger begins processing the waiting request as soon as a thread become available.
- PIE-40664
The JSSE-enabled HTTPS ports fail to use the updated certificate.
After adding a new certificate to the truststore and then reloading the truststore using the

Integration Server Administrator (Security > Keystore), the JSSE-enabled ports do not use the added certificate.

This issue is now resolved. The JSSE-enabled HTTPS ports now use the updated certificate after the corresponding truststore is reloaded using Integration Server Administrator.

- PIE-40684
Integration Server Administrator sometimes displays two scrollbars right next to each other instead of a single scrollbar.
This issue is now resolved.
- PIE-40687
Display non-ASCII Unicode characters in the body of a Java service.
When you persist Java services, Integration Server encodes non-ASCII Unicode characters as ASCII Unicode escape sequences (for example, the Korean character '?' is saved as '\uD55C'). By default, Integration Server does not decode the escape sequences. Therefore, when Integration Server sends Java service code to Software AG Designer, escape sequences appear in Designer instead of the original Unicode characters.
This fix lets you configure Integration Server to decode the escape sequences. In Integration Server Administrator, go to Extended Settings and add "watt.server.ns.decodeJavaService=true." The change will take effect immediately (no server restart necessary).
Note: After this change, Unicode escape sequences will no longer display in the Java service code in Integration Server. For example, if you use the string "\uD55C" in Java service code, the string will save correctly, but the editor will display it as "?".
- PIE-40688
Invoking pub.storage:get causes a pub.storage lock to be released immediately.
When using pub.storage:get to retrieve an item, Integration Server released the lock on the item immediately. However, when using pub.storage:get to retrieve an item, the lock on the item should remain until the lock is removed by a call to pub.storage:unlock, a call to pub.storage:put, or the lock expires.
This issue is now resolved.
- PIE-40693
Slow file transfer speed using pub.client.ftp:put.
While transferring files with large size (around 1 GB) using pub.client.ftp:put service, the file transfer speed is observed to be slow. Hence, it takes longer to complete the file transfer. This issue was due to the buffer limit in the amount of data that can be read and written to the stream (1KB).
This issue is now resolved. Now, the buffer size is increased (32 KB) and hence, the transfer time for large files is less and is comparable with the FTP client on Microsoft Windows.
- PIE-40702
Change to Integration Server to address security vulnerability.
A certain form of URL, if executed, can cause a redirection that allows for the possibility of credential theft. The form of URL no longer causes a redirection. Instead, the Integration Server user receives an HTTP 404 response.

- PIE-40742
 Package deployment fails because some packages are partially loaded.
 When using Deployer to deploy IS packages, some packages only partially load. This causes deployment to fail.
 This issue is now resolved. Packages are no longer partially loaded.
- PIE-40867
 Modification to Integration Server to control the logging of messages about duplicate universal names for document types.
 Whenever Integration Server loads a document type with an explicit universal name (such as when a package is loaded), Integration Server registers the universal name. If Integration Server loads a document type whose explicit universal name is a duplicate of one that is already registered, Integration Server logs the following error message:
 "[ISC.0081.0002E] Error registering universal name <universal_name>, already registered to <document_type>" filling the IS logs
 A user cannot use Designer or Developer to assign the same universal name to a document type. However, if the same WSDL document is used to generate multiple consumer Web service descriptors and/or multiple WSDL first provider Web service descriptors, Integration Server creates multiple document types with the same universal name explicit. When loading a package containing these document types, Integration Server logs the error message mentioned above multiple times which can clutter the log file. To suppress the error message about registering a duplicate universal name, add the following to extended settings on the Settings > Extended > Edit Extended Settings page
 watt.server.ns.logErrorsOnRegisteringMultipleDocTypesForAUniversalName=false
 After you save your changes, you do need to restart Integration Server for these changes to take effect. If you want Integration Server to resume logging message about registering duplicate universal names, set watt.server.ns.logErrorsOnRegisteringMultipleDocTypesForAUniversalName to true. The default value is true.
 Note: Setting watt.server.ns.logErrorsOnRegisteringMultipleDocTypesForAUniversalName to false suppresses the error messages about duplicate universal names only. It does not resolve the duplicate names.
- PIE-40940
 OAuth authentication is not disabled for requests that contain an Authorization header with a Bearer access token even after setting the watt.server.auth.oauth.accessToken.useHeaderFields parameter to false.
 When watt.server.auth.oauth.accessToken.useHeaderFields is set to false, Integration Server should not perform OAuth authentication if Authorization header field contains an access_token. However, if the Authorization header field of an incoming request contains a Bearer access token and the watt.server.auth.oauth.accessToken.useHeaderFields is set to false. Integration Server writes the following messages to the server log:
 [ISS.0012.0012W] Authentication of user "bearer" failed with exception: [ISS.0010.8044] Integration Server rejected the request to access this resource. The access token is either invalid or expired.

[ISS.0053.0002C] Access denied for user bearer on port 5555 -> ...

This issue is now resolved. If `watt.server.auth.oauth.accessToken.useHeaderFields` is set to `false` and a request contains an Authorization header field with a Bearer access token, Integration Server disables OAuth authentication for the request.

- **PIE-40950**
Failure to close JSSE SSL connections after handling stateless requests may negatively impact performance.
When Integration Server finishes handling a stateless request received on a port that uses JSSE for SSL, Integration Server does not close the SSL connection. This causes Integration Server to build up stale connections and slows down port allocation, which affects performance.
This issue is now resolved.
- **PIE-40962**
The `encode` method of `IDataXMLCoder` class closes the `OutputStream` after encoding the `IData` object.
The `encode` method of `IDataXMLCoder` class, `encode(java.io.OutputStream os, IData data)` encodes the `IData` object and closes the `OutputStream` even though the `OutputStream` is created by the caller.
This issue is resolved.
- **PIE-40984**
Configuration property to determine whether missing SOAP headers result in an error message.
Prior to Integration Server version 9.0, when processing a SOAP received by a web service descriptor, Integration Server did not properly validate that all of the SOAP headers were present. As a result, Integration Server did not throw an error when a SOAP response was missing a SOAP header. Note that all headers in a web service descriptor, whether generated from the original WSDL document or added to the web service descriptor, are treated as required. This is an application of the WS-I basic profile rules declaring that all headers in a WSDL be treated as required. Beginning in Integration Server version 9.0, Integration Server properly validates SOAP responses for required headers. If a required header is not present, Integration Server throws the following error: [ISS.0088.9443] One or more required Headers <headerName> are not present in the SOAP response."
After upgrading to Integration Server 9.0 or higher from an earlier version, web service connectors that previously succeeded when a required header was missing from a SOAP response now fail. While failure when a required header is missing is the correct and documented behavior, Integration Server now provides a configuration property to control whether missing required headers in a SOAP response results in an error. This can be useful in migration situations.
When `watt.server.SOAP.ignoreMissingResponseHeader` is set to `true`, a SOAP response that does not contain a required header will not result in an error. The default is `false`, a SOAP response that does not contain a required header results in an error. Integration Server restart is not required for changes to take effect.
- **PIE-40985**

SOAP web service fixes for backward compatibility.

This fix bundles four separate fixes relating to upgrade and migration of web services. The fixes allow backward compatibility of behavior that has since been prohibited through stricter validation in newer releases.

(1) Allow 7.1.3 web service providers to process requests with mismatched SOAP versions.

A range of fix levels in 7.1.3 allowed web service providers to process SOAP requests with a SOAP version that did not match the SOAP version that was declared in the WSDL binding. This fix adds a new extended server setting to emulate the old, less strict behavior so that consumers will not have to update their SOAP requests. The setting only affects web service providers that were created in releases older than 8.2.2 and for which the property "Pre-8.2 compatibility mode"=true. The new parameter is `watt.server.SOAP.pre82WSD.ignoreVersionMismatch`. By default, this parameter is set to false. To use the parameter to emulate the old behavior, go to Integration Server Administrator and add `watt.server.SOAP.pre82WSD.ignoreVersionMismatch=true` to the extended settings.

(2) Correctly perform document type-directed decoding of a SOAP message when the corresponding namespace qualified document type field does not have a namespace prefix. Starting with release 8.2.2, any document type field that contains a namespace must have a namespace prefix in its name in order to be used with web services. In 7.1.x, this prefix was not required. With this fix, document types migrated from 7.1.x that do not meet the 8.2.2 and later IS web service requirements will match the decoded IData representations of SOAP messages as they did in 7.1.x. Specifically, this issue applies in cases where the Decoder needs to choose between converting a simple type value into a string field or into a mixed content DocType with a *body field.

(3) Allow empty or absent *body values to pass validation.

A validation error occurred in some 9.x releases during decoding of mixed content elements that have an enumeration restriction. The error occurred when optional missed content was absent, because an empty value was not listed as an option in the enumeration set for the *body field. This fix adds an extended setting to skip this validation and treat the *body as completely optional. The new parameter is `watt.core.validation.skipAbsentStarBody`. By default, this parameter is set to false.

To use the parameter to skip the new validation, go to Integration Server Administrator and add `watt.core.validation.skipAbsentStarBody=true` to the extended settings. (4) Process one-way SOAP request without a Null Pointer Exception. After receiving a SOAP request, some one-way web service providers return an error containing a Null Pointer Exception. This fix handles the condition that resulted in an error when processing the SOAP request.

- PIE-41066

When executing the `pub.soap.wsrn:createSequence` service without specifying the value of the `sequenceKey` input parameter, Integration Server incorrectly reports the messaging sequence as null.

This issue is now resolved. When the `pub.soap.wsrn:createSequence` service is executed without specifying the value of the `sequenceKey` input parameter, Integration Server returns the server Sequence ID associated with the messaging sequence.

- PIE-41090

Enhancement to Integration Server to use regular expressions for specifying the URIs for allowed origin servers.

When CORS is enabled, the `watt.server.cors.allowedOrigins` server configuration parameter indicates the origin servers from which Integration Server will accept requests. The `watt.server.cors.allowedOrigins` value can be a comma-separated list identifying the specific origin servers or a `"*"`, meaning any origin server is allowed. A comma-separated list can become long and difficult to maintain.

To make the list of origin servers easier to maintain, Integration Server now supports the use of regular expressions in the comma-separated list of allowed origin servers. Integration Server treats any value in the comma-separated list that begins with `"r:"` as a regular expression. Integration Server treats any value that does not begin with `"r:"` as a simple string. The server configuration parameter uses the Java regular expression syntax, as documented at <https://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html>. A regular expression value must match the entire value of the Origin header in the HTTP request for it to be considered a match.

Example:

```
watt.server.cors.allowedOrigins=http://test1.domain.com,r:https?://.*.test2.domain.com:[0-9]+,r:.\.[a-zA-Z]*-int.domain.com
```

Integration Server treats the first value, `"http://test1.domain.com"`, as a simple string. If an Origin header contains this value, it will be allowed.

The second value, `"r:https?://.*.test2.domain.com:[0-9]+"`, contains a regular expression. The `"r:"` is not part of the regular expression. The actual regular expression used to match supplied Origin headers is `"https?://.*.test2.domain.com:[0-9]+"`.

The third value, `"r:.\.[a-zA-Z]*-int.domain.com"`, contains a regular expression. The `"r:"` is not part of the regular expression. The actual regular expression used to match supplied Origin headers is `".\.[a-zA-Z]*-int.domain.com"`.

`"Origin: http://test1.domain.com"` will be allowed because it is equal to the first value.

`"Origin: http://my.test2.domain.com:8080"` will be allowed because it matches the second value.

`"Origin: https://my.test2.domain.com:8088"` will be allowed because it matches the second value.

`"Origin: http://my.test2.domain.com"` will not be allowed. If it had a port number, it would match the second value.

`"Origin: nbps://example.prod-int.domain.com"` will be allowed because it matches the third value.

`"Origin: example.qa.staging-int.domain.com"` will be allowed because it matches the third value.

`"Origin: example.dev1-int.domain.com"` will not be allowed. If the second token of the host name did not include any digits, it would have matched the third value.

```
watt.server.cors.allowedOrigins=*
```

The simple wildcard `"*"` continues to work as before. It indicates that any origin server will be allowed.

Regular expressions that match any host name, IP address and port (e.g. `"r:+"` and `"r:.*"`) will have the same effect as `"*"`.

Note that when CORS is enabled, Integration Server evaluates the list of regular expressions in `watt.server.cors.allowedOrigins` sequentially for every request. Integration Server performs a regular expression match operation on each regular expression until a match is found or all regular expressions in the list have been evaluated. Software AG recommends that you put the more

frequently matched regular expressions at the beginning of the comma-separated list.

- **PIE-41109**
Swagger document and REST Resources tab for REST API descriptor does not represent service input correctly when the service signature includes one or more Document, Document List, Document Reference, or Document Reference List field and another type of field.
The Swagger document and REST Resources tab for a REST API descriptor that includes a REST resource (service) whose service signature includes one or more Document, Document List, Document Reference, or Document Reference List fields in addition to another type of field does not correctly represent the service signature. Integration Server places all of the input parameters for that resource into a REST Definition and sets the source to Body which is incorrect. Integration Server should place the Document, Document List, Document Reference, or Document Reference List fields in the REST Operation as Body parameters that reference the REST Definitions and place the parameters that are not documents in the REST Operation as Header parameters.
This issue is now resolved.

- **PIE-41121**
Integration Server does not show reliable messaging sequence reports in certain situations. When Integration Server exchanges messages with a reliable messaging client where the web service operation uses In-Only Message Exchange Pattern (MEP), the reliable messaging sequence reports are not displayed on Integration Server.
The issue is now resolved. The reliable messaging sequence reports specifying the Server Sequence ID are displayed on Integration Server in situations where the web service operation uses In-Only MEP.

- **PIE-41147**
Integration Server Administrator returns a blank page when you run the `wm.server.query:getSettings` service from the `WmRoot` package.
This issue is resolved.

- **PIE-41152**
When `watt.net.http401.throwException` is set to false and the `pub.client:http` service receives a 401 error, the service does not include the body of the response in the output.
If the `watt.net.http401.throwException` server configuration property is set to false, when the `pub.client:http` service receives a 401 error response, the service places the HTTP response header and body in the output pipeline. However, the `pub.client:http` service was not placing the response body in the output pipeline.
This issue is now resolved.

- **PIE-41155**
Integration Server fails to act as an FTP server. An attempt by an FTP client to connect to an FTP port on Integration Server fails with a `NullPointerException`.
This issue is now resolved.

- PIE-41176
 Web service fails with a RampartException while handling a holder-of-key SAML 2.0 assertion. When handling a holder-of-key SAML 2.0 assertion, a web service fails with the following exception: RampartException: No crypto property file supplied for decryption. This issue is now resolved.
- PIE-41281
 Integration Server Administrator displays incorrect values for fields in Security > Ports. Integration Server Administrator displays incorrect values (default values) for "Use JSSE" and "Client Authentication" fields in the Security Configurations section of Security > Ports. This issue is now resolved.
- PIE-41315
 When Integration Server receives a request with a REST-style URL and an OAuth Bearer token, Integration Server rejects the request with a 403 HTTP status code. This issue is now resolved. An OAuth Bearer token can be used to access a REST resource that is in the scope for which the token was issued.
 Note: Because the service name does not appear in a REST-style URL, do not include the service name when defining the OAuth scope. A scope for an Integration Server REST resource must include the full name of the resource but not the name of the service.
- PIE-41320
 Memory leak in transaction manager.
 Transaction manager retains an association to com.wm.app.b2b.server.ServerThread which is not being removed during transaction manager cleanup. Consequently, the association between the transaction and the server thread is not being cleaned up, which prevents the transaction manager entry from being removed. This issue is now resolved.
- PIE-41324
 Integration Server Administrator displays a Package Load Error loading a package created on Integration Server 7.x or earlier.
 Using Integration Server Administrator to load a package created on Integration Server 7.x might result in a Package Load Error caused by a NullPointerException.
 This issue was originally resolved in fixes that included PIE-33326. Unfortunately, recent revisions of some Integration Server fixes introduced a regression that allowed the problem to reappear. This fix resolves the regression.
- PIE-41345
 The pub.utils.messaging:migrateDocTypesTriggersToUM service does not migrate filters with parentheses properly.
 When migrating a trigger filter condition that contains parentheses, the pub.utils.messaging:migrateDocTypesTriggersToUM service removes the parentheses from the

filter. The `pub.utils.messaging:migrateDocTypesTriggersToUM` service now preserves parenthesis when migrating trigger filter conditions.

- **PIE-41375**
Integration Server does not indicate when it is disconnected from a cluster.
Integration Server does not log an error message when it is disconnected from a cluster, preventing automatic detection of the situation. Now, Integration Server logs the following error message when it is disconnected from the cluster.
[ISS.0033.151] The cluster is now not operational,
Additionally, Integration Server logs the following error message when it rejoins the cluster.
[ISS.0033.152] The cluster is now operational.
- **PIE-41444**
Integration Server throws a `NullPointerException` instead of logging trace level information when the HTTP Header (0038) server log facility is set to Trace and an incoming request does not contain any credentials.
This issue is now resolved.
- **PIE-41464**
Exception handling for the `pub.client:smtp` service results in a `ClassCastException` being thrown for a `SocketTimeoutException` or a `ConnectException`.
The `pub.client:smtp` incorrectly handles the exceptions `java.net.SocketTimeoutException` and `java.net.ConnectException` by throwing them as a `java.lang.ClassCastException`.
This issue is now resolved.
- **PIE-41497**
A request that includes a URL alias for a REST resource and includes an OAuth access token fails. When a URL alias exists for a REST resource and partial matching is enabled for URL aliases (`watt.server.url.alias.partialMatching=true`), the URL alias can be used with variable trailing tokens to create a request URL that invokes services for the REST resource and passes the `$resourceID`, `$path`, and custom variables to the services. (With the Bearer authorization scheme, the Authorization request header includes an OAuth access token instead of username and password.) However, if the request URL uses the Bearer authorization scheme, the request fails and Integration Server returns an HTTP 401 status code.
This issue is now resolved. A request that uses the Bearer authorization scheme can also use a URL alias to request a REST resource.
- **PIE-41502**
When acting as a web service client, Integration Server displays an error if the response from a web service provider contains the value of the content-encoding header as identity.
When acting as a web service client, Integration Server displays the following error when the response from a web service provider contains the value of the content-encoding header as identity:
"Received unsupported content-encoding: identity"
This issue is now resolved.

- PIE-41529

Info message is logged continuously in the server log.

If you configure the email port to IMAP email server and select "Log out after each mail check" functionality, then the following Info message is continuously logged in the server log:
"[ISP.0068.0029I]Logging out of IMAP Server"

This issue is now resolved. Now, this message is moved to DEBUG level.
- PIE-41582

Integration Server Administrator does not allow Username/Password as the Client Authentication method if Realm URL specifies nsps or nhps as the protocol.

When creating a Universal Messaging Connection Alias, Integration Server Administrator performs a UI validation to check if Realm URL specifies nsps or nhps as the protocol. Integration Server Administrator displays the following message when you select Username/Password as the Client Authentication method. "Client authentication must be set to SSL if Realm URL specifies includes nsps or nhps."

This issue is resolved. The UI validation is now disabled which allows you to select Username/Password as the Client Authentication method if Realm URL specifies nsps or nhps as the protocol.
- PIE-41611

Client side queue for webMethods messaging does not drain completely.

When Integration Server drains the client side queue by publishing or delivering messages from the client side queue to the webMethods messaging provider, some failures may occur. Integration Server handles the failures based on the exception type. If the exception type is fatal, Integration Server writes the message to the audit log and removes it from the client side queue. If the exception type is transient, Integration Server retries publishing or delivering the message. However, if the client side queue contains published and delivered messages and fatal exception occurs during publish or delivery of a message, Integration Server might add the message back to the client side queue instead of writing it to the audit log. This prevents the client side queue from fully draining and may cause the queue to drain slowly.

This issue is now resolved.
- PIE-41677

Integration Server fails to clear the connections of failed remote service executions that were invoked using the pub.remote:invoke service.

This issue is now resolved. Now, if a remote service execution fails, then the connection to remote server is closed completely.
- PIE-41782

No option to select Java Secure Socket Extension (JSSE) socket factory for HTTPS outbound connections.

When accessing a web service through HTTPS protocol using the pub.client:soapClient service, there is no option to select JSSE for HTTPS outbound connections. The server configuration

property "watt.net.ssl.client:useJSSE" applies to all outbound HTTPS connections and hence, user cannot select JSSE for individual web services call.

This issue is now resolved. Now, an optional input parameter "useJSSE" is added to the pub.client:soapClient service.

If the value of "useJSSE" field is left empty or unspecified, then Integration Server uses JSSE for outbound web service call based on the server configuration property "watt.net.ssl.client.useJSSE".

If the value of "useJSSE" field is set to "yes", then the Integration Server uses JSSE for the outbound web services call.

If the value of "useJSSE" field is set to "no", then the Integration Server uses Entrust IAIK library for the outbound web services call.

- PIE-41794

Changes to Integration Server to notify a caller that provides an expired session id.

After applying a fix or installing a release that includes PIE-37166, Integration Server prompts users for credentials if no matching session object can be found for a given session ID stored in a cookie.

This behavior is controlled by the watt.security.session.forceReauthOnExpiration server configuration property. When set to true, the more secure behavior, Integration Server rejects requests that include a session id mapped to an expired or invalid session object and prompts the user for credentials. Setting the server configuration property to false instructs Integration Server to create a new session object, assuming the session id is valid and trusted. In case of the server configuration property value set to true (the default), Integration Server does not notify the caller that a valid session object no longer exists. Nor does Integration Server request that the caller deletes the cookie with the expired session.

Now, when Integration Server receives a request with a session id mapped to a session object that no longer exists, Integration Server notifies the caller and requests that the cookie containing the expired session id be deleted. It is then up to the caller, usually a browser, to react to the notification.

- PIE-41806

Enhancement to allow loggers on multiple Integration Servers to share a Universal Messaging queue for logging across a Universal Messaging realm.

The naming convention for the Universal Messaging queue used by a logger now includes the client prefix if the Share Client Prefix check box is selected for the Universal Messaging connection alias used by the logger. By including the client prefix in the namespace of the queue name, loggers for Integration Servers in a stateless or stateful cluster can share same Universal Messaging queue across a Universal Messaging realm. That is, loggers on multiple Integration Servers can write log entries to and read log entries from one Universal Messaging queue shared across a Universal Messaging realm.

Previously, the Universal Messaging queue name did not include the client prefix as part of the queue namespace regardless of the state of the Share Client Prefix check box. If the Share Client Prefix check box is selected for the Universal Messaging connection alias, the Universal Messaging queue name uses the naming convention: wm/is/audit/clientPrefix/logger_nameQueue (for example, wm/is/audit/myClientPrefix/SessionQueue). If the Share Client Prefix check box is not selected for the Universal Messaging connection alias, the Universal Messaging queue name uses

the naming convention: `wm/is/audit/logger_nameQueue` (for example, `wm/is/audit/SessionQueue`). If you change the state of the Share Client Prefix check box on the Universal Messaging connection alias is used by loggers, you must restart Integration Server for changes to the queue name to take effect. Additionally, you must make the same changes to other Integration Servers in the cluster that use the alias.

- **PIE-41825**
Cookies not included in some HTTP requests.
When the `pub.client:http` service is used to send requests to a remote server, and the server returns a cookie, in some cases the cookie is not included in subsequent calls to the server.
With this fix, the cookies are used on subsequent calls.

- **PIE-41880**
webMethods Messaging triggers do not recognize nested documents in a filter when parent and child documents have the same name.
The name of inline document fields are represented as "Message" names in the protobuf descriptors used by Universal Messaging filters. Because of the message name conflict, the filters are never satisfied and the triggers do not fire.
This fix corrects the problem by using the name of the parent plus the actual field as the protobuf message name. After you install this fix, edit and save the root document type, then sync it with the UM Provider. The filters will begin to work properly.

- **PIE-41900**
The `pub.client:http` fails to set the auth token.
If the 'auth' type is set to Bearer, then the `pub.client:http` service fails to set the auth token into the Authorization header while sending a request to the HTTP server.
This issue is now resolved.

- **PIE-41986**
After installing a fix or a release that includes PIE-37166, a dynamic server page (DSP) request that includes a service invoke of a stateless service prompts for credentials even though the service and DSP are secured with the Anonymous ACL.
The resolution for PIE-37166 introduced the server configuration parameter `watt.security.session.forceReauthOnExpiration`, which, when set to true, causes Integration Server to reject any request that includes a cookie identifying an expired or invalid session even if the request includes valid user credentials. That is, when set to true, Integration Server does not implicitly create a new session for an expired session and Integration Server deletes the session object for the expired session. A result of this change is that a sequence a DSP requests that includes a service invoke of a stateless service prompts for credentials even though the service and DSP are secured with the Anonymous ACL.
Note: This issue does not occur when `watt.security.session.forceReauthOnExpiration` is set to false. Keep in mind that setting `watt.security.session.forceReauthOnExpiration` to true, the default value, offers more secure behavior.
Now, Integration Server preserves the session object.

- PIE-42132

Enhancement to limit the number of server log files that Integration Server keeps on the file system. Integration Server writes its server log to a file named server.log. When this file is rotated, the existing contents of server.log are saved to a new file named server.log_<current-date>_<current-time>. Over time, the number of archived server log files increases and may consume large amounts of disk space.

To address this issue, Integration Server provides the `watt.server.serverlogFilesToKeep` server configuration parameter to control the number of server log files that Integration Server maintains on the file system, including the current server log file. When Integration Server reaches the limit for the number of server log files, Integration Server deletes the oldest archived server log file each time Integration Server rotates the server log.

If the parameter value is 30, for example, Integration Server keeps the current server log file and up to 29 previous server log files.

If the parameter value is less than 1, Integration Server keeps an unlimited number of server log files.

If the parameter value is 1, Integration Server keeps the current server log file (server.log) and no previous (archived) server logs.

The default for `watt.server.serverlogFilesToKeep` is -1, which places no limit on the number of server log files kept on the file system.

You must restart Integration Server for changes to this parameter to take effect.

- PIE-42173

New parameter for existing service `pub.cache.serviceResults:resetServiceCache` and new service `pub.cache.serviceResults:addCacheEntry` are added to Integration Server as part of this fix.

`pub.cache.serviceResults:addCacheEntry`

`WmPublic`. Adds cached entry into service results for the service without executing the actual service. This service can be useful to perform bulk load of service results at Integration Server startup.

Input Parameters <serviceName> String Required. Name of the service for which to add the cache entry into cached service results.

<input> Document Required. An IData containing the key/value pairs for the cached service input.

<output> Document Required. An IData containing the key/value pairs for the cached service output.

Output Parameters None.

Usage Notes If <serviceName> does not exist in Integration Server, or it is not enabled to cache service results, Integration Server throws an exception.

`pub.cache.serviceResults:resetServiceCache` An optional input parameter is added to delete specific cached elements for a particular service.

<input> Document Optional. An IData containing key/value pairs that indicate the cached elements to remove.

- PIE-42308

Support for session caching for transacted JMS connection aliases.

Previously, Integration Server provided session caching for non-transacted JMS connection aliases. Now, Integration Server provides support for session caching for transacted JMS connection aliases when the JMS provider is Universal Messaging 9.10 or WebSphere MQ 7.5. The following paragraphs provide more information about this feature.

When sending a JMS message, Integration Server creates and closes a new JMS session object and a JMS MessageProducer object for each message. This can introduce overhead for some JMS providers. To improve performance when sending JMS messages using a transacted JMS connection alias, you can configure session caching for a JMS connection alias.

For each JMS connection alias, Integration Server can create the following session pools:

- A default session pool containing JMS Session objects. When a default session pool is defined for a JMS connection alias, Integration Server draws from a pool of open JMS Sessions for sending a JMS message instead of opening and closing a JMS Session for each JMS message. Integration Server uses the default session pool only when sending a message to a destination that does not have its own pool. Integration Server creates a new MessageProducer each time it sends a JMS message.
- Destination-specific session pools containing JMS Session objects for sending JMS messages to identified destinations. Integration Server creates a pool for each specified destination. When sending a JMS message to one of the specified destinations, Integration Server uses a Session object from the pool instead of creating and closing a Session object for each JMS message. Integration Server creates a new MessageProducer each time it sends a JMS message.

Note: When using destination-specific session pools for a transacted JMS connection alias, Integration Server creates a new MessageProducer each time it sends a JMS message. This is different from the destination pools that Integration Server creates for a non-transacted JMS connection alias where each entry in the destination-specific pools contains a Session object and a Message Producer object.

Note: A transacted JMS connection alias specifies LOCAL_TRANSACTION or XA_TRANSACTION for the Transaction Type.

To configure session caching for a transacted JMS connection alias, do the following:

1. Using Integration Server Administrator, open the JMS connection alias for editing.
2. Under Producer caching, for Caching Mode, select ENABLED PER DESTINATION.
3. To configure the size of the default session pool for this alias, specify the following:
 - In the Minimum Pool Size field, specify the minimum number of entries in the default session pool for this alias. The default is 1.
 - In the Maximum Pool Size field, specify the maximum number of entries in the default session pool for this alias. The default is 30.
4. To configure the size of the session pools created for specific destinations, specify the following:
 - In the Minimum Pool Size Per Destination field, specify the minimum number of entries in each destination-specific pool.
 - In the Maximum Pool Size Per Destination field, specify the maximum number of entries in each destination-specific pool. A value of 0 (or blank) indicates that Integration Server does not create separate pools for any of the destinations associated with the JMS connection alias.

In the Destination Lookup Name List, specify a semicolon-delimited list of the lookup names for the destinations for which you want Integration Server to create separate session pools for this alias.

5. Click Save Changes.

This issue is now resolved.

- **PIEAR-799**

The threads which creates the connection and the threads which ends the connection from the connectionPool can go to blocked state.

To recover from such a situation, Connection pool now has a monitor thread which is a pool interrupter thread, to interrupt the server threads which are in blocked state while creating or ending the connection.

The following properties are introduced to configure the connection time outs:

1-> `watt.server.jca.connectionPool.threadInterrupter.sleepTime`. The default value of the `watt` property is 2000msec, which is the sleep time for the pool interrupter thread.

2-> `watt.server.jca.connectionPool.threadInterrupter.waitTime`. This property specifies the wait time, measured in milliseconds, that elapses before Integration Server Connection pool interrupts a connection creating or ending thread which is in a wait state. The pool interrupter thread will start monitoring the server threads, only if this property is set. There is no default value.

You must restart Integration Sever for changes to this parameter to take effect.

Use the following service to manage the connections which are hung:

`pub.art.connection:getInterruptedThreadStatus`. This service takes the connection alias name as input and lists the connection threads which are not responding even after interrupting by the connection pool interrupter. We recommend a manual intervention for the server threads which gets hung while creating or destroying the connections and also not responding to the interrupt mechanism.

Release 9.10

- **PIE-36529 (IS_8.2_SP2_Core_Fix18, IS_9.0_SP1_Core_Fix12, IS_9.5_SP1_Core_Fix9, IS_9.6_Core_Fix8)**

An outbound web service request that used MTOM attachments causes out of memory issues if the attachments are large.

An outbound web service request incorrectly used a memory-buffered output stream when requests with large MTOM attachments were made to external endpoints.

Outbound web service requests that use MOTM attachments now use the correct output stream and chunk the large MTOM attachment.

- **PIE-36506 (IS_8.2_SP2_Core_Fix19, IS_9.0_SP1_Core_Fix12, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)**

The `pub.security.keystore:setKeyAndChain` service clears security parameters after each outbound call instead of when the `pub.security.keystore:clearKeyAndChain` service executes.

In previous releases, Integration Server did not clear the security parameters configured for the outbound call by the `pub.security.keystore:setKeyAndChain` service until the `pub.security.keystore:clearKeyAndChain` executed. Now, Integration Server clears the security parameters after each outbound call. The new behavior is not backward compatible.

Now, the security values configured by `pub.security.keystore:setKeyAndChain` stay in effect until `pub.security.keystore:clearKeyAndChain` gets invoked.

- **PIE-37676 (IS_8.2_SP2_Core_Fix19, IS_9.0_SP1_Core_Fix12, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)**

An attempt to authenticate a cache manager user fails with an `IllegalThreadStateException`.
When authenticating users of a cache manager, a race condition caused an `IllegalThreadStateException` to be thrown.
The race condition that caused the `IllegalThreadStateException` is now resolved.

- PIE-36620 (IS_8.2_SP2_Core_Fix19, IS_9.0_SP1_Core_Fix13, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
Integration Server returns an HTTP 500 error in response to successive requests made on persistent HTTP connections.
When client reuses a persistent HTTP connection to send multiple requests, Integration Server should handle the request and send a response. However, Integration Server sometimes incorrectly returned a HTTP 500 error, which caused the request to fail.
This issue is now resolved.
- PIE-37093 (IS_8.2_SP2_Core_Fix19, IS_9.0_SP1_Core_Fix13, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
When using the VCS Integration feature in Integration Server, delete and safe delete of a version-controlled node fails with an error.
This issue is now resolved.
- PIE-37325 (IS_8.2_SP2_Core_Fix19, IS_9.0_SP1_Core_Fix13, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
Integration Server does not send HTTP cookies after a request is redirected.
When the `pub.client:http` service sends a request to an HTTP server and the HTTP server responds with a redirection (a 300-level status code) and a `Set-Cookie` header, Integration Server includes the cookie in the request to the HTTP server to which the request is being redirected. However, subsequent requests to the second server do not include the cookie.
Now, when a HTTP server includes a `Set-Cookie` header in a redirection response, Integration Server includes the cookie in the request and subsequent requests sent to the HTTP server to which the request is redirected.
- PIE-38245 (IS_8.2_SP2_Core_Fix19, IS_9.0_SP1_Core_Fix13)
The `pub.flow:transportInfo` document type signature is incorrect as seen on the Input/Output tab.
The signature for the `pub.flow:transportInfo` document type is missing 2 fields: `filePolling/lastModified` and `filePolling/length`.
This issue is now resolved. The signature of the `pub.flow:transportInfo` document type now includes the `filepolling/lastModified` and `filepolling/length` parameters.
- PIE-38290 (IS_8.2_SP2_Core_Fix19, IS_9.0_SP1_Core_Fix13)
Integration Server resets the central user management system when lookup for a user in the central user management fails with an error.
The reset causes a failure of other operations on the central user management system.
This issue is resolved.
- PIE-37898 (IS_8.2_SP2_Core_Fix19, IS_9.6_Core_Fix9)
Integration Server becomes unresponsive during startup.
Integration Server sometimes became unresponsive at startup when initializing the scheduler. This

occurred because of a database deadlock.

This issue is now resolved and the deadlock will no longer occur.

- PIE-36477 (IS_8.2_SP2_Core_Fix19)
After transitioning to or from daylight savings time, scheduled tasks run twice or not at all. Integration Server runs scheduled tasks based on the time relative to the time zone. This caused issues when entering and exiting daylight savings time, specifically tasks ran twice or not at all. Now, Integration Server runs scheduled tasks based on coordinated universal time (UTC). Because Integration Server runs the tasks without regard to the relative time zone, the start and end of daylight savings time does not affect the execution of scheduled tasks.
- PIE-36386 (IS_9.0_SP1_Core_Fix11, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix8, IS_9.7_Core_Fix5)
When a flow service invokes a service that does not exist, Integration Server handles operations on the pipeline first before issuing a runtime exception.
When a flow service invokes a service that does not exist, Integration Server performs operations on the pipeline, such as removing the dropped variables from the pipeline, before issuing a `com.wm.lang.flow.FlowException` about the unknown service.
This issue is resolved. Integration Server now issues the runtime exception when a flow service invokes a service that does not exist and will not proceed with the pipeline operations.
- PIE-35990 (IS_9.0_SP1_Core_Fix12, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix8)
When using Oracle Streams Advanced Queuing (AQ) as the JMS provider, if the JMS connection alias is disabled due to an error from the JMS provider, the JMS connection continues to show the status as "in progress".
If a JMS connection alias, when attempting to establish a connection with Oracle Streams Advanced Queuing (AQ) as the JMS provider, is disabled due to an error from the JMS provider, the JMS connection continues to show the status as "in progress" and cannot be enabled or disabled.
This issue is resolved.
- PIE-37022 (IS_9.0_SP1_Core_Fix12, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix8)
Integration Server logs the following error twice for the same port in the server logs while shutting down:
[ISP.0046.0011I] Disabling HTTP Listener on port <port number>
This issue is resolved. Integration Server now logs this message only once while shutting down.
- PIE-35873 (IS_9.0_SP1_Core_Fix12, IS_9.5_SP1_Core_Fix9, IS_9.6_Core_Fix8)
Change to allow use of forward slash in a regular expression.
Prior to this change, regular expressions used in BRANCH steps and trigger filters could not include a forward slash. Now, a forward slash can be used. However, the forward slash must be preceded by the backslash (\) escape character. For example, to use a regular expression to match a variable whose value contains the string "a/b", use the regular expression `%variableName% = /a\b/`
- PIE-37271 (IS_9.0_SP1_Core_Fix12, IS_9.5_SP1_Core_Fix9, IS_9.6_Core_Fix8)
In certain scenarios, invoking the `pub.xml:getNextXMLNode` service fails with an exception. Invoking the `pub.xml:getNextXMLNode` service fails with a `NullPointerException` if the specified `NodeIterator` is returned by the `pub.xml:getXMLNodeIterator` service that has the criteria input

parameter set to a string value that matches an element node of the XML node.
This issue is resolved.

- PIE-34881 (IS_9.0_SP1_Core_Fix12)
Integration Server does not behave as expected when validating an XML document against a document type created from an XML schema definition containing `<xs:restriction base="xs:anyType">`.
When using the `pub.schema:validate` service to validate an XML document against a document type created from an XML schema definition containing `<xs:restriction base="xs:anyType">`, the validation does not fail even if the target element contains attributes other than the attributes defined in the XML schema definition.
This issue is resolved. Now, validation of XML fails if there are any attributes other than the defined attributes.
- PIE-36217 (IS_9.0_SP1_Core_Fix13, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
An outbound HTTPS web service request fails when using JSSE.
If an outbound HTTPS web service request specifies the use of JSSE, the request fails because Integration Server does not send the client certificates to the endpoint.
This issue is now resolved.
- PIE-37071 (IS_9.0_SP1_Core_Fix13, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
Integration Server error log does not include the names of the affected JDBC pools in the error messages.
When issuing error messages such as, “`java.sql.SQLException: Could not get connection`”, Integration Server error log does not include the names of the JDBC pools to which the error message is referring.
This issue is resolved. Integration Server error log now includes the name of the JDBC pools in its error messages.
- PIE-37551 (IS_9.0_SP1_Core_Fix13, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
Processing of webMethods messaging triggers fails with a `java.lang.ClassCastException` if the `watt.server.trigger.local.checkTTL` server configuration parameter is set to true.
Processing of webMethods messaging triggers fails with the following error if the `watt.server.trigger.local.checkTTL` server configuration parameter is set to true:
Unable to check document ttl: `java.lang.ClassCastException: [B cannot be cast to com.wm.data.IData`
This issue is now resolved.
- PIE-37645 (IS_9.0_SP1_Core_Fix13, IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
Incorrect logging level specified for error message.
Integration Server gives the logging level of the following error message of Trace logging level incorrectly as Info:
[ISC.0042.0001I] baseURL (Connection reset)
This issue is now resolved. The logging level of the message is corrected to Trace.
- PIE-38560 (IS_9.0_SP1_Core_Fix13, IS_9.5_SP1_Core_Fix10)
Integration Server logs a `NullPointerException` at one minute intervals.

Once per minute, the Integration Server scheduler uses a background thread to check for scheduled tasks in an invalid state. If Integration Server is in a cluster and the `watt.server.scheduler.logical.hostname` property is not set, a `NullPointerException` occurs and is written to the error log.

Now, the background thread used by the scheduler no longer throws a `NullPointerException` if Integration Server is in a cluster and the `watt.server.scheduler.logical.hostname` property is not set.

- **PIE-35697 (IS_9.0_SP1_Core_Fix13)**
After refreshing a consumer web service descriptor, response services reference the backup folders. After refreshing a consumer web service descriptor, the response services contain references to the backup folders.
This issue is now resolved.
- **PIE-37921 (IS_9.0_SP1_Core_Fix13)**
The `pub.client:http` service does not include the supplied data in PUT requests. When using the PUT method with the `pub.client:http` service, the request transmitted to the remote HTTP server does not include the value supplied for the `data/args`, `data/string`, `data/table`, `data/bytes`, `data/stream` or `data/mimeStream` input variables.
This issue is now resolved. When the PUT method is used with `pub.client:http`, the value of the `data/args`, `data/string`, `data/table`, `data/bytes`, `data/stream` or `data/mimeStream` input variables is now included as the HTTP entity of the transmitted request.
- **PIE-37988 (IS_9.0_SP1_Core_Fix13)**
A `NullPointerException` occurs when `SoftwareAG-IS-Services.xml` configuration file is not present in the caching directory.
When `SoftwareAG-IS-Services.xml` configuration file is not present in the directory location `<Integration Server_directory>\instances\<instance_name>\config\Caching`, Integration Server throws a `NullPointerException`.
This issue is resolved. Integration Server now logs the following error message in the server logs to indicate that the configuration file is not present in the caching directory.
Cannot cache "{0}" service. The cache manager `SoftwareAG.IS.Services` is not initialized.
- **PIE-38019 (IS_9.0_SP1_Core_Fix13)**
Integration Server acting as an SFTP client faces issues while attempting to connect to an SFTP server.
When attempting to connect to an SFTP server, Integration Server acting as an SFTP client issues the following error:
[ISS.0147.9010] Cannot get host key from server [host_X]:22.
Details: `com.jcraft.jsch.JSchException: Algorithm negotiation fail`
This issue occurs because there is no common key exchange algorithm between the SFTP client and SFTP server.
This issue is now resolved by updating the `jsch` jar file from 0.1.51 version to 0.1.53 version. The 0.1.53 version supports most of the key exchange algorithms that are required to be present in the SFTP client to connect to SFTP server.
- **PIE-38429 (IS_9.0_SP1_Core_Fix13)**

Integration Server logs an exception when a service is invoked through a JSSE-enabled HTTPS port with client authentication set to "Username/Password" or "Request Client Certificates".

When a service is invoked through an HTTPS port that uses JSSE and has client authentication set to "Username/Password" or "Request Client Certificates", Integration Server logs the following exception in the error log:

```
javax.net.ssl.SSLPeerUnverifiedException: peer not authenticated.
```

The issue is now resolved. Integration Server does not log any exception upon invoking a service successfully through a JSSE-enabled HTTPS port with client authentication set to "Username/Password" or "Request Client Certificates".

- PIE-38599 (IS_9.0_SP1_Core_Fix13)
The pub.client:http service does not honor the value set for the useJSSE parameter.
The useJSSE parameter of the pub.client:http service indicates whether to enable the use of the Java Secure Socket Extension (JSSE) socket factory for creating outbound HTTPS connections. However, the pub.client:http service fails to consider the value set for the useJSSE parameter and does not use JSSE for SSL connection with proxy configuration even if the "useJSSE" parameter is set to yes.
This issue is now resolved. The pub.client:http service now honors the value set for the useJSSE parameter.
 - If "useJSSE" is set to yes, Integration Server uses JSSE for HTTPS connection.
 - If "useJSSE" is set to no, Integration Server uses Entrust IAIK library for HTTPS connection.
- PIE-39117 (IS_9.0_SP1_Core_Fix13)
Subsequent HTTPS requests through proxy to the server fail.
When you send repeated outbound HTTPS requests to a server using the pub.client:http service with useJSSE=yes and HTTPS proxy alias, the first request is sent successfully but the subsequent requests fail with exception.
The issue is now resolved.
- PIE-37937 (IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix8)
Publishing a document as a JMS message from Designer fails with a NamingException.
Using Designer to publish an instance of an IS document type as a JMS message fails with a NamingException. This occurs because the pub.jms:send service, which is used by Designer to publish the JMS message, includes a new input parameter named JMSMessage/header/replyTo that Designer populates with an empty String. Integration Server attempts a JNDI lookup using the empty String, which results in the NamingException.
Now, Integration Server verifies that JMSMessage/header/replyTo parameter is not empty before performing the JNDI lookup.
- PIE-34925 (IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
When executing a web service connector, Integration Server does not check the Execute ACL of the parent consumer web service descriptor.
When executing a web service connector contained in a consumer web service descriptor that has a Pre-8.2 compatibility mode value of false, Integration Server did not check the Execute ACL of the consumer web service descriptor. The documented behavior is to always verify the Execute ACL for a web service descriptor, but it was not being done.
Now, when executing a web service connector, Integration Server checks the Execute ACL of the

parent consumer web service descriptor.

- PIE-37804 (IS_9.5_SP1_Core_Fix10, IS_9.6_Core_Fix9)
Integration Server KPI that Optimize uses to display lifetime statistics displays current statistics instead.
The Integration Server KPI, `getNumCompletedRequests`, which Optimize uses to display the number of completed requests over the lifetime of the Integration Server, returns the number of requests in the last polling period instead.
This issue is now resolved.
- PIE-37586 (IS_9.5_SP1_Core_Fix10)
Integration Server deletes the parent task when all the servers in a cluster are shut down. When you schedule a task to run on all servers in a cluster, and when one server in the cluster is shut down, Integration Server deletes the corresponding child task. However, when all the servers in the cluster are shut down, Integration Server deletes all the child tasks as well as the parent task. As a result, the parent task information is lost upon restart of Integration Server.
This issue is resolved.
- PIE-38131 (IS_9.5_SP1_Core_Fix10)
Change to Integration Server to allow the `pub.client:http` service to return response headers and response entity when receiving a 501 to 599 level response from a remote HTTP server.
HTTP servers indicate success or failure by including a status code in the response. The `pub.client:http` service returns the status code, response headers, and response body. When the HTTP response includes a status code in the 501 to 599 range, the `pub.client:http` service throws a `ServiceException`. Some HTTP servers include useful information in the response headers and entity when returning a 501 to 599 level status code. However, this information is lost when `pub.client:http` throws a `ServiceException`.
To address this issue, Integration Server now includes a server configuration parameter, `watt.net.http501-599.throwException`, which you can use to indicate how the `pub.client:http` service handles a 501 to 599 level response.
`watt.net.http501-599.throwException`
Specifies whether or not the `pub.client:http` service throws a `ServiceException` when receiving a 501 to 599 level response from a remote HTTP server. When set to true, the `pub.client:http` service throws a `ServiceException` when it receives a 501 to 599 level response from a remote HTTP server. When set to false, the `pub.client:http` service does not throw a `ServiceException` when it receives a 501 to 599 level response from a remote HTTP server. Instead, when the `pub.client:http` service returns a status code in the 501 to 599 range, the service returns the status code, response headers, and response body. The default is true. You do not need to restart Integration Server for changes to this parameter to take effect.
Note: When the remote HTTP server returns a response code of 500, the `pub.client:http` service returns the status code, response headers, and response body.
- PIE-38442 (IS_9.5_SP1_Core_Fix10)
Integration Server does not handle an element with `xsi:nil=1` correctly.
If an IS document type used by the web service contains a field of type Object for which the Allow null property is set to true and the XML instance document contains a corresponding element with

an `xsi:nil=1` and an `xmlns:xsi=` attribute, Integration Server does not properly convert the element. Instead, Integration Server incorrectly converts the element to a Document field with no children instead of converting the element to a field of type Object with a "null" value.

This issue can also occur when the `pub.xml:xmlNodeToDocument` service executes and all of the following are true:

- The `preserveUndeclaredNS` input parameter is set to true
 - The IS document type in the `documentTypeName` field contains a field of type Object for which the Allow null property is set to true
 - The supplied node contains a corresponding element with an `xsi:nil=1` and an `xmlns:xsi=`
- Now, Integration Server properly converts the `xsi:nil=1` element to an field of type Object with a "null" value.

▪ PIE-36837 (IS_9.5_SP1_Core_Fix9, IS_9.6_Core_Fix8)

Sessions are not being shared among Integration Servers in a cluster.

When a dynamic server page (DSP) is requested by a client, a session is created in Integration Server and a Set-Cookie response header is returned with the session ID. If a subsequent request from the client includes the session ID but a different Integration Server in the cluster receives the request, the second server should find the session in the cluster's session store and then use the session to service the request. This was not happening. The first server did not save the session to the cluster's session store. As a result, the client's session was not available to the second server. This has been fixed. When a clustered Integration Server receives a request for a DSP, the server saves the session to the cluster's session store.

▪ PIE-36942 (IS_9.5_SP1_Core_Fix9, IS_9.6_Core_Fix8)

Integration Server may become unresponsive when processing concurrent requests to disable file polling ports in different packages.

If Integration Server receives concurrent requests to disable a file polling port in a package and the ports are located in different packages, Integration Server may become unresponsive. Additionally, HTTP/S ports on Integration Server might become unavailable. This issue is now resolved.

▪ PIE-37005 (IS_9.5_SP1_Core_Fix9, IS_9.6_Core_Fix8)

Enhancement to `pub.jms:send` service to allow specifying a destination for the `JMSReplyTo` header. Currently, if you want to specify the `JMSReplyTo` header for a JMS message that you are sending, you must use the `pub.jms:sendAndWait` service. The `pub.jms:sendAndWait` service sends a request message and waits for a response. The act of waiting for a response message comes with extra overhead for Integration Server which is unnecessary if you merely want to specify a `JMSReplyTo` destination but do not want the sending service to wait for a reply.

To address this issue, the `pub.jms:send` service has been enhanced to include a new input parameter for setting the `JMSReplyTo` header. When invoking the `pub.jms:send` service, you can set the `JMSMessage/header/replyTo` field which is an optional parameter of type String. Set this parameter to one of the following:

- If the JMS connection alias used by the `pub.jms:send` service connects to the JMS provider using JNDI, set `JMSMessage/header/replyTo` to be the lookup name of the destination lookup object name.

If the JMS connection alias used by the `pub.jms:send` service connects to the JMS provider using a

native Broker connection, set `JMSMessage/header/replyTo` to the Broker queue name. That is, if the JMS connection alias specifies the Broker as the JMS provider and uses the native `webMethods` API to connect directly to the `webMethods` Broker, specify the name of the queue on the Broker that should receive replies to the message.

Note: When using the native `webMethods` API to connect to the Broker, the `JMSMessage/header/replyTo` destination must be a queue. Topics are not supported.

When executing the `pub.jms:send` service with a valid value for the `JMSMessage/header/replyTo` parameter, Integration Server creates the `javax.jms.Destination` and maps it to the `JMSReplyTo` field within the message header. Integration Server sends the message and returns immediately. The service does not wait for the response message.

If `JMSMessage/header/replyTo` is empty, then Integration Server does not set the `JMSReplyTo` header for the JMS message. If `JMSMessage/header/replyTo` is invalid, then Integration Server throws a `ServiceException`.

- **PIE-37261 (IS_9.5_SP1_Core_Fix9, IS_9.6_Core_Fix8)**
Integration Server logs the message “[B cannot be cast to com.wm.util.Values” in `server.log` file”. If the `ServiceResults` cache has the `Copy on Read` and `Copy on Write` check boxes selected and the number of entries in the cache exceeds the value specified for `Maximum Elements in Memory`, Integration Server logs the following error message to the server log: `[B cannot be cast to com.wm.util.Values`
This issue is now resolved.
- **PIE-37126 (IS_9.5_SP1_Core_Fix9, IS_9.6_Core_Fix9)**
In the Integration Server Administrator page, the `Routing` option under `Adapters` is not available in some cases.
After a successful installation of the `OnRamp for Commerce One Marketplace Adapter`, the `Routing` option under `Adapters` in the Integration Server Administrator page is not available. The Integration Server console is unable to add this option in the Integration Server Administrators page because an incorrect configuration in the `Server Log Facilities` page results in a failure in the initialization process for the adapter.
This issue is resolved. The `Routing` option is now available under the `Adapters` section in the Integration Server Administrator page.
- **PIE-35036 (IS_9.6_Core_Fix8, IS_9.7_Core_Fix5)**
Integration Server does not reflect the changes made to the `Create Temporary Queue` option after the JMS connection alias is created.
If the `Create Temporary Queue` option is selected while creating a JMS connection alias and if this selection is cleared later, Integration Server does not reflect this change. The setting of the `Create Temporary Queue` option that was chosen while creating a JMS connection alias is maintained.
This issue is resolved. Integration Server now reflects the changes made to the `Create Temporary Queue` option after the JMS connection alias is created.
- **PIE-37419 (IS_9.6_Core_Fix9)**
File polling consumed server threads that were not terminated at the end of file polling process. The thread leak related to file polling ports is now resolved.

- PIE-37581 (IS_9.6_Core_Fix9)
 Integration Server erroneously logs a message about the connection pool threshold being exceeded. Integration Server erroneously writes the following message to the server log:
 [ISS.0096.0008I] JDBC Connection pool threshold exceeded, 0 available for pool
 CentralUsers:CentralUsersPool
 This issue is now resolved.
- PIE-37893 (IS_9.6_Core_Fix9)
 The FTP command PWD returns incorrect structure in certain situations
 The FTP command PWD sometimes returns an incorrect directory path .
 The PWD command now returns a valid directory path.
- PIE-37268 (IS_9.6_Core_Fix9)
 When JSSE is enabled for outbound HTTP connections from Integration Server, `watt.security.ssl.client.ignoreEmptyAuthoritiesList` does not work as expected.
 When JSSE is enabled for outbound HTTP connections, Integration Server client accepts empty trusted authorities lists from the SSL server but does not return its CA certificate even if the `watt.security.ssl.client.ignoreEmptyAuthoritiesList` is set to true.
 This issue is resolved. Now, when JSSE is enabled for outbound HTTP connections, Integration Server client accepts empty trusted authorities lists from the SSL server and returns its CA certificate if the `watt.security.ssl.client.ignoreEmptyAuthoritiesList` property is set to true.
- PIE-36514
 Integration Server creates an exception when you invoke a flow service exposed as a web service. Integration Server creates the following exception when you try to invoke a flow service exposed as a web service:
 ServiceException: Fault returned by invoked service.
 This exception which is created in the stack before the service is executed is not logged in the Integration Server log but, monitoring tools like Dynatrace might interpret the exception as an error even though the service is successfully executed.
 This issue is resolved.
- PIE-37042
 When invoking the `pub.client:http` service with the `useJSSE` input parameter set to yes, a `java.lang.ClassCastException` error occurs if the server configuration parameter, `watt.security.ssl.keypurposeverification` is set to true.
 This issue is resolved.
- PIE-37056
 When the `pub.client:http` service sends an HTTPS request, the SSL handshake process fails when the `useJSSE` parameter is set to yes.
 When the `pub.client:http` service sends an HTTPS request and the `useJSSE` input parameter is set to yes, during the SSL handshake process Integration Server sends its certificate even though no matching CA certificate from the other endpoint server is found in the Certificate Authorities list for the SSL handshake.
 This issue is resolved. Integration Server now sends its CA certificate during the SSL handshake

process only if a matching CA certificate is available from the other endpoint server.

- PIE-37178

The `watt.server.compile` parameter is not set correctly if Integration Server instance is created through migration utility.

When running the migration utility, selecting the option to create an Integration Server instance during migration results in an incorrect value for the `watt.server.compile` server configuration parameter, specifically: `{javac.exe} -classpath {0} -d {1} {2}`

Now, the migration utility substitutes the value of `{javac.exe}` to point to `javac` executable located in the JDK shipped with Integration Server.

- PIE-37308

At start up, Integration Server logs a warning message about audit logging even though no audit loggers are configured.

At start up, Integration Server attempts to determine if there are sufficient JDBC connections in the JDBC connection pool even when audit loggers are not configured. If there are no connections, Integration Server logs the following message, which is erroneous if there are no configured audit loggers:

```
[ISS.0095.0022W] Audit Logging Initialization: Unable to verify the Max Connections setting for the Audit Logging database.
```

Now, Integration Server does not check for JDBC connections at start up if there are no configured audit loggers.

- PIE-37377

Update of scheduled tasks to use UTC fails.

At startup, if Integration Server updates scheduled tasks to use UTC (Coordinated Universal Time) instead of local time, Integration Server writes the following message to the `server.log`:

```
[ISS.0137.0035E] Migration of scheduled tasks to UTC timezone failed
```

When Integration Server updates the tasks to UTC, Integration Server also validates the tasks, which, for a task scheduled to run on another Integration Server in a cluster, includes checking that the Integration Server is a member of the cluster. If the Integration Server is not in the cluster, task validation fails and the task timestamp is not updated to UTC. However, Integration Server incorrectly considers that the task that failed validation is now in UTC when, in fact, the task is still in local time.

Now, the migration utility no longer validates the scheduled task during migration. Consequently a validation failure does not prevent the migration utility from updating the scheduled tasks.

- PIE-37457

A `webMethods` messaging request-reply fails when the encoding type of the reply document type is protocol buffers.

When encoding the reply message, Integration Server uses the encoding type specified for the publishable document type used as the reply. However, reply messages are sent to a dedicated reply-to channel on Universal Messaging. The channel is generic and is not associated with any specific document type. To receive a protocol buffer encoded message, a channel must be associated with a specific document type. Because the channel is generic, a reply message encoded as a protocol buffers cannot be processed.

To address this issue, Integration Server now always encodes reply message as IData.

- PIE-37560
Upgrade does not include layered products if the destination Integration Server does not have an instance with the same name as the source Integration Server.
When upgrading to an Integration Server instances that does not have the same name as the source Integration Server instance, upgrade does not include layered products such as webMethods Mediator and webMethods Monitor.
This issue is now resolved. Upgrade always includes layered products, regardless of whether or not the destination Integration Server instance is the same as the source Integration Server instance.
- PIE-37825
When the logging level is set to Trace, upon invoking the pub.client:http service, Integration Server logs an entry for the HTTP Get method in the server log irrespective of the HTTP request type.
This issue is resolved. Upon invoking the pub.client:http service, Integration Server now logs entries according to the HTTP request type. \
- PIE-37844
After a database outage, Integration Server can run out of threads.
When Integration Server cannot connect to the audit logging database, either due a problem in the database or a problem in the network, all threads requesting database connections can pause indefinitely. If this situation persists, all threads in the server thread pool will be paused. No new requests will be accepted.
This issue has been fixed. A point of thread contention in the JDBC pool implementation was removed. Threads requesting database connections after a database outage no longer experience lengthy pauses.
- PIE-36702 (IS_9.8_SPM_Fix1)
Command Central cannot be used to set the all the JDBC connection pool alias properties.
When using Command Central to create or edit a JDBC connection pool, the Available Connections Warning Threshold and Waiting Thread Threshold Count properties cannot be specified.
The Available Connections Warning Threshold and Waiting Thread Threshold Count properties can now be set through Command Central.
- PIE-34577
Integration Server does not validate a certificate that uses the SHA-256 algorithm but should.
The certificate chain verifier in Integration Server validates the certificates presented by SSL clients.
The certificate chain verifier does not validate a certificate chain that uses the SHA-256 algorithm, but should. Instead Integration Server throws the following error:
[ISC.0009.9002] Error in certificate chain: Entrust cannot verify the certificate chain: [ISC.0009.9002]
Error in certificate chain
The certificate chain verifier in Integration Server now validates a certificate chain that uses the SHA-256 algorithm.
- PIE-35422
The jcode utility fails with a MalformedPatternException on Linux.
Execution of jcode.sh on Linux fails with the following error: Exception in thread "main"

java.lang.NoClassDefFoundError: org/apache/oro/text/regex/MalformedPatternException

The exception occurs because one of the required jar files in common/lib/ext is missing from the jcode utility classpath.

The jcode.sh file is now updated to include the required jars in the classpath. This issue is now resolved.

- PIE-35643

After upgrading to version 9.7, when a webMethods messaging trigger, which has join conditions that are dependent on two or more documents, does not start because Integration Server cannot locate the publishable document types, Integration Server does not reload and start the trigger when the publishable document types are later loaded.

Integration Server does not start a webMethods messaging trigger, which has join conditions that are dependent on two or more publishable document types, if it cannot find the document types specified in the join condition. However, Integration Server issues a PDT_DOES_NOT_EXIST (InvalidDocumentException) exception stating the reason the trigger did not fully load and when Integration Server loads the package containing the publishable document types to which the trigger subscribes, Integration Server reloads the trigger. But, after upgrading to version 9.7, instead of the PDT_DOES_NOT_EXIST exception, Integration Server returns an INVALID (ServiceException) exception incorrectly. As a result, Integration Server fails to reload and start the trigger when the publishable document types are later loaded.

This issue is resolved. Integration Server now returns the PDT_DOES_NOT_EXIST exception correctly if it cannot find the document types specified in the join condition. In addition, when Integration Server loads the package containing the publishable document types specified in the join condition, Integration Server reloads the trigger.

- PIE-35716

If the database associated with Integration Server is restarted abruptly, service invocations or database calls made through JDBC connection pool alias fail and do not return any records from the database.

When the database associated with Integration Server is restarted abruptly, Integration Server does not close and release the existing JDBC connections that are stale due to the abrupt database shutdown, and fails to return these connections to the JDBC connection pool. As a result, service invocations or database calls made through the JDBC connection pool alias fail and do not return any records from the database.

The issue is now resolved.

- PIE-35907

After migrating to Integration Server 9.5 or later, Integration Server displays an exception when calling a web service and using MTOM streaming.

After migrating 2 to Integration Server 9.5 or later, sending a web service request that uses MTOM streaming for which more than one chunk is sent, the following exception occurs:

"Exception --> org.apache.axis2.AxisFault: Connection reset by peer: socket write error"

This issue is now resolved.

- PIE-36224

When installed on Unix or Linux, scripts located in

IntegrationServer/instances/<instanceName>/packages/<packageName> directory, such as IntegrationServer/instances/myInstance/packages/WmDeployer, do not have permission to execute. This issue is now resolved.

- PIE-36288
Executing the jcode.bat/sh utility removes the name of the output template assigned to a service. Running the jcode.bat/sh utility removes the value of the Output template Name property for services in the package containing the Java service. This issue is now resolved.
- PIE-36399
Using pub.schema:createXSD to create an XML Schema definition for a document type ends with a NullPointerException. Using the pub.schema:createXSD service to create an XML Schema definition for a document type ends with a NullPointerException if the document type contains a field named *body. This issue is now resolved.
- PIE-36644
Memory leak with JMS request-reply. Integration Server did not correctly close MessageProducer objects when invoking the pub.jms:reply service, resulting in a memory leak. This issue is now fixed.
- PIE-36824
Users that are members of the Developers group cannot use Designer to build and upload processes. When a user that belongs to the Developers group uses Designer to build and upload a process, Designer throws the following exception:
Error: Build of process <processName> failed.
com.webmethods.process.connection.is.IntegrationServerConnectionException: [ISC.0064.9314] Authorization Required: [ISS.0084.9004] Access Denied
at
com.webmethods.process.generator.util.GeneratorUtils.getJMSProviderData(GeneratorUtils.java:934)
at
com.webmethods.process.generator.util.GeneratorUtils.createJNDIDestinations(GeneratorUtils.java:718)
This issue is now resolved. Members of the Developers group can now use Designer to build and upload processes.
- PIE-36962
Integration Server fails to enable email ports configured to use transport layer security. Upon creating an email port configured to use transport layer security, Integration Server fails to enable the email port and issues the following error message:
Unsupported ciphersuite SSL_DH_DSS_WITH_3DES_EDE_CBC_SHA
This issue is resolved.

- PIE-36990

Duplicate port aliases are assigned during upgrade.

When upgrading Integration Server 9.5 or earlier to version 9.6 or later, Integration Server might assign the same port alias to multiple Internal Server ports. Upon startup of Integration Server, this may result in only one of the Internal Server ports becoming active and in the following warning:

```
[ISS.0070.30] yyyy-mm-dd hh:mm:ss zone WARN: Duplicate alias
protocolListener_portNumber_packageName encountered creating protocol listener on port
portNumber.
```

This occurs because the naming convention used to assign a port alias to an existing port creates a duplicate alias if more than one Internal Server ports connect to an Enterprise Gateway Server through the same Registration port.

Now, to ensure that each port alias is unique, Integration Server includes the host name for the port in the port alias for migrated ports. The new naming convention is:

```
protocolListener_portNumber _hostName_packageName
```
- PIE-37004

Integration Server reloads packages more than once in a run-time-based deployment.

When there are dependencies across different Integration Server packages in a run-time-based deployment, Integration Server reloads these packages more than once.

Now, Integration Server reloads the packages just once.
- PIE-37041

While converting an XML node to a document, Integration Server does not issue any validation errors upon receiving empty fields that are not included in the enumeration list.

When converting an XML node to a document that contains fields with *body attributes, the text specified in *body represents the values of the document fields. These elements can have valid values specified in the enumeration fields. However, while converting, if Integration Server receives an empty value for a field and if the empty value is not included in the enumeration list, Integration Server should issue a validation error, but does not.

This issue is resolved. Integration Server now issues the following validation error upon receiving an empty field that is not included in the enumeration list:

```
No matching enumeration value."
```
- PIE-37142

On Windows, the script for creating a new instance of Integration Server does not use the JVM bundled with the product installation.

The script for creating an Integration Server instance now uses the JVM bundled with the Integration Server product installation.
- PIE-37159

Users can log in to Integration Server with old and new passwords.

When central user management is configured for Integration Server and an authentication cache is enabled for Integration Server, users can log in with an old password even after successfully log in with the new password. This occurs because Integration Server does not remove the cached entry with the old password until after the time specified in `watt.server.auth.cache.timeout` elapses.

Now, once a user logs in with a new password, Integration Server clears the cached password.

- PIE-37166

Web browser users are not forced to supply credentials after their session expires. When Integration Server receives a request that includes a cookie that identifies a session and valid user credentials, Integration Server will do one of two things:

 - If the session identified by the cookie is valid, Integration Server will use the existing session for the request.
 - If the session identified by the cookie has expired, or is otherwise invalid, Integration Server will use the supplied credentials to authenticate the client and create a new session for the request.

Modern web browsers cache user credentials and send them to servers with each request. For Integration Server Administrator users, this means a user's session may expire but because the browser sends the user's credentials every time, a new session is created and the user may continue to use Integration Server Administrator without re-entering their user name and password. The only way to force the re-entry of a user's credentials is to close the browser. This may be considered a security flaw.

Integration Server addresses this situation by adding a new configuration parameter: `watt.security.session.forceReauthOnExpiration`. When set to true, Integration Server rejects any request that includes a cookie identifying an expired or invalid session, even if the request includes valid user credentials. The rejection response directs the browser to clear its session identifier and to prompt the user for credentials. When set to false, Integration Server creates a new session using the credentials in the cookie. The default value for `watt.security.session.forceReauthOnExpiration` is false. A value of true offers more secure behavior. Changes to `watt.security.session.forceReauthOnExpiration` take effect immediately.
- PIE-37211

The `pub.flow:getLastError` service does not return a `lastError` document.

The `pub.flow:getLastError` service does not return a `lastError` document if a parent SEQUENCE step specifies a timeout value but a service called by an INVOKE step within the SEQUENCE causes the SEQUENCE to time out.

This issue is now resolved.
- PIE-37222

A signed XML document or a node in an XML document fails verification performed by the `pub.security.xml:verifyXML` service.

If the `pub.security.xml:signXML` service signs an XML document or a node in an XML document and the service input specifies "true" for the `includeCertChain` parameter or a value other than the default value "X509_CERTIFICATE" for the `certData` parameter, the resulting signed document fails verification performed by the `pub.security.xml:verifyXML` service.

This issue is resolved.
- PIE-37276

After migrating a consumer web service descriptor that uses Web Services Addressing (WS-Addressing) from Integration Server 8.2 to a later release, invoking a web service connector in the descriptor ends with `SocketTimeoutException`.

If a consumer web service descriptor was created from a WSDL document that declared both the Web Services Addressing namespace as defined in the World Wide Web Consortium (W3C) Recommendation and the namespace from the earlier W3C Submission, after migration from

Integration Server 8.2, execution of a web service connector fails with the following:

java.net.SocketTimeoutException: Read timed out

The timeout occurs because Integration Server incorrectly uses the WS-Addressing namespace from the earlier W3C Submission of WS-Addressing when executing the web service connector. As a result, the web service provider does not respond in the expected manner which causes the request to time out.

Now, if a web service descriptor that uses WS-Addressing was created from a WSDL that declared namespace declarations for the WS-Address Recommendation and the Submission, Integration Server uses the namespace declarations defined in the Recommendation.

- **PIE-37473**
Package deployment fails when the package contains a document type.
Using Deployer to deploy a package that contains a publishable document type fails with the following error:
[DEP.0005.0326] An error occurred while deploying package "packageName". Item "documentTypeName" could not be loaded because of reason "[ISS.0026.9112] Document type documentTypeName has failed to load. Duplicate Broker document type name brokerDocumentTypeName is also referenced by documentTypeName". The package may have partially loaded; please check your target server.
This issue is now resolved.
- **PIE-37508**
Integration Server logs an erroneous exception when executing web services without an outbound callback service.
When executing web service that is part of a provider web service descriptor for which an outbound callback service is not specified, Integration Server sometimes writes the following erroneous message to the error log:
ISC.0088.9998E Exception --> null.
This issue is now resolved. Integration Server no longer logs an exception when executing a web service that is part of a web service descriptor that does not specify an outbound callback service.
- **PIE-37589**
XSLT services experience intermittent XSLT transformation errors or exhibit slow transformation performance.
This fix addresses both issues.
- **PIE-37790**
Enhancement to use Universal Messaging for audit logging queues.
In asynchronous audit logging, Integration Server first writes a log entry to a queue and subsequently writes the log entry from the queue to the log destination. Previously, Integration Server provided only an internal queue, sometimes called the light-weight queue, to use as an audit logging queue. With this enhancement, Integration Server provides the option of using a Universal Messaging queue instead of the internal queue as the audit logging queue. Using Universal Messaging with audit logging offers increased performance for asynchronous and synchronous logging.

- PIE-37834

When using the jcode utility to compile Java services, Integration Server generates wrong jcode utility classpath.

When using the jcode utility to compile Java services, Integration Server generates wrong jcode utility classpath if the code or classes directory added to the classpath does not include path separator.

This issue is now resolved.
- PIE-37977

A web service connector throws a SOAPException about an invalid envelope.

When executing a web service connector for an InOnly or InOnlyRobust operation for which the parent consumer web service descriptor has the Pre-8.2 compatibility mode property set to true, Integration Server may log the following extraneous error message indicating an invalid SOAP Envelope has been received even though the connector executed successfully.

[ISS.0088.9155] this SOAPMessage does not contain a valid Envelope object

This fix eliminates the extraneous error message.
- PIE-38099

A trigger with an AND join fails when receiving messages concurrently across a cluster of Integration Servers or a non-clustered group of Integration Servers.

When a JMS trigger or a webMethods messaging trigger with an AND join receives messages at the same time on two different Integration Servers, one of the messages might be lost. The lost message will not be processed which prevents the AND join from being satisfied and causes the trigger to fail. This situation occurs when the trigger resides on Integration Servers in a cluster or on a group of non-clustered Integration Servers.

This issue is now resolved.
- PIE-38134

Enhancement to provide the ability to configure the allowed protocols for JSSE on a per port basis. In Integration Server, the `watt.net.jsse.server.enabledProtocols` server configuration parameter specifies the allowed protocols for all JSSE ports. However, there might be times where you wish to allow specific protocols for use with a particular JSSE port only. Integration Server now provides the ability to specify the allowed protocols for JSSE on a per port basis.

Note: The 'jsseEnabledProtocols' value specified for the port record in the `listeners.cnf` file overrides the value set by `watt.net.jsse.server.enabledProtocols` server configuration parameter.

Note: When the logging facility 0006 Server SSL Interface is set to the Debug logging level, Integration Server writes messages about protocols used for inbound and outbound ports to the server log. At the Trace logging level, Integration Server writes messages about the enabled cipher suites. You can use these server log messages to confirm the enabled protocols for any JSSE port.
- PIE-38158

The `pub.utils.messaging:migrateDocTypesTriggersToUM` service incorrectly returns a message indicating a filter was successfully converted.

When the `pub.utils.messaging:migrateDocTypesTriggersToUM` service cannot convert a filter for a webMethods messaging trigger because of a filter conversion error, the service returns the error message in the `failedTriggers/reason` parameter. However, the service also returns a message

stating that the filter was successfully converted, which is not correct.

If a filter cannot be converted successfully, the

`pub.utils.messaging:migrateDocTypesTriggersToUM` no longer returns a message about successful conversion of the filter.

- **PIE-38244**
Web service fails with a `RampartException` while handling a holder-of-key SAML assertion. When handling a holder-of-key SAML assertion, a web service fails with the following exception: `RampartException: Invalid signature algorithm for Asymmetric binding`. This issue is now resolved.
- **PIE-38300**
A concurrent JMS trigger makes a retry attempt after Integration Server suspends the trigger because of a transient error. When a concurrent JMS trigger encounters a transient error when processing a message, the trigger makes an extra retry attempt after the trigger is suspended. For example, if Max retry attempts is 3, Integration Server suspends the trigger after the third retry attempt fails. However, the trigger makes a fourth retry attempt. This issue is resolved. When Integration Server suspends a concurrent JMS trigger because of a transient error, the trigger does not make an additional retry attempt.
- **PIE-38391**
In a clustered environment, Integration Server sometimes does not create child tasks when a new server is added to the cluster or when an existing server is restarted. When a task is scheduled to run on all servers in a clustered environment, Integration Server creates a parent task and a child task for each server in the cluster. When a new server is added to the cluster or when an existing server in the cluster is restarted, Integration Server creates a corresponding child task upon server restart. However, Integration Server sometimes does not create the child task for the newly added server or for the server that was restarted. As a result, the complete information for all servers in the cluster is not available on the Scheduler screen. This issue is resolved.
- **PIE-38404**
Integration Server resumes document retrieval and/or processing for webMethods messaging triggers after package reload or server restart even when the Apply Change Permanently option was selected. When using the Integration Server Administrator or the built-in services for suspending document retrieval and/or processing for a webMethods messaging trigger that receives documents from Universal Messaging, Integration Server does not honor the value of the Apply Change Permanently check box. When selected, the Apply Change Permanently check box indicates that Integration Server persists the change in document retrieval or document processing across package reloads and server restart. However, Integration Server reverts the state and enables retrieval and/or processing of documents when Integration Server restarts or when a package is reloaded. This issue is resolved.

- PIE-38473

While creating an Enterprise Gateway Server port that uses the HTTPS protocol, the 'Use JSSE' option is not available in the Edit Enterprise Gateway Server Configuration screen in Integration Server Administrator.

This issue is resolved. The 'Use JSSE' option is now available in the Edit Enterprise Gateway Server Configuration screen while creating an Enterprise Gateway Server port that uses the HTTPS protocol.
- PIE-38526

When creating an Internal Server port that uses the HTTPS protocol, the "Use JSSE" option is not available in the Edit Internal Server Configuration screen in Integration Server Administrator.

This issue is resolved. The Edit Internal Server Configuration screen now includes the "Use JSSE" option when the selected protocol is HTTPS.
- PIE-38530

An HTTPS port that uses JSSE fails to start when a keystore is specified.

If Use JSSE is set to Yes for an HTTPS port and the key alias password is different from the password used for the keystore, the port does not start.

This issue is now resolved.
- PIE-38536

The pub.client:smtp service completes successfully even though the service finishes with errors. Improper exception handling allowed the pub.client:smtp service to execute successfully even though the service encountered errors. This could lead to missing MIME parts in the email sent by the service.

Now, the pub.client:smtp service includes proper exception handling, which prevents successful completion of the service when the service encounters errors.
- PIE-38557

When entering quiesce mode, Integration Server writes messages to the client-side queue because Integration Server disables connection aliases before disabling packages.

When Integration Server enters quiesce mode, Integration Server disables the webMethods messaging connection aliases and JMS connection aliases before disabling packages. As a result, services that publish messages might execute after the needed connection alias is disabled, causing messages to be written to the client-side queue.

Now, when entering quiesce mode, Integration Server disables packages before disabling connection aliases. This will prevent new messages from being written to the client-side queue as Integration Server enters quiesce mode.
- PIE-38660

The jcode utility does not scan the jar files present in the static folders of the packages.

The jar files present in the static folders of the packages (package's code\jars\static folders) were not being scanned and added to the classpath by the jcode utility.

This issue is resolved.
- PIE-38881

When migrating from Integration Server 8.2 or later to a newer version of Integration Server, the migration utility overwrites server configuration parameter values set during an earlier migration. If you migrated Integration Server from version 7.x to version 8.2 or later and the later version introduces new behavior for existing functionality, the migration utility prompts you to choose the new behavior or the existing behavior and then sets a server configuration parameters in the server.cnf accordingly. When you migrate Integration Server from 8.2 or later to a newer version of Integration Server, the migration utility prompts to use the new or existing behavior for existing functionality that was changed in the new version. However, the Integration Server migration utility incorrectly overwrites some of the settings that were already set during earlier migrations. This issue is now resolved. Integration Server migration utility retains previously selected behavior and sets server configuration parameters for behavior introduced in the new version only.

- PIE-38921
Setting the `watt.server.db.connectionCache` server configuration parameter to `server` increases the number of database connections.
Setting the `watt.server.db.connectionCache` server configuration parameter to `server` tells Integration Server to maintain a pool of connections for each database. However, when a connection became stale, a new connection was getting created without the stale connection getting closed. This increased the number of connections even though the pool reached its maximum number of connections.
This issue is resolved. Integration Server now closes all the stale connections before creating new connections.

Release 9.9

- PIEAR-616 (WAR_9.5_SP1_Fix5, WAR_9.7_Fix1)
The change in status of an adapter listener is not displayed in Integration Server Administrator if you query the listener using an existing search filter.
In Integration Server Administrator, if you query an adapter listener using an existing search filter, the status of the listener is not refreshed and the current status of the listener is not displayed correctly.
This issue is resolved. The status of an adapter listener is now refreshed on access to display the correct status using an existing search filter.

Release 9.8

- PIE-31623 (IS_7.1.2_Core_Fix47, IS_7.1.3_Core_Fix27, IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3, IS_9.7_Core_Fix1)
Sessions on remote servers are reused by different users. When Integration Server is acting as an HTTP client, it retains the HTTP cookies from the remote server. If two users connect to the same remote server, they use the same set of cookies. Since the session ID is often stored in a cookie, this can result in the two users using the same session on the remote server. Each user should use a unique session when connecting to a remote server.
This issue is resolved. HTTP cookies are now associated with a specific user as well as the remote server so that a separate set of cookies are used for each user.

- PIE-31372 (IS_7.1.3_Core_Fix27, IS_9.0_SP1_Core_Fix7, IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)
 Secure LDAP connections fail when running Integration Server on HP-UX.
 When running Integration Server on HP-UX, configuring LDAP connections using SSL fails because the default secure random algorithm in the Entrust library fails with an exception. This issue is resolved. Integration Server introduces the new `watt.net.ssl.randomAlgorithm` server configuration parameter to identify the random algorithm name. The default is `FIPS186_2usingSHA1`. If you change the setting of this parameter, you must restart Integration Server for the changes to take effect.

- PIE-34054 (IS_7.1.3_Core_Fix27, IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)
 Remove use of SSLv3 from any HTTPS or FTPS Integration Server ports.
 In order to protect against POODLE vulnerability (CVE-2014-3566), this fix exposes server configuration parameters that allow you to disable the use of SSLv3.0 on Integration Server HTTPS and FTPS ports.

Depending on whether connections use the Entrust library (`entoolkit.jar`) or JSSE (where `useJSSE=true`), you use a different procedure to disable SSLv3.0. Follow the appropriate procedure as follows:

For connections that use Entrust (`entoolkit.jar`) library:

When Integration Server uses the Entrust library to handle inbound and outbound requests, you disable SSLv3.0 by setting the following server configuration parameters:

- `watt.net.ssl.server.handshake.minVersion`
- `watt.net.ssl.server.handshake.maxVersion`

Possible values for these server configuration parameters are `"sslv3"` and `"tls"` (the default). With this fix, these two parameters take the default value `"tls"`, which indicates that all server side SSL listeners will support only TLSv1 and no longer accept SSLv3 connections.

When Integration Server acts as a client and makes an outbound request, it configures the allowed protocols using the following server configuration parameters:

- `watt.net.ssl.client.handshake.minVersion=sslv2`
- `watt.net.ssl.client.handshake.maxVersion=tls`

Possible values for these server configuration parameters are `"sslv2"`, `"sslv3"`, and `"tls"`. If you want to disable the use of `"sslv3"`, set `watt.net.ssl.client.handshake.minVersion` as follows:

`watt.net.ssl.client.handshake.minVersion=tls`

To change the values of the server configuration parameters, from Integration Server Administrator, navigate to Settings > Extended and add the parameters as follows:

`watt.net.ssl.server.handshake.minVersion=tls`
`watt.net.ssl.server.handshake.maxVersion=tls`
`watt.net.ssl.client.handshake.minVersion=tls`
`watt.net.ssl.client.handshake.maxVersion=tls`

If any of your clients require SSLv3 to connect (the previous default), set

`watt.net.ssl.server.handshake.minVersion` as follows:

`watt.net.ssl.server.handshake.minVersion=sslv3`

When making outbound connections, you can configure Integration Server to first try to connect using `sslv3` and, if that fails, to use `tlsv1`, set `watt.net.ssl.client.handshake.minVersion` as follows:

watt.net.ssl.client.handshake.minVersion=sslv3

This will allow Integration Server to use sslv3 with endpoints that do not support tlsv1.

For connections that use JSSE (where useJSSE=true):

When Integration Server uses JSSE to handle inbound and outbound requests, you disable SSLv3.0 by setting the following server configuration parameters:

- watt.net.jsse.server.enabledProtocols

- watt.net.jsse.client.enabledProtocols

Possible values for these server configuration parameters are a comma-separated values consisting of one or more of the following:

- SSLv2Hello

- SSLv3

- TLSv1

- TLSv1.1

- TLSv1.2

With this fix, watt.net.jsse.server.enabledProtocols and watt.net.jsse.client.enabledProtocols are set to the default value of "TLSv1,TLSv1.1,TLSv1.2", which indicates that all server side SSL listeners and client side outbound connections that use JSSE will not accept any SSLv3 or SSLv2 connections. To change the values of the parameters, from Integration Server Administrator, navigate to Settings > Extended and add the parameters as follows:-

watt.net.jsse.server.enabledProtocols=TLSv1,TLSv1.1,TLSv1.2

watt.net.jsse.client.enabledProtocols=TLSv1,TLSv1.1,TLSv1.2

Note: These values are case-sensitive. Specify the values exactly as shown.

If any of your clients need to connect using SSLv3, add SSLv3 to

watt.net.jsse.server.enabledProtocols, for example:

watt.net.jsse.server.enabledProtocols=TLSv1,TLSv1.1,TLSv1.2,SSLv3

When starting JSSE ports, at DEBUG level of logging facility 6 (Server SSL Interface), Integration Server logs a message to indicate what protocols are enabled for each JSSE port.

- PIE-34463 (IS_7.1.3_Core_Fix27, IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
When you invoke the pub.client:http service, Integration Server does not create a new session when the newSession parameter is set to yes.
This issue is resolved.
- PIE-33340 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3, IS_9.7_Core_Fix1)
While deploying BPM (ProcessModel) assets using webMethods Deployer, Integration Server issues an exception if a specified package does not exist in the source server.
While creating a deployment map and adding it to the target group to deploy BPM (ProcessModel) assets using webMethods Deployer, Integration Server issues a NullPointerException if a specified package does not exist in the source Integration Server.
This issue is resolved. In the above scenario, Integration Server now issues an error message stating that the specified package does not exist in the source server.
- PIE-32686 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)
Integration Server returns a 401 Unauthorized error when it receives a request with no authentication scheme specified in the Authorization header.
Requests with no authentication scheme do not conform to the HTTP Authorization header

specification. Integration Server versions prior to 8.2 CoreFix 12 did not strictly enforce the HTTP Authorization header specification, so requests from HTTP clients that did not conform to that standard still worked. However, starting with Integration Server version 8.2 CoreFix 12, Integration Server forces requests to have an authentication scheme in the Authorization header. This issue is resolved. Now, Integration Server treats requests with no authentication scheme in the Authorization header as BASIC authentication requests and performs authentication for valid credentials.

- PIE-32800 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)
In certain scenarios, SOAP fault that Integration Server generates includes the hostname and IP address of the endpoint reference.
Upon invoking a web service that does not exist in the Integration Server, the SOAP fault that Integration Server generates includes the hostname and IP address of the endpoint reference in the values specified for fault reasons and actor/role attributes.
This issue is resolved. This fix introduces a new server configuration parameter, `watt.server.SOAP.hideEPRHostInFault`, to hide the endpoint reference host name and IP address details in the SOAP fault. When this parameter is set to true, Integration Server replaces the host name and IP address with `*:*` in the SOAP fault. When this parameter is set to false, Integration Server includes the host name and IP address details of the endpoint reference in the SOAP fault. The default value of this parameter is false.
Note: This parameter applies only when the Pre-8.2 compatibility mode property of the web service descriptor is set to false.
- PIE-32920 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)
Integration Server returns the wrong error information when the format of an incoming HTTP Authorization header value is invalid.
When Integration Server receives a request whose HTTP Authorization header value is in an invalid format, it returns a 500 Internal Server Error. Integration Server should return a 401 Unauthorized Error.
This issue is resolved.
- PIE-33354 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2) Integration Server logs an exception in the server logs while invoking web services that has no value set for the Outbound callback service property.
While invoking certain web services, Integration Server logs following message in the server logs:
ISC.0088.9998E Exception --> null.
This issue occurs only if no value is set for the Outbound callback service property of the web service descriptor.
This issue is now resolved. Integration Server no longer logs an exception if a value is not set for the Outbound callback service property of the web service descriptor.
- PIE-33421 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)
After applying IS_9.0_SP1_Core_Fix6, the Dynamic Server Pages in Integration Server Administrator are not cached.
This issue is resolved. This fix introduces a new server configuration parameter, `watt.cachedirective.exclude.packages` that you can use to specify a comma-separated list of

packages whose Dynamic Server Pages you want the browser to cache. You can specify the packages as regular expressions. The asterisk (*) is the only wildcard character allowed in the regular expression. By default, the value of this parameter is empty, meaning none of the Dynamic Server Pages in the Integration Server Administrator are cached.

Note: Software AG recommends that you use `watt.cachedirective.exclude.packages` to cache the Dynamic Server Pages that are related to custom packages only.

Important: If you change the setting of this parameter, you must restart Integration Server for the changes to take effect.

- PIE-31376 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3)
Querying for a user in an LDAP directory results in a "size limit exceeded" error.
When Integration Server is configured to use LDAP for external user management, and Integration Server Administrator is used to query LDAP for a specific LDAP user or group of users, Integration Server retrieves all users from the LDAP directory instead of retrieving only the specified user(s). This behavior results in the following error in the Integration Server server log:
ISS.0002.0044E Error querying for user(s) on (LDAP server) with search root (LDAP: error code 4 - Sizelimit Exceeded)
This issue is resolved.
- PIE-32725 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3)
In certain scenarios, the entries in the stats.log file become corrupted.
This issue is now resolved.
- PIE-33193 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3)
Even after modifying the database columns to make them wider, Integration Server truncates the length of data in the `WMSERVICEACTIVITYLOG.FULLMESSAGE` and `WMSERVICECUSTOMFLDS.STRINGVALUE` columns of the audit logging database. The `WMSERVICEACTIVITYLOG.FULLMESSAGE` and `WMSERVICECUSTOMFLDS.STRINGVALUE` columns of the audit logging database are hard-coded in Integration Server to be 1024 characters and 512 characters respectively. When you edit the width of the database columns to make them wider, Integration Server truncates the data to the hard-coded values before writing the data to the database table.
This issue is resolved. You can now increase the widths of the `WMSERVICEACTIVITYLOG.FULLMESSAGE` and `WMSERVICECUSTOMFLDS.STRINGVALUE` columns in the audit logging database. Now, when the audit records corresponding to the `WMSERVICEACTIVITYLOG` and `WMSERVICECUSTOMFLDS` database tables are created during server initialization, Integration Server checks the database to determine the width of the `WMSERVICEACTIVITYLOG.FULLMESSAGE` and `WMSERVICECUSTOMFLDS.STRINGVALUE` columns, and sets the length accordingly.
Keep the following points in mind when increasing the column widths:
 - The data written to these columns can contain multibyte characters. With multibyte characters, data can be truncated in the middle of a character, which could cause the last character written to the database column to be a character other than the intended character. To avoid this, Integration Server truncates the last character boundary before the maximum length of the field, meaning that the data might be slightly smaller than the maximum value set in the audit logging database.
 - Integration Server checks the database for column width by obtaining the metadata and

examining the CHAR_OCTET_LENGTH field of the column. If the database vendor does not supply a CHAR_OCTET_LENGTH value for the column, Integration Server uses the hard-coded default lengths for the fields as follows:

- WMSERVICEACTIVITYLOG.FULLMESSAGE = 1024
- WMSERVICECUSTOMFLDS.STRINGVALUE = 512

Integration Server introduces the following new server configuration properties:

- The `watt.server.audit.schemaName` server configuration parameter specifies the user name of the ISCoreAudit JDBC functional alias. By default, Integration Server uses the user name from the ISCoreAudit JDBC functional alias.

- The `watt.server.audit.dbEncoding` server configuration parameter specifies the character set that the audit logging database uses. The default is UTF-8. The value for this property must be a standard IANA-assigned character set name, as defined on

<http://www.iana.org/assignments/character-sets>.

If you change the setting of either of these server configuration parameters, you must restart Integration Server for the changes to take effect.

- PIE-33250 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3)
Settings within a package's `node.ndf` files are changed after the package is compiled during a repository-based deployment.
When a package is deployed to a target Integration Server using webMethods Deployer, and the Compile Package on Install global value is set to true, Integration Server performs a fragmentation step during compilation of the package. This step creates new `node.ndf` files for any Java services contained in the package. As a result, any settings previously defined in the `node.ndf` files are lost. This issue is resolved. This fix introduces a new global property - `fragPackage` (Fragment Package on Install), for package assets. A value of true indicates that you want to allow Integration Server to perform the fragmentation step and overwrite a package's `node.ndf` files when the package is compiled on the target server. A value of false indicates that you want to prevent the server from performing the fragmentation step, thereby preserving custom settings defined in the package's `node.ndf` files when the package is compiled on the target server. The default is true.
- PIE-33256 (IS_9.0_SP1_Core_Fix7, IS_9.6_Core_Fix3) Integration Server gives the datatype of the `returnCode` output parameter of the `pub.client.sftp*` services incorrectly as integer instead of correctly specifying it as string.
This issue is now resolved. The `returnCode` output parameter of the `pub.client.sftp*` services is now updated to string datatype.
- PIE-32235 (IS_9.0_SP1_Core_Fix7) Integration Server thread dump indicates a deadlock state between webMethods Broker polling and synchronization threads when Broker disconnects while a publishing event is in progress.
The BrokerPoller thread, which determines Broker's connection status, and the BrokerSynchronizer thread, which synchronizes documents between Integration Server and Broker, may become deadlocked. This can happen when Broker disconnects and/or reconnects while Integration Server is publishing a document to Broker.
This issue is resolved. Integration Server now prevents deadlocks from occurring between the two threads.

- PIE-32458 (IS_9.0_SP1_Core_Fix7)
 Integration Server support for using user-defined prefixes with the document types created when creating or refreshing a consumer web service descriptor or a WSDL first provider web service descriptor.
 This fix contains the server-side functionality required to support the Designer functionality described in WED-4598 which is available in a webMethods Service Development fix. Together, PIE-32458 and WED-4598 provide the ability to associate user-defined prefixes with the namespaces used in document types instead of the prefixes defined in the schema contained in or referenced by a WSDL document used as the source for a web service descriptor. For more information about this functionality, see the readme file for the webMethods Designer Service Development fix. For webMethods Service Development 9.6, WED-4598 is available in ESB_9.6_Fix2.
 Note: The ability to refresh a web service descriptor is available in Integration Server and webMethods Service Development versions 9.0 SP1 and later.
- PIE-32990 (IS_9.0_SP1_Core_Fix7)
 Copying and pasting Integration Server packages from one server to another results in a "no such remote server" error.
 When a package is copied from an Integration Server and pasted to a remote Integration Server using Software AG Designer, the following error message is displayed in the server log:(ISS.0085.9102) No such remote server
 This occurs even when the target server is running normally and has a valid remote server alias. This issue is resolved.
- PIE-32895 (IS_9.0_SP1_Core_Fix8, IS_9.5_SP1_Migration_Fix1, IS_9.6_Core_Fix3, IS_9.6_Migration_Fix2)
 Integration Server issues a java.lang.NullPointerException error while migrating from Integration Server version 8.0 SP1 to Integration Server version 9.0 SP1 or 9.5 SP1 in a different installation directory.
 This issue is now resolved.
- PIE-30419 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix3, IS_9.7_Core_Fix1)
 Editing the connection details of a remote Integration Server does not work as expected.
 After a remote Integration Server connection is created and a successful connection is established, if the remote server connection settings are changed (for example, if a different port number is specified), Integration Server does not create a new session when subsequent connection attempts are made. As a result, Integration Server Administrator indicates that subsequent connections are successful, even if the revised remote server settings are incorrect (for example, if the revised port number is not valid).
 This issue is resolved.
- PIE-32999 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix3, IS_9.7_Core_Fix1)
 After applying a fix containing PIE-30630 or when using Integration Server 9.7, if Platform Manager is in use, Integration Server can have a high number of active stateful sessions.
 The number of active stateful sessions becomes high because sessions created by the Platform Manager were not being cleaned up upon disconnect but were not being removed from memory until the session time out limit elapsed. This issue was introduced by PIE-30630 which is included

in Integration Server 9.7, IS_8.2_SP2_Core_Fix13, IS_9.0_SP1_Core_Fix6, IS_9.5_SP1_Core_Fix3, IS_9.6_Core_Fix1, and IS_9.6_Core_Fix2.

Integration Server now removes a session created by the Platform Manager from memory at the time the Platform Manager disconnects.

- PIE-31474 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)

Integration Server creates additional session threads while executing triggers and displays a misleading expiration value.

When `watt.server.trigger.reuseSession` (for Broker/local triggers in Integration Server version 9.0 SP1 and earlier or `webMethods` messaging triggers in version 9.5 SP1 and later) or `watt.server.jms.trigger.reuseSession` (for JMS triggers) is set to `false`, Integration Server displays an additional session thread for the trigger. In addition, regardless of the values set for `watt.server.trigger.reuseSession` and `watt.server.jms.trigger.reuseSession` server configuration parameters, Integration Server Administrator displays the session on the Server > Statistics > Session screen with a Session Expires value of 9223372036854775807 and the session never expires. This issue is resolved. When either the `watt.server.trigger.reuseSession` or `watt.server.jms.trigger.reuseSession` server configuration parameter is set to `false`, Integration Server no longer creates extra sessions for the respective trigger. For these sessions, Integration Server Administrator displays "upon service completion" in the Session Expires column on the Server > Statistics > Sessions screen for trigger-related sessions. This indicates that the session will remain until the service executed by the trigger finishes. Once the service finishes, Integration Server removes the session.

When either the `watt.server.trigger.reuseSession` or `watt.server.jms.trigger.reuseSession` server configuration parameter is set to `true`, Integration Server shares one session for each execution of the trigger. In this case, Integration Server Administrator displays a value of "never" in the Session Expires column of the Server > Statistics > Sessions screen and the session does not expire.

- PIE-33336 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)

In case of web services, Integration Server does not keep the undeclared namespaces in the resulting document (IData), when the endpoint service is invoked.

If an XML element that is part of a SOAP Message has elements with attributes containing undeclared namespaces, Integration Server does not keep the undeclared namespaces in the resulting document (IData), when the endpoint service is invoked.

This issue is resolved. The generated IData now includes the undeclared namespace definitions.

Note: In case of web services, the undeclared namespace functionality is available only if the datatype of the document is of type 'Object'.

- PIE-33632 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)

The service results cache does not delete expired elements from Ehcache.

Expired service results remain in Ehcache even after the cache sweeper thread runs.

This issue is resolved.

- PIE-33376 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix3)

Integration Server issues a `NullPointerException` when it receives an HTTP request with no User-Agent header.

This issue is resolved.

- PIE-33601 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
Integration Server is unable to process incoming XML files if the XML files contain non-English characters, such as Greek or Japanese.
While processing an incoming XML file, Integration Server uses the charset encoding defined in the XML header. If the encoding is not defined in the XML header, Integration Server processes the XML file using the charset encoding of the request or the character encoding specified in the `watt.server.fileEncoding` server configuration parameter. However, after migrating to higher versions, using the character encoding specified in `watt.server.fileEncoding` parameter to process an incoming XML file causes Integration Server to return garbled messages.
This issue is resolved. To resolve this issue, this fix introduces a new server configuration parameter, `watt.server.xml.encoding`, that specifies the encoding that Integration Server must use when processing incoming XML files.
If an encoding is not defined in the XML header, Integration Server attempts to process the XML file using the charset encoding of the http or ftp request. If charset encoding is not available in the request header, then Integration Server uses the character encoding specified in the `att.server.xml.encoding` server configuration parameter. There is no default value for this parameter. You must restart Integration Server for changes to this parameter to take effect.
Note: If you have configured Integration Server to use the character encoding specified in the `watt.server.fileEncoding` parameter to process incoming XML files, after installing this fix or after upgrading to a higher version of Integration Server, ensure that the value of `watt.server.fileEncoding` parameter is set to the same value specified for `watt.server.xml.encoding`. If you have not configured `watt.server.fileEncoding` for processing XML files previously, after installing this fix or after upgrading to a higher version of Integration Server, you can configure `watt.server.xml.encoding` to process incoming XML files. You can use `watt.server.fileEncoding` to process all files other than incoming XML files.

- PIE-33615 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
Integration Server is unable to parse incoming XML files if the XML files contain non-English characters, such as Greek or Japanese.
While processing an incoming XML file, Integration Server uses the charset encoding defined in the XML header. If the encoding is not defined in the XML header, Integration Server parses the XML file using the charset encoding of the request or the character encoding specified in the `watt.server.fileEncoding` server configuration parameter. However, after migrating to higher versions, using the character encoding specified in `watt.server.fileEncoding` parameter to process an incoming XML file causes Integration Server to return garbled messages.
This issue is resolved. To resolve this issue, this fix introduces a new server configuration parameter, `watt.server.xml.encoding`, to specify the encoding that Integration Server must use when processing incoming XML files.
If an encoding is not defined in the XML header, Integration Server attempts to process the XML file using the charset encoding of the http or ftp request. If charset encoding is not available in the request header, then Integration Server uses the character encoding specified in the `watt.server.xml.encoding` server configuration parameter. There is no default value for this parameter. You must restart Integration Server for changes to this parameter to take effect.
Note: If you have configured Integration Server to use the character encoding specified in the `watt.server.fileEncoding` parameter to process incoming XML files, after installing this fix or after

upgrading to a higher version of Integration Server, ensure that the value of `watt.server.fileEncoding` parameter is set to the same value specified for `watt.server.xml.encoding`.

- PIE-33637 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
Integration Server generates an invalid WSDL document if it encounters identically named elements that are declared to be of different types but belonging to the same target namespace. When generating the WSDL document, if Integration Server encounters identically named elements that are declared to be of different types but belonging to the same target namespace, Integration Server cannot correctly represent this in a WSDL document. Instead, when generating the WSDL document, Integration Server replaces the element declaration it creates for the identically named element first with the element declaration that it generates last.
With this fix, when creating a service first web service descriptor, Integration Server issues a warning message stating that the element declaration it creates for the identically named element first is replaced with the element declaration that it generates last.
- PIE-33762 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
The `pub.cache.atomic:putIfAbsent` and `pub.cache.atomic:replaceIfKeyExists` services show the incorrect data type for the `oldValue` output parameter.
The `pub.cache.atomic:putIfAbsent` and `pub.cache.atomic:replaceIfKeyExists` services display the `oldValue` output parameter as a `String` data type, but should display the data type as `Object`. This issue is resolved.
- PIE-33799 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
When running flow services in debug mode, Integration Server cannot retrieve parent service information from a calling child service.
Integration Server cannot retrieve the parent service information from a calling child service in debug mode.
This issue is resolved. Integration Server introduces the new public service `pub.flow:getCallingService` that can be invoked from the calling child service and returns the service and package name of the parent service. The `pub.flow:getCallingService` service returns the following output parameters:
 - `svcName` - `String` (optional) Fully qualified namespace name of the parent service.
 - `pkgName` - `String` (optional) Package name of the parent service.If `pub.flow:getCallingService` is invoked from a service that does not have a parent service (for example, if the service is a top level service), Integration Server does not return the output parameters.
- PIE-33938 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
Integration Server continues to execute a flow service even after global variable substitution fails. When a flow service expects an input through global variable substitution and if the expected global variable does not exist, Integration Server displays an error message and continues executing the flow service instead of stopping the service execution.
The issue is resolved. If the global variable that a flow service expects does not exist, Integration Server displays an error message and stops the flow service execution.

- PIE-34080 (IS_9.0_SP1_Core_Fix8, IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
 Adding an HTTP or HTTPS port in the Integration Server Administrator results in a blank Security > Ports screen.
 When the server configuration parameter `watt.server.http.listRequestVars` is set to "error" and a new HTTP or HTTPS port is added, Integration Server returns a blank Security > Ports screen and writes the following message to the server log:
 (ISC.0038.2) Duplicate query tokens found in URI: threadPool
 This issue is resolved.
- PIE-33821 (IS_9.0_SP1_Core_Fix8)
 The `watt.server.SOAP.hideEPRHostInFault` server configuration parameter is not available in the Extended Settings page in Integration Server Administrator.
 This issue is resolved. The `watt.server.SOAP.hideEPRHostInFault` server configuration parameter now appears in the Extended Settings page.
- PIE-32771 (IS_9.6_Core_Fix3, IS_9.7_Core_Fix1)
 JMS provider connection attempts are refused during the processing of messages.
 When a trigger service uses a transacted JMS connection alias, and that service sends JMS messages using the `pub.jms:send` service to a different transacted connection alias, the sessions on the JMS provider are not closed. As a result, connection attempts are refused because channel and process limits are reached.
 This issue is resolved.
- PIE-33468 (IS_9.6_Core_Fix3, IS_9.7_Core_Fix1)
 Web service descriptor does not use user-defined prefixes for fault and handler document types.
 When creating a consumer web service descriptor or a WSDL first provider web service descriptor, the user creating the web service descriptor can specify namespace prefixes to replace the prefixes used in the WSDL document. However, Integration Server was not using the user-defined prefixes for the IS document types generated for fault or handlers messages.
 This issue is now resolved.
- PIE-33531 (IS_9.6_Core_Fix3, IS_9.7_Core_Fix1)
 After installing a newly generated license file for Terracotta, Integration Server disables editing for BigMemory settings.
 The issue occurs because of a parsing error in the license file.
 This issue is resolved.
- PIE-33660 (IS_9.6_Core_Fix3, IS_9.7_Core_Fix1)
 Integration Sever does not acknowledge all of the messages received in a batch by a JMS trigger.
 When a standard JMS trigger receives and processes a batch of messages from the JMS provider, the JMS trigger should acknowledge the last message received in the batch. This results in acknowledgement of all the messages received in the batch. However, the JMS trigger currently acknowledges the first message received in a batch instead of the last which causes several messages to be unacknowledged.
 Now, a JMS trigger that performs batch processing acknowledges the last message in the batch.

- PIE-33552 (IS_9.6_Core_Fix3, IS_9.7_Core_Fix2)
 The pub.client:http service fails with an access denied HTTP response if the url input parameter contains special characters.
 If the encoded value of the url input parameter of the pub.client:http service contains hexadecimal characters that are greater than 7f, the authorization information for the resource specified in the url parameter is not submitted by the http service. As a result, the pub.client:http service fails with the following error:
 (ISC.0064.9314) Authorization Required: (ISS.0084.9004) Access Denied
 This issue is resolved.
- PIE-33498 (IS_9.6_Core_Fix4, IS_9.7_Core_Fix2) Integration Server generates an invalid WSDL document if it encounters identically named elements that are declared to be of different types but belonging to the same target namespace.
 When generating a WSDL document, if Integration Server encounters identically named elements that are declared to be of same type and belonging to the same target namespace, Integration Server issues a warning message stating that the element declaration it creates for the identically named element first is replaced with the element declaration that it generates last. Integration Server must do this only in case of identically named elements that are declared to be of different types but belonging to the same target namespace.
 This issue is resolved. Integration Server now issues a warning message only if the identically named elements are declared to be of different types but belonging to the same target namespace.
- PIE-33539 (IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
 Wrong message type identifier assigned to some messages of "0088 SOAP" server log facility. Integration Server assigns I (Info) as the message type identifier, instead of D (Debug), to some messages of "0088 SOAP" server log facility. As a result, there are discrepancies in the list of log messages displayed in the Logs > Server page of the Integration Server Administrator when Integration Server receives SOAP requests.
 The issue is resolved. The log messages of "0088 SOAP" server log facility now have the correct message type identifier.
- PIE-33764 (IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
 Wrong port shown in Mediator WSDL if multiple ports are configured in Integration Server. If a HTTP or HTTPS port is defined in Integration Server's Security > Ports page and added in the selected ports section in Solutions > Mediator > Administration > General page, the non-primary port with the lowest port number should be shown for the virtual service endpoint deployed in Mediator. However, Integration Server randomly chooses a port and makes the port available to the Mediator WSDL.
 This issue is resolved. The virtual service WSDL in Mediator will now see the non-primary port with the lowest port number if multiple ports are configured for Mediator in Integration Server.
- PIE-33811 (IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)
 Integration Server becomes unresponsive while processing requests from a JMS provider. In certain circumstances, thread cleanup tasks performed during the processing of JMS messages fail to release the lock on an object. Other server threads that need that object go into a wait state, which eventually causes the server to become unresponsive.

This issue is resolved.

- PIE-33886 (IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)

The `pub.flow:getLastError` service does not return any results when the invoke step of a flow service reaches its timeout threshold.

When a parent flow service (service A) invokes a child flow service (service B) with the Timeout parameter specified at the invoke step, and the child flow service invokes another flow service (service C), and timeout is triggered, `pub.flow:getLastError` in parent flow service (service A) returns no error information.

This issue is resolved. When you invoke the `pub.flow:getLastError` service in the parent flow service (service A), the parent flow service returns the `lastError` output parameter as an `errorType` element with a value of `"com.wm.lang.flow.FlowTimeoutException"`.

- PIE-34051 (IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)

Integration Server Administrator displays the number of completed requests over the lifetime of the Integration Server in the Current column on the Statistics screen.

Integration Server Administrator displays the number of completed requests in two columns in the Requests area of the Statistics screen: Current and Lifetime. The Current column should display the number of requests in the last polling period, and the Lifetime column should display the number of completed requests over the lifetime of the Integration Server. Instead, the Current column displays the number of requests over the lifetime of the Integration Server, and the Lifetime column does not display any value. In addition, Integration Server Administrator does not properly display the Average Time row.

This issue is resolved.

- PIE-34109 (IS_9.6_Core_Fix4, IS_9.7_Core_Fix2)

Integration Server does not honor the `watt.server.control.triggerInputControl.delays` or `watt.server.control.triggerInputControl.delayIncrementInterval` values.

When the values for `watt.server.control.triggerInputControl.delays` or `watt.server.control.triggerInputControl.delayIncrementInterval` are changed, Integration Server saves but does not use the specified values if an exception occurred when parsing the supplied values. Regardless of which parameter value caused the parsing error, Integration Server uses the default values for both parameters. However, there was no log message or other type of notification to inform the user that the default values would be used instead.

Now, Integration Server logs the following message if it cannot parse the values of the `watt.server.control.triggerInputControl.delays` or `watt.server.control.triggerInputControl.delayIncrementInterval` parameters:

```
ISS.0125.0015W
```

The trigger input control parameters,

`watt.server.control.triggerInputControl.delaysIncrementInterval` and

`watt.server.control.triggerInputControl.delays`, were set to their default values because an exception occurred when parsing the supplied values.

Cause

Integration Server could not set the trigger input control configuration parameter

`watt.server.control.triggerInputControl.delaysIncrementInterval` and/or

`watt.server.control.triggerInputControl.delays` to the new value because an exception occurred

when parsing the new value. Integration Server reset both configuration parameters to the default values.

Action

Use a valid value for the `watt.server.control.triggerInputControl.delaysIncrementInterval` or `watt.server.control.triggerInputControl.delays` parameter.

- **PIE-32407 (IS_9.7_Core_Fix2)**
While using the `pub.schema:validate` service to validate an object, Integration Server considers a string ending with a decimal point to be a valid long data type.
While validating an object using the `pub.schema:validate` service, Integration Server considers a string ending with a decimal point (for example, `<longType>5.</longType>`) to be a valid long data type.
This issue is resolved. Integration Server no longer considers a string ending with a decimal point to be a valid long data type.
- **PIE-33458 (IS_9.7_Core_Fix2)**
Validation of web service request or response fails because of an extraneous `*body` field.
When processing an inbound web service request or response that has an element that contains whitespace only and the associated field in the IS document types is defined as a document with optional children, Integration Server inserts a `*body` field to contain the whitespace. This is incorrect as Integration Server should ignore the whitespace. The presence of the extraneous `*body` field causes validation to fail because the associated IS document type does not contain a `*body` field.
This issue is now resolved.
- **PIE-33583 (IS_9.7_Core_Fix2)**
A thread dump of Integration Server suggests a thread contention issue occurs during the creation of secure outbound connections.
When Integration Server attempts to create multiple secure outbound connections at the same time, only one connection is created and other threads must wait.
This issue is resolved.
- **PIE-33990 (IS_9.7_Core_Fix2)**
An inbound HTTP/S request that includes `access_token` in the query parameters or header fields is rejected and results in an error message in the `security.log` file.
If an inbound HTTP/S request includes `access_token` in the query parameter or header fields, Integration Server performs OAuth authentication, denies the request, and logs the following message in the `security.log` file:
SYSTEM AUTHENTICATION Integration Server rejected the request to access this resource. The access token is either invalid or expired.
To address this issue, Integration Server now includes server parameters to enable or disable OAuth authentication by Integration Server when an inbound HTTP/S request `access_token` in the query parameters or header fields
`watt.server.auth.oauth.accessToken.useHeaderFields`
Specifies whether Integration Server performs OAuth authentication when an inbound HTTP/S request includes an `access_token` in the header fields. Specify `true` to perform OAuth

authentication. Specify false to skip OAuth authentication. The default is true.
For changes to this parameter to take effect, you must restart Integration Server.

watt.server.auth.oauth.accessToken.useQueryParameters

Specifies whether Integration Server performs OAuth authentication when an inbound HTTP/S request includes an access_token in the query parameter. Specify true to perform OAuth authentication. Specify false to skip OAuth authentication. The default is true.

For changes to this parameter to take effect, you must restart Integration Server.

- PIE-34248 (IS_9.7_Core_Fix2)

While using the extended character sets, Integration Server issues a java.io.UnsupportedEncodingException error.

This issue occurs because the wm-converters.jar file, which is required to support extended character sets, is not present in the Integration Server classpath.

To resolve this issue in Integration Server version 9.5 SP1, do the following:

1. Shut down Integration Server.
2. Using a text editor, navigate to and open the following file:
Software AG_directory/profiles/configuration/custom_wrapper.conf
3. Edit the custom_wrapper.conf file by adding the following jar entry:
wrapper.java.classpath.100=Software AG_directory\common\lib\wm-converters.jar
Where <Software AG_Directory> is the installation directory for Software AG products.
4. Save your changes to the custom_wrapper.conf file and then close the file.
5. Start Integration Server.

To resolve this issue in Integration Server versions 9.6 and 9.7, do the following:

1. Shut down Integration Server.
2. Using a text editor, navigate to and open the following file:
Software AG_directory/profiles/IS_instance_name/configuration/custom_wrapper.conf
3. Edit the custom_wrapper.conf file by adding the following jar entry:
wrapper.java.classpath.100=Software AG_directory\common\lib\wm-converters.jar
Where <Software AG_Directory> is the installation directory for Software AG products.
4. Save your changes to the custom_wrapper.conf file and then close the file.
5. Start Integration Server.

- PIE-34323 (IS_9.7_Core_Fix2)

While executing SQL queries, the Integration Server scheduler sometimes leaves the database cursor open, which causes the database cursor to reach its maximum limit.

The issue is now resolved.

- PIE-34330 (IS_9.7_Core_Fix2)

Integration Server returns different results in debug and run modes when an input variable to a MAP step is mapped to an output variable of the same name.

When an input variable to a MAP step is mapped to an output variable of the same name, Integration Server returns different results in debug mode than it does in run mode.

This issue is resolved.

- PIE-34444 (IS_9.7_Core_Fix2)

The pub.client:http service does not consider the Content-Type headers if the data/args and data/table input parameter value is empty.

If the data/args and data/table input parameter value of the pub.client:http service is not null, but empty, Integration Server uses the "application/x-www-form-urlencoded" content type and ignores the specified Content-Type headers.

The issue is resolved, If the data/args and data/table input parameter value of the pub.client:http service is empty, Integration Server now uses the "application/x-www-form-urlencoded" content type only if no Content-Type is specified as the value of input field /headers/Content-Type.
- PIE-34481 (IS_9.7_Core_Fix2)

Integration Server does not use service output templates for JSON requests.

When Integration Server receives a request containing the header "Accept: application/json" and the invoked service has an output template defined, Integration Server does not use the template to create the output.
- PIE-34482 (IS_9.7_Core_Fix2)

Integration Server throws a NullPointerException followed by a SocketTimeoutException.

If Integration Server sends an outbound HTTP request and the request times out before a response is received, Integration Server may throw the following superfluous NullPointerException before throwing a SocketTimeoutException:

```
java.lang.NullPointerException: null
at com.wm.app.b2b.server.ServerThread.removeListener(ServerThread.java:272)
at com.wm.util.lifecycle.LifecycleManager.unregisterLifecycleListener(LifecycleManager.java:103)
at com.wm.app.b2b.server.HTTPMessageHandler.process(HTTPMessageHandler.java:266)
at com.wm.app.b2b.server.HTTPDispatch.handleRequest(HTTPDispatch.java:173)
```

This issue is now resolved.
- PIE-34503 (IS_9.7_Core_Fix2)

The pub.string:bytesToString does not filter the byte order mark (BOM) characters while converting a byte array to string.

This issue is now resolved. This fix introduces a new optional input parameter, ignoreBOMChars, to the pub.string:bytesToString service.

If ignoreBOMChars is set to true and the value of the encoding input parameter is set to UTF-8, UTF-16, or UTF-32, Integration Server will remove the byte order mark (BOM) characters before converting the input byte array to string, if the byte array contains BOM characters.

If ignoreBOMChars is set to false and the value of the encoding input parameter is set to UTF-8, UTF-16, or UTF-32, Integration Server will include the byte order mark (BOM) characters while converting the input byte array to string, if the byte array contains BOM characters. The default is false.
- PIE-34661 (IS_9.7_Core_Fix2)

Integration Server connects to an SFTP server directly while retrieving the public key of an SFTP server, even if the connection between Integration Server and the SFTP server is through a proxy. While defining an SFTP server alias, upon clicking the Get Host Key button to retrieve the public

key of the SFTP server, Integration Server connects to the SFTP server directly even if the connection between Integration Server and the SFTP server is through a proxy.

The issue is now resolved. If the connection between Integration Server and an SFTP server is through a proxy, Integration Server connects to the SFTP server through the proxy while retrieving the public key of the SFTP server.

- PIE-34662 (IS_9.7_Core_Fix2)

When creating an IS schema from an XML document that does not reference a DTD, Integration Server throws a `NullPointerException`.

An IS schema can be created from an XML document only if the XML document references an existing DTD or contains a DTD inline. However, when creating an IS schema from an XML document that does not reference or contain an existing DTD, Integration Server throws a `NullPointerException` instead of a `ServiceError`.

This issue is now resolved.

- PIE-31363

Integration Server Administrator displays facility code 0090 as "Unknown".

On the View Server Logger Details Screen, Integration Server Administrator displays the facility name for code 0090 as "Unknown". The facility name for 0090 should be "pub Flow Services".

This issue is resolved.

- PIE-33148

When creating a WSDL first provider web service descriptor, the Addressing action property in the WSDL does not show up correctly in the web service descriptor.

Upon creating a WSDL first provider web service descriptor from a WSDL document containing the Addressing action property in the `prefix:namespace` format, Integration Server does not include the prefix in the Addressing action property of the newly created WSDL first provider web service descriptor.

The issue is now resolved. The same Addressing action property that is in the WSDL is included in the web service descriptor created from the WSDL.

- PIE-34287

Integration Server does not recycle the `stats.log` file at the correct time.

By default, Integration Server should recycle the `stats.log` file daily at midnight. However, the first time to recycle the `stats.log` file was incorrect. This caused the file to be recycled daily but at the wrong time.

This issue is now resolved. Integration Server now recycles the `stats.log` file daily at midnight.

- PIE-34509

After setting the server configuration property `watt.core.xml.expandGeneralEntities` to `false`, Integration Server does not encode the apostrophe properly.

This issue is now resolved.

- PIE-34627

Integration Server displays an error message when editing a JNDI provider alias if Integration Server Administrator is accessed via Mozilla Firefox or Google Chrome.

Integration Server displays the following error message when attempting to edit any field of a JNDI

provider alias using Integration Server Administrator: JNDI Alias Name must be specified. This issue occurs only if Integration Server Administrator is accessed via Mozilla Firefox or Google Chrome.

The issue is now resolved.

- PIE-34669
When Integration Server is installed as a service, the service is not unregistered when it is uninstalled.
While uninstalling Integration Server that is installed as a service on a Windows operating system, Integration Server does not uninstall the registered service.
This issue is resolved.
- PIE-34898
A scheduled task configured to run on any one of the Integration Servers connected to the same database does not run as expected and fails with an error.
When a scheduled task is configured to run on any of the Integration Servers that are part of a non-clustered group of Integration Servers in which the ISInternal functional alias on each server points to the same database, the scheduled task fails with the following error:
Could not complete last run
This issue is resolved.
- PIE-34899
The pub.client.ftp:get service does not consider files of 0 KB size to be large files even if the largefilethreshold input parameter is set to 0.
If the largefilethreshold input parameter of the pub.client.ftp:get service is set to 0, Integration Server must consider all files to be large files. The output parameter islargefile must be true and the file content must be returned in the output parameter contentstream as a java.io.InputStream object. However, the pub.client.ftp:get service does not behave as expected. The pub.client.ftp:get service does not consider files of 0 KB size to be large files even if the largefilethreshold input parameter is set to 0. The service returns the islargefile parameter as false and the file content is returned in the output parameter content.
This issue is resolved. The pub.client.ftp:get service now considers files of 0 KB size to be large files if the largefilethreshold input parameter is set to 0.
- PIE-35009
Integration Server appears to become unresponsive because of blocked client requests.
An internal component in Integration Server fails to release a lock when it should. Client threads wait indefinitely to acquire the lock, eventually causing requests to block. As result, Integration Server appears to be unresponsive.
This issue has been resolved.
- PIE-35012
At start up, Integration Server resets the persistence strategy of a cache to none.
When Integration Server starts, it resets the default persistence strategy for a cache to “none”.
Integration Server should change the default persistence strategy to “none” only when TerracottaConfiguration is set to “clustered”.

This issue is resolved.

- **PIE-35059**

Integration Server does not display an appropriate error message when an FTP client makes an active connection to the Integration Server FTP port, if there is a port in Integration Server with port number that is one less than the FTP port.

If the FTP port configured in Integration Server is running in active mode and if there is a port in Integration Server with port number that is one less than the FTP port, when the FTP client makes an active connection to the FTP port configured in Integration Server, the connection fails with a `NullPointerException` message.

The issue is resolved. Integration Server now issues the following error message to the FTP client: `Could not create data port <port number>. Try passive mode."`
- **PIE-35143**

After applying `IS_8.2_SP2_Core_Fix13` or higher, deserialization of SOAP messages to `IData` does not work as expected.

After applying `IS_8.2_SP2_Core_Fix13` or higher, while deserializing SOAP messages to `IData`, Integration Server erroneously adds a default namespace prefix of `"xmlns"` to those fields that are contained in an implicit namespace declaration. This results in broken endpoint service field mappings. This issue occurs only when the `Pre-8.2 compatibility mode property` of the web service descriptor is set to `true`.

This issue is now resolved.
- **PIE-35151**

Integration Server logs an `SQLException` during startup.

When Integration Server starts, it uses the schema name to:

 - Access the audit logging database to determine the length of the `WMSERVICECUSTOMFLDS.STRINGVALUE` and `WMSERVICEACTIVITYLOG.FULLMESSAGE` columns. Integration Server uses the lengths to automatically adjust the size of the values inserted into these columns.
 - Retrieve the database metadata.

By default, Integration Server uses the User ID specified in the `ISCoreAudit` functional alias as the schema name for the audit logging database. You can override this value by specifying a value for the `watt.server.audit.schemaName` server configuration parameter. However, if the User ID does not match the schema name specified in `watt.server.audit.schemaName`, Integration Server logs a `SQLException` to the server log.

This issue is resolved. Integration Server no longer uses the User ID of the `ISCoreAudit` functional alias as the default schema name. Now, Integration Server requires that you specify a value for the `watt.server.audit.schemaName` parameter to retrieve the database metadata and resize the audit logging values in the `WMSERVICECUSTOMFLDS.STRINGVALUE` and `WMSERVICEACTIVITYLOG.FULLMESSAGE` columns. If `watt.server.audit.schemaName` is not set, Integration Server does not retrieve the metadata and assumes the lengths of the `WMSERVICECUSTOMFLDS.STRINGVALUE` and `WMSERVICEACTIVITYLOG.FULLMESSAGE` columns are 512 and 1024, respectively.

Note: Some databases are case-sensitive. When specifying the value for `watt.server.audit.schemaName`, you should match the case of the schema name with the schema

name required by the database.

- **PIE-32619**
After fix IS_8.2_SP2_Core_Fix12 is installed, the SOAP response for a migrated web service contains a document instead of a document list.
Before the fix is installed, data elements are formatted in the SOAP response in a document list, as expected. After the fix is installed, these data elements are formatted as separate documents. This change in output structure makes it difficult to map the documents in the flow.
This issue is resolved. The SOAP response is formatted in the same way it was before fix IS_8.2_SP2_Core_Fix12 was installed.
- **PIE-32737**
Deployer fails when deploying a package that contains a new trigger that does not exist in the earlier version of the package on the target Integration Server.
When deploying a package that contains a trigger that does not exist in the same package on the target Integration Server, Deployer fails because the target Integration Server attempts to suspend the trigger that does not exist. Deployment might fail with one of the following messages:
(ISS.0098.9067) Trigger not found for triggerName: "triggerName"
(ISS.0098.9074) Unable to suspend document retrieval for Trigger "triggerName"
Now, when a package containing triggers is deployed to a target Integration Server, the target Integration Server suspends a trigger only after first verifying that the trigger exists on the target Integration Server.
- **PIE-32873**
It takes a long time for Integration Server to stop all JMS triggers once the triggers are disabled. All the JMS triggers on Integration Server can be disabled at the same time when Integration Server is shut down or when Integration Server Administrator is used to disable all JMS triggers. Once disabled, all of the JMS triggers stop receiving messages immediately. However, Integration Server closes the JMS Session and MessageConsumer objects sequentially. Some JMS providers, including WebSphere MQ, may take a second or two to close the objects for each JMS trigger. When many JMS triggers are disabled at once, closing the JMS Session and MessageConsumer objects can take a significant amount of time.
Now, Integration Server stops closes JMS Session and MessageConsumer objects simultaneously instead of sequentially.
- **PIE-33279**
Option to control how Integration Server decodes duplicates of nested elements in an element of type anyType in a SOAP request.
If the input signature of an IS service exposed as a web service contains a document that has a child variable of type object, the object is represented as an element of type anyType. When decoding a SOAP request for the service, Integration Server places duplicate elements in line in the resulting IData. However, some users want Integration Server to create an array for the duplicate elements in the resulting IData.
To accommodate this, Integration Server now includes a server configuration parameter that you can use to specify how Integration Server decodes duplicate elements nested in an anyType element.

watt.server.xml.xmlNodeToDocument.makeArrayforWS

Specifies how Integration Server decodes duplicate elements contained in an anyType Element.

- Set watt.server.xml.xmlNodeToDocument.makeArrayforWS to true if you want Integration Server to create an array for duplicate elements contained in an element of type anyType.

- Set this parameter to false if you want Integration Server to leave duplicate elements as separate, repeated elements in the element defined to be of type anyType. When set to false, Integration Server does not create an array for elements that appear more than once in the element defined to be of type anyType.

The default is false.

- PIE-33318
Integration Server displays the wrong message in the server log.
Integration Server logs the following message to the server log:
(ISC.0088.9443W) Message not found for messageKey 88.9443
This issue is resolved.
- PIE-33326
After IS_8.2_SP2_Core_Fix14 is installed, Integration Server issues a NullPointerException while loading some custom services.
This issue is resolved.
- PIE-33553
Logging off the Integration Server Administrator fails with an error.
Attempting to log off the Integration Server Administrator fails with the following error message:
Logoff Failed.
This issue is resolved. Upon clicking Log Off, Integration Server terminates the session and displays a message confirming that the session is terminated.
- PIE-33883
Integration Server does not include the initial SOAP request message in the SOAP fault that is generated when an error occurs upon invoking a web service.
Upon invoking a web service, in case of an error, the SOAP fault that Integration Server generates does not include the initial SOAP request even if the pub.soap.handler:getInitialSOAPMessage service is used to retrieve the initial SOAP request message in the outbound callback service.
This issue is resolved. Integration Server now includes the initial SOAP request message in the SOAP fault that is generated when an error occurs upon invoking a web service.
- PIE-33895
Enhancements to Integration Server to provide NTLM (Windows NT LAN Manager) authentication support to allow clients to access resources in web servers that support NTLM authentication. Prior to this fix, Integration Server could use Integrated Windows Authentication as a means of authenticating its identity while establishing connections between Integration Server and web servers on an intranet only if the Integration Server was running on a Windows platform. If Integration Server is running as an NT service, it uses the local system rights for authentication when responding to an Integrated Windows Authentication request. If you log on as a user, Integration Server uses the credentials associated with that session when responding to an

Integrated Windows Authentication request.

After installing this fix, when Integration Server is acting as a client to access resources in web servers, the authentication credentials must be provided by the user explicitly. You can specify NTLM as the authentication type while configuring the transport properties for web service endpoints to authenticate clients who are already logged into a domain using their existing credentials.

In addition, you can specify NTLM as the value for the `auth\` type parameter of `pub.client:http` or `pub.client:soapClient` services and web service connector. If you specify NTLM as the authentication type, you must specify the user name that the service will submit when requesting a protected resource in the following format: `domain_name/user_name`.

- **PIE-34213**
Enabling a Salesforce.com Adapter connection results in a `javax.net.ssl.SSLHandshakeException`. If the Salesforce.com Adapter is installed on an Integration Server for which the WmCloud package is enabled, enabling a Salesforce.com Adapter connection results in the following exception:
`javax.net.ssl.SSLHandshakeException: sun.security.validator.ValidatorException: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target`
This issue is now resolved.
- **PIE-34311**
The Polling Notifications page in localized versions of Integration Server Administrator displays an Adapter Runtime error if there are no scheduled tasks for polling notifications. When there are no scheduled tasks associated with adapter polling notifications, localized versions of Integration Server return a `SQLException` that cannot be verified by the Adapter Runtime causing the following message to be displayed in the Polling Notifications page:
(ART.116.3020) Adapter Runtime (Notification): Unable to get list of notification types supported by node JDBCAdapter. `java.sql.SQLException: Invalid task ID`
Now, when there are no scheduled tasks for polling notifications, a localized Integration Server returns a `SQLException` with a predefined error code that can be verified by the Adapter Runtime.
- **PIE-34373**
Integration Server throws a `java.lang.NoClassDefFoundError` when a user defined in the central user management attempts to log in. A jar file required for central user management, specifically `jsf-api.jar`, is not present in the Integration Server classpath. To resolve this issue in Integration Server versions prior to 9.5 SP1 do the following:
 1. Shut down Integration Server.
 2. Using a text editor, navigate to and open the following file:
`IntegrationServer_InstallDirectory/bin/ini.cnf`
 3. Edit the `ini.cnf` file by adding the following jar entry to the `application.classpath` parameter:
`<Software AG_Directory>/common/lib/ext/jsf-api.jar`
Where `<Software AG_Directory>` is the installation directory for Software AG products.
Note: Make sure to use the appropriate directory separator for the operating system.
 4. Save your changes to the `ini.cnf` file and then close the file.
 5. Start Integration Server.

Note: For versions of Integration Server 9.5 SP1 and later, the Software AG Update Manager will handle updates to the ini.cnf.

- PIE-35015
Updating webMethods Cloud account settings from Integration Server Administrator fails. Using Integration Server Administrator to update webMethods Cloud account settings results in the following error when changes to the account are saved:
The following error occurred while saving account settings :
com.wm.app.b2b.server.ServiceException:
iaik.security.ssl.SSLException: Server certificate rejected by ChainVerifier. See the error log for the full stack trace.
This issue is now resolved.
- PIEAR-625 (WAR_9.5_SP1_Fix5)
Exceptions thrown by adapter services are logged multiple times in the error log. The error log contains multiple entries of an exception thrown by an adapter service because the exception is logged by Adapter Runtime, Adapter Development Kit, and Integration Server.
This issue is resolved. The following two server configuration parameters are introduced to prevent error logging by Adapter Runtime and Adapter Development Kit for exceptions in adapter services:
 - watt.art.adapterService.disable.errorlogging
 - watt.adk.adapterService.disable.errorloggingThe default value of each parameter is false.
Add the new server configuration parameters to Extended Settings in Integration Server Administrator and set the parameters to true.

6.0 Documentation Changes

This section describes significant changes to the documentation, such as the addition, relocation, or removal of product guides, online help, chapters, or other major content. A release is listed in this section only if changes occurred in that release.

Release 10.2

- The *Developing Microservices with webMethods Microservices Container* guide has been renamed *Developing Microservices with webMethods Microservices Runtime* to reflect the product name change from webMethods Microservices Container to webMethods Microservices Runtime.

Release 10.1

- The *Web Applications Developer's Guide* is no longer available. This guide explained how to use the WmTomcat package to incorporate web applications into the Integration Server environment. As of Integration Server 10.1, the WmTomcat package has been removed from the product.

Release 9.8

- Software AG documentation is no longer available on the Software AG installer. You can access all Software AG documentation on the [Documentation website](#).

7.0 Terminology Changes

A release is listed in this section only if changes occurred in that release.

Release 10.2

Old Term	New Term
webMethods Microservices Container	webMethods Microservices Runtime

8.0 Added, Removed, Deprecated, or Changed Items

This section lists functionality, controls, portlets, properties, or other items that have been added, removed, deprecated, or changed. A release is listed in this section only if changes occurred in that release.

Release 10.2

Added Item	Description
Enhanced messaging logging	Integration Server writes detailed log entries when sending or receiving and processing messages. The detailed logging makes it possible to track an individual message across your Integration Servers from the time the message is published through the time a trigger receives, processes, and acknowledges the message. Enhanced logging includes the introduction of the messaging audit log.
WebSocket support	Integration Server enables development of applications using the WebSocket protocol. The WebSocket protocol provides simultaneous two-way communication between a client endpoint and server endpoint over a single TCP connection.

Added Item	Description
Whitelist filtering	To prevent Integration Server from deserializing untrusted Java objects, Integration Server now performs whitelist filtering and includes a whitelist of classes that can be loaded and deserialized in Integration Server.
Support for default scope for Integration Server acting as the OAuth resource server	When configuring an external authorization server, you can specify a default scope in the Default Scope field.
SSL field on the About page of Integration Server Administrator	<p>To help troubleshoot connectivity issues, the About page of Integration Server Administrator, the Software section now contains an SSL field that displays one of the following:</p> <ul style="list-style-type: none"> - The JCE Unlimited Strength Jurisdiction Policy File was not found. Please install it. - The JCE Unlimited Strength Jurisdiction Policy File was found
Support for security response headers	<p>To enhance the security of response headers, Integration Server now supports the following HTTP response security headers:</p> <ul style="list-style-type: none"> - X-Content-Type-Options - X-XSS-Protection - Strict-Transport-Security
Improved support of REST multipart messages	Integration Server provides Java API that enables you to register your own services along with the associated coders for multipart content handler.
Support for strong password policies	<p>To enhance the security, Integration Server now supports stronger password policies for user accounts.</p> <p>To achieve this, Integration Server now covers the following aspects of password security:</p> <ul style="list-style-type: none"> - Stronger password requirements - Password expiration requirements - Account locking
Circuit Breaker	The circuit breaker feature provides the ability to configure a circuit breaker for any service residing on a Microservices Runtime or an appropriately licensed Integration Server.

Added Item	Description
Any field type support for Digital Event Services	An Object or Object List with a Java wrapper type of UNKNOWN is included in the Digital Event Servers event type definition as an Any or Any[] field. Previously, an Object or Object List field with an UNKNOWN Java wrapper type were not included in the corresponding Digital Event Services type definition. Additionally, IS document types can be used to define the structure and content for an Any field in a document published to Digital Event Services.
Map field type support for Digital Event Services	An empty inline document variable configured to allow unspecified fields in a publishable document type is now represented as Map field in the corresponding Digital Event Services type definition.
Synchronizing document types with Digital Event Services automatically	Document types can be set to automatically synchronize with a corresponding Digital Event Services event type definition in the Digital Event Services repository upon save. Set this property to true for a document type that may be used to define the structure and content for the value of an Any field in a document that is published to Digital Event Services.

Added Item	Description
Migration Utility handles updates for hostname	<p>Prior to release 10.2, if you upgraded your Integration Server by installing the new release on a different machine, you had to update the database components below to reflect the host name of the new machine, as follows:</p> <ul style="list-style-type: none"> - If you had scheduled tasks that executed on Integration Server on specific machines in your old installation, you had to update database component ISInternal, table IS_USER_TASKS, column TARGET. - If you wanted to be able to resubmit services that ran before you upgraded, you had to update the following: <ul style="list-style-type: none"> - Database component ProcessAudit, table WMRULEDIST, column SERVID. - Database components IS CoreAudit, ProcessAudit; tables WMDOCUMENT, WMSERVICE, WMSERVICE4X, MSERVICEACTIVITYLOG, WMSERVICEASSOC, WMSERVICECUSTOMFLDS; column SERVERID. - If you wanted to be able to resubmit processes and process steps that ran before you upgraded, you had to update database component ProcessAudit, table PRA_PROCESS_STEP, SERVERID column. <p>The Integration Server migration utility now updates all of these tables for you automatically.</p>

Removed Item	Replacement, if any
No support for host-based name format.	Integration Server does not support the host-based name format to specify the Service Principal Name Format when adding or editing an HTTP(s) port.

Changed Item	Description
updateLanguagePack command for is_instance script.	Provides an option to specify the package and/or feature list to which to apply the language pack installation.
create command for is_instance script.	Provides an option for specifying a default IP address to which to bind ports on the new Integration Server instance.
Server log entries can be written to the console, a file, or both.	Previously, Integration Server could write server log entries to a file or the console (STDOUT). Now, the -log switch includes the both option to write server log information to the computer screen (STDOUT) and to the destination specified by the watt.debug.logfile parameter.
Docker container for an Integration Server Docker image writes the server log to the console (STDOUT).	When running a Docker image of Integration Server in a Docker container, Integration Server writes the server log to the console as well as to the server.log file. Many container deployment solutions provide the ability to view the console log STDOUT for a container.
Repeating tasks scheduled to execute in the Daylight Savings Time (DST) overlap.	If a task includes an hours mask (e.g., "1:15" for running every date at 0115), then the task will run only once, at the pre-overlap ("first") matching time. However, if a task does not have an hours mask (e.g., "hh:15", meaning run at 15 minutes past each hour), then the task will run at the pre-overlap time (the "first" 1:15), and also at the overlap time (the "second" 1:15). Previously, the absence of an hours mask did not result in the task executing twice.
Diagnostic collector	<p>To help with debugging, the Integration Server diagnostic collector returns information regarding whether or not the JCE Unlimited Strength Jurisdiction Policy File is applied to the JVM in the config\ServerAbout.txt file.</p> <p>Integration Server also adds the following file to the list of security providers returned with the diagnostic data: config\JCESecurityInfo.txt</p>

Changed Item	Description
Digital Event Service type definition	<p>With the introduction of support for Any fields and Map fields for Digital Event Services, the Digital Event Type definition generated for a publishable document type changes in the following ways:</p> <ul style="list-style-type: none"> ▪ An Object or Object List with a Java wrapper type of UNKNOWN is included in the Digital Event Services event type definition as an Any or Any[] field. Previously, an Object or Object List field with an UNKNOWN Java wrapper type were not included in the corresponding Digital Event Services type definition ▪ An empty inline document variable in a publishable document type is now represented as Map field in the corresponding Digital Event Services type definition. Previously, an inline document variable became an empty nested event.

Release 10.1

Added Item	Description
SFTP Host Key Checking field in the SFTP User Alias Advanced Settings screen	Allows Integration Server to verify the host key of SFTP server during connection.
Create JMS administered objects on demand.	Integration Server can create administered objects in the JNDI namespace automatically when the lookup for the object fails. This functionality, which is controlled by the Create Administered Objects On Demand option for a JMS connection alias, is available only when Universal Messaging is the JMS provider and the JNDI provider.
Support for using volumes when running an Integration Server image in a Docker container.	Docker volumes can be used for externalizing the contents of the logs directory for the Integration Server instance and the Integration Server profile as well as the config directory for the Integration Server instance running in the Docker container.
The is_container.sh script, which facilitates interaction with Docker, now provides an option for specifying a base image.	When using the createDockerFile or createLeanDockerfile command to create a Docker image for Integration Server, specify a base image using the argument -Dimage.name.

Added Item	Description
Support for using an Integration Server running in a Docker container as the local development server for the local service development feature in Designer.	When using the createDockerFile or createLeanDockerfile command to create a Docker image for Integration Sever, specify that the Integration Server will be used as the local development server using the argument -Dtarget.configuration=localdev
New flag, useAlternateNameForSegment, added in the WmFlatFile package.	useAlternateNameForSegment: Determines whether the flat file parser uses an alternate name for segments if an alternate name is provided.
Support for RFC 7662, OAuth 2.0 Token Introspection	Access tokens generated by an external authorization server can be used to access resources in Integration Server. Access tokens generated by an Integration Server authorization server can be used to access resources in a resource sever that is not an Integration Server.
Support for RFC 7009, OAuth 2.0 Token Revocation	Access tokens generated by Integration Server can be revoked.
Support for REST V2 resources	<p>Integration Server provides a flexible way of defining resources for REST requests. In the Service Development perspective, a developer can define a REST V2 resource with operations that include the following:</p> <ul style="list-style-type: none"> • The format of the URL that REST clients must follow when sending requests to Integration Server acting as the REST server. • The HTTP methods supported by the resource operation. • The flow service associated with a resource operation.
Support for JWT authentication	Integration Server provides support for JSON Web Token (JWT) authentication. JWT provides a secured means of exchanging claims between two parties.

Added Item	Description
HTTP Interceptor	Integration Server provides a framework for creating and registering an HTTP interceptor. An HTTP interceptor intercepts all received HTTP requests and outbound HTTP responses. Integration Server makes the raw HTTP request and response, including the HTTP header information, accessible to the HTTP interceptor.
Serial JMS triggers supported with Universal Messaging.	When using a connection factory configured for a durable subscriber, Integration Server now supports the use of serial JMS triggers with Universal Messaging.
Additional client side queue (CSQ) logging.	When facility code 134 is set to debug or trace levels, Integration Server provides additional logging when CSQ errors occur.
Integration Server now supports migration when source and target version are same.	The migration utility can now be used for migration where the source and target versions of Integration Server are the same. This can be useful for data center or machine moves

Removed Item	Replacement, if any
WmTomcat package	Existing web applications based on the WmTomcat package can be moved to webMethods Application Platform.

Changed Item	Description
Archived stats.log file name format.	Integration Server now uses the following naming format for the archived stats.log files: stats.yyymmdd_hhmmss.log where the timestamp indicates the time at which Integration Server archived the log file. Previously, the archived stats.log name did not include the time portion which caused a naming conflict that prevented successful log rotation if Integration Server attempted to rotate the stats.log more than once on the same day.
Manage Destinations check box	The Manage Destinations check box for a JMS connection alias is now named Enable Destination Management with Designer.

Changed Item	Description
Share Client Prefix check box	The Share Client Prefix check box for webMethods messaging connection aliases is now named Client Prefix Is Shared. The field name now includes the text “prevents removal of shared messaging provider objects” in parentheses to clarify the purpose of the option.

Release 10.0

Added Item	Description
Allowed HTTP method per service	For each service, you can identify the HTTP methods that can be used to invoke the service.
Default web service provider endpoint alias	For HTTP and HTTPS, you can identify a provider web service endpoint alias as the default for each protocol. If a provider web service descriptor contains a binder that specifies the default endpoint alias or no alias for the Port alias property, Integration Server uses the information in the default alias when constructing the WSDL for the descriptor and during run-time processing.
Follow the master support for JMS connection aliases	When Universal Messaging is the JMS provider, you can configure the JMS connection alias to follow the master for consumer and producer connections.
JDBC pool configuration when Integration Server is offline	In conjunction with Command Central, Integration Server makes it possible for administrators to change the configuration of JDBC pools when Integration Server is offline.
JMS Trigger Groups	A JMS trigger group is a collection of two or more identical JMS triggers. Using the methods in the <code>com.wm.app.b2b.server.jms.consumer.JMSTriggerGroupFacade</code> class, users can quickly create JMS triggers for a trigger group. This can be especially helpful when needing to create several identical triggers that point different Universal Messaging realms, such as when using the new Universal Messaging round robin cluster feature for JMS.

Added Item	Description
PATCH support	<p>Integration Server offers support for HTTP PATCH method, which is used for partial updates. In Service Development, a developer can specify PATCH support when defining REST resources, Integration Server also supports the PATCH methods in the pub.client:http service as well as with CORS (Cross Origin Resource Sharing) requests.</p>
REST Resource URLs	<p>Integration Server provides a flexible way to define resource URLs for REST APIs. The new capability allows a developer to define a URL format for the service that does not adhere to the folder structure within the package. Additionally, you can define multiple REST resources for a single service with the HTTP methods of your choice.</p>
Retrieving and Setting HTTP Headers	<p>Integration Server provides the ability to get and set HTTP request and response headers easily.</p> <p>To get the details of an HTTP request for a service, you can add a document reference variable named \$httpRequest that references the pub.flow:HTTPRequest document type in the input signature of a service. While invoking a service that includes a \$httpRequest document reference variable, Integration Server populates the \$httpRequest variable with information from the from the HTTP request.</p> <p>To get the details of an HTTP request for a service, you can add a document reference variable named \$httpRequest that references the pub.flow:HTTPRequest document type in the input signature of a service. While invoking a service that includes a \$httpRequest document reference variable, Integration Server populates the \$httpRequest variable with information from the from the HTTP request.</p>
Searching for JMS Triggers	<p>You can search for or filter the displayed triggers using the Search Triggers link on the Settings > Messaging > JMS Trigger Management page. You can search by trigger name, JMS connection alias or destination to which the trigger subscribes.</p>

Added Item	Description
Session ID Masking in Logs	Session IDs in the session and server logs can now be secured by masking the ID. When you enable the option for securing Session IDs, Integration Server uses the asterisk (*) character to mask the Session ID strings in the logs which prevents user who access the log entries from viewing the actual Session IDs.

Changed Item	Description
Format of the <soap:address location> element in the WSDL of a provider web service descriptor.	The <soap:address location> element in the WSDL document generated for a service first provider web service descriptor will be in the following format regardless of specifying either a provider web service endpoint alias or a host and port as the endpoint: http://host_name.port/ws/namespace:descriptorname
Follow the master is configurable per Universal Messaging connection alias	For a Universal Messaging connection alias, you can now enable follow the master on a per connection alias basis. Additionally, you can decide whether message producers and/or consumers follow the master realm server using the “Enable Follow the Master for Producers “and “Enable Follow the Master for Consumers” options respectively. Previously, following the master was either enabled or disabled for all connections.

Changed Item	Description
JMS Trigger Management page	<p>The Settings > Messaging > JMS Trigger Management page includes the following changes:</p> <p>Clicking Search Triggers displays a simple search feature.</p> <p>In the JSM trigger list, columns are re-ordered to put the State column right next to the trigger name.</p> <p>The State column values can now be “Enabled”, “Disabled”, or “Suspended”. Previously, the possible State values were “Yes”, “No”, “Suspended”.</p> <p>The Status column is new. Possible values are “Running” or “Not Running” followed up by an explanation, such as “Not Running (trigger disabled)”.</p> <p>The SOAP-JMS triggers list is displayed only if Integration Server contains a loaded package with a SOAP-JMS trigger.</p>
Maximum Entries in Cache field	<p>For a distributed cache, the Maximum Entries in Cache value must be set and must be set to a value greater than 0.</p>
Universal Messaging connection alias Client Authentication	<p>The Client Authentication option named SSL is now named Certificate Based, which is more accurate.</p>

Release 9.12

Added Item	Description
IS_DES_CONNECTION messaging connection alias	<p>A predefined messaging connection alias used to establish a connection with a Universal Messaging server for purposes of sending and receiving events with Digital Event Services.</p>
Kill All Sessions Except Your Session option in the Server > Statistics > Sessions screen	<p>Use the Kill All Sessions Except Your Session option in the Current Sessions Screen to kill all the sessions except the session you are currently running.</p>
is_container scripts to create Integration Server images for use in Docker containers	<p>Use the is_container scripts, created by the Integration Server installation, to build Integration Server images for use in Docker containers. You can access the scripts from the following location: Integration Server_directory/docker.</p>

Added Item	Description
Caches and cache managers can be edited while in quiesce mode	Public and system cache managers and caches can be edited when Integration Server runs in quiesce mode.
Public cache can be used for service results caching.	A local or distributed public cache can now be used for service results caching. Previously, Integration Server stored cached service results in the ServiceResults system cache which is part of the SoftwareAG.IS.Services system cache manager.

Changed Item	Description
FTPS ports that use the HTTPS protocol can now be configured using the “Use JSSE” option.	FTPS ports that support TLS 1.1 or TLS 1.2 protocol can now be configured using the “Use JSSE” option in the Edit FTPS Port Configuration screen of Integration Server Administrator. Previously, this option was not available.
Allowed cipher suites can be specified in a file.	To make it easier to specify a long list of cipher suites for use with inbound and outbound SSL connections, Integration Server now allows specifying a file as the value for the cipher suite server configuration properties watt.net.ssl.client.cipherSuiteList, watt.net.ssl.server.cipherSuiteList, watt.net.jsse.client.enabledCipherSuiteList, watt.net.jsse.server.enabledCipherSuiteList.

Changed Item	Description
<p>Some screens in Integration Server Administrator are disabled if webMethods API Gateway is installed on Integration Server.</p>	<p>The following screens under Security screens are disabled:</p> <ul style="list-style-type: none"> • Enterprise Gateway Rules • Enterprise Gateway Rules > Default Alert Options • Enterprise Gateway Rules > Default Alert Options > Edit • Enterprise Gateway Rules > Denial of Service Options • Enterprise Gateway Rules > Denial of Service Options > Configure Global Denial of Service • Enterprise Gateway Rules > Denial of Service Options > Configure Denial of Service by IP Address • Enterprise Gateway Rules > Mobile Application Protection Options • Enterprise Gateway Rules > Mobile Application Protection Options > Device Types > Edit • Enterprise Gateway Rules > Mobile Application Protection Options > Mobile Applications > Edit • Enterprise Gateway Rules > Rules > Create
	<p>The following screens under Security > Ports > Add Port are disabled:</p> <ul style="list-style-type: none"> • Edit Enterprise Gateway Server Configuration • Edit Internal Server Configuration

Release 9.10

Added Item	Description
<p>Configure the allowed protocols for JSSE on a per port basis</p>	<p>Integration Server now provides the ability to specify the allowed protocols for JSSE on a per port basis.</p>

Added Item	Description
Out of the box support for Content-Type application/xml.	Integration Server content handler named ContentHandler_XML now includes support for the content type application/xml.
Kerberos authentication for outbound service requests.	The pub.client:http service now includes Kerberos-related parameters that you can use to acquire a Kerberos ticket for inclusion in the outbound service request.
Java API Classes for working with XMLData in Java services	A set of classes in com.wm.data for manipulating XMLData in Java services. For a list of classes, see the Added APIs for the Release 9.10 in section 11.0 Added, Removed, Deprecated, or Changed APIs
Universal Messaging connection alias can now be configured to connect to an SSL-enabled port on a Universal Messaging realm server	When configuring a Universal Messaging connection alias, you can specify SSL as the client authentication and provide the truststore and keystore information needed for Integration Server to establish an SSL connection with the Universal Messaging realm server. Previously, to establish a secure connection to the Universal Messaging realm server, you needed to configure the javax.net.ssl properties in the JVM used by Integration Server.
A queue on Universal Messaging can be used as the audit logging queue	Integration Server provides the option of using a Universal Messaging queue instead of the internal queue as the audit logging queue on a per logger basis.
Follow the master for webMethods messaging connections to Universal Messaging server	Integration Server now uses the Universal Messaging client setting “follow the master” which indicates that the client session always connects to the master realm server. The follow the master behavior takes precedence over the behavior indicated through the use of a comma-separated or semicolon-separated list of realm servers in the Realm URL field for a Universal Messaging connection alias.

Removed Item	Replacement, if any
EventBus predefined JMS connection alias	The EventBus alias is no longer available as a predefined JMS connection alias. That is, new Integration Server installations will not include this predefined JMS connection alias. However, if you migrated to Integration Server version 9.10 or later from an earlier version, then you might still have the EventBus alias.
EventBusJndiProvider predefined JNDI provider alias	The EventBusJndiProvider alias is no longer available as a predefined JNDI provider alias. That is, new Integration Server installations will not include this predefined JNDI provider alias. However, if you migrated to Integration Server version 9.10 or later from an earlier version, then you might still have the EventBusJndiProvider alias.

Deprecated Item	Replacement, if any
WmTomcat Package	Existing web applications based on the WmTomcat package can be moved to webMethods Application Platform.

Changed Item	Description
Client queue window size	The Universal Messaging window size for the client queue that corresponds to webMethods messaging trigger is now set to the sum of the Capacity and Max execution threads trigger properties. Previously, Universal Messaging determined the window size.

Changed Item	Description
<p>Filters in a webMethods messaging trigger must be identical if the conditions subscribe to the same publishable document type.</p>	<p>If more than one condition in the webMethods messaging trigger specifies the same publishable document type and the trigger receives messages from Universal Messaging, the provider filters must be identical in each condition but the local filters can be different. Specifically, the contents of the Provider Filter (UM) column must be identical for each condition that subscribes to the publishable document type. The contents of the Filter column can be different.</p> <p>When using <code>pub.trigger:createTrigger</code> to create a trigger, the contents of <code>conditions/messageTypeFilterPairs/providerFilter</code> must be the same for each condition. The contents of <code>conditions/messageTypeFilterPairs/filter</code> can be different for each condition.</p> <p>The filter requirement is unchanged for a webMethods messaging trigger that receives messages from Broker. That is, if more than one condition in the webMethods messaging trigger specifies the same publishable document type and the trigger receives messages from the Broker, the filters in the conditions must be the same.</p>
<p>Integration Server Administrator</p>	<p>The HTML-based utility used to administer the webMethods Integration Server has a new look and feel. Menu and screen locations have not changed.</p>
<p>Internal Server ports that use the HTTPS protocol can now be configured using the “Use JSSE” option.</p>	<p>Internal Server ports that support TLS 1.1 or TLS 1.2 protocol can now be configured using the “Use JSSE” option in the Edit Internal Server Configuration screen of Integration Server Administrator. Previously, this option was not available.</p>
<p>OAuth Errors now returned in the content type specified in the client request</p>	<p>Previously, Integration Server always returned OAuth errors as HTML. Now, Integration Server returns OAuth errors in the content type specified in the client request.</p>
<p>Reply messages for webMethods messaging routed through Universal Messaging.</p>	<p>Integration Server now always encodes reply message as IData even when the encoding type of the reply document type is protocol buffers.</p>

Release 9.9

Added Item	Description
Cluster synchronization for webMethods messaging triggers that receive documents from Universal Messaging.	If a webMethods messaging trigger receives documents from Universal Messaging resides on Integration Servers that is a member of a cluster, changes that you make for document retrieval, document processing, and for trigger properties can be propagated to other servers in the cluster automatically. Previously, cluster synchronization was available for webMethods messaging triggers that received documents from webMethods Broker only.
CDATA block support to outbound SOAP processing	Integration Server now provides CDATA block support for processing of outbound SOAP messages only when Integration Server hosts the web service provider. When a service used as an operation in a web service provider returns String values containing CDATA blocks, when encoding the IData into a SOAP message, Integration Server places the CDATA text in the outbound SOAP message in a CDATA section and does not url-encode special characters in the delimiters or text block. Previously, Integration Server ignored CDATA delimiters in String fields and treated text in a CDATA block as regular text.
Dedicated MessageConsumer for receiving synchronous replies per JMS connection alias	When a JMS connection alias has the Create Temporary Queue check box and the Enable Request-Reply Listener for Temporary Queue check box selected, Integration Server uses a dedicated consumer to retrieve replies to all requests sent using that JMS connection alias.
Default instance name can be defined by the user	The default instance of Integration Server can now be specified during the installation process. Previously, the default instance was always named “default”.
Fail-fast mode for a JDBC functional alias.	A JDBC functional alias can now be configured to enter fail-fast mode when a transient error caused by an unavailable database prevents the connection pool alias used by the JDBC functional alias from establishing a connection to the database.

Added Item	Description
Instances of Integration Server can be started and stopped from the Windows Start menu.	Any instance of Integration Server can be started or stopped from the Windows Start menu. The Integration Server Administrator can also be started from the Windows Start menu for any instance of Integration Server. Previously, you could start, stop, or launch Integration Server Administrator for the default instance only.
IS_LOCAL_CONNECTION messaging connection alias	A predefined messaging connection alias for use with publishable document types that will be published locally only.
New filter added to the Enterprise Gateway rule	Custom filter added to the list of filter that can be included in an Enterprise Gateway rule. You can use this filter to customize and invoke services in the Enterprise Gateway Server.
URL aliases can now have specific destinations per port and for the empty path	A URL alias can now identify a port-specific destination for the alias. Additionally, an alias can now be created for the empty path for a port.
XMLData and XML document types	XMLData is a new data type that more faithfully represents XML content. This enables Integration Server to support XML Schema constructs such a substitution groups, nested model groups, the any wildcard, and the anyAttribute wildcard. XMLData also provides improvements with regards to handling of XML namespaces. XMLData is surfaced in Integration Server and Designer through XML document types which are created from XML Schema definitions.

Removed Item	Replacement, if any
startup.bat/sh and shutdown.bat/sh scripts contained in the following directory: <i>Integration Server_directory\bin</i>	The startup.bat/sh and shutdown.bat/sh scripts contained in and <i>Integration Server_directory\bin</i> directory have been removed. These scripts are replaced by the startup.bat/sh and shutdown.bat/sh scripts respectively in the following directory: <i>Software</i> <i>AG_directory\profiles\IS_instance_name\bin</i>

Removed Item	Replacement, if any
<p>server.bat/sh scripts contained in the following directory: <i>Integration Server_directory</i>\bin</p>	<p>The server.bat/sh scripts contained in the <i>Integration Server_directory</i>\bin directory have been removed. These scripts are replaced by the startup.bat/sh scripts in the following directory: <i>Software</i> <i>AG_directory</i>\profiles\IS_instance_name\bin</p>
Deprecated Item	Replacement, if any
<p>WmVCS and VCS Integration feature</p>	<p>Use the local service development feature (Local Version Control Integration) to check package elements and their supporting files into and out of a version control system (VCS) directly from Designer.</p>
Changed Item	Description
<p>Destination List parameter renamed to Destination Lookup Name List</p>	<p>On the Create JMS Connection Alias, Edit JMS Connection Alias, and JMS Connection Alias details screens available under Settings > Messaging in Integration Server Administrator, the Destination List parameter was renamed to Destination Lookup Name List to clarify that the destination lookup names must be provided.</p>
<p>Default messaging connection alias</p>	<p>Previously, you did not have to select a default messaging connection alias for use with webMethods messaging. You could select [LOCAL ONLY] which would indicated that publishable document types that specified the DEFAULT messaging connection alias could be published locally only. Now, you must select a messaging connection alias as the default. If you want the default to be local publishing, specify IS_LOCAL_CONNECTION as the default messaging connection alias in Integration Server Administrator.</p>
<p>WmTomcat package</p>	<p>The WmTomcat package is still installed by default, but is no longer enabled by default. The WmTomcat must be disabled if you want to use webMethods Application Platform with Integration Server.</p>

Changed Item	Description
Client Principal and Client Principal Password fields renamed to Principal and Principal Password	On the Create, Edit, or View Web Services Consumer Endpoint Alias details screens available under Settings > Web Services > Create Web Service Endpoint Alias in Integration Server Administrator, under the Kerberos Properties section, the Client Principal and Client Principal Password fields were renamed to Principal and Principal Password.
Use Subject Credentials field renamed to Use Subject Credentials Only	On the Security > Kerberos and Security > Kerberos > Edit screens, the Use Subject Credentials field has been renamed to Use Subject Credentials Only.
Serial webMethods messaging triggers that receive documents from Universal Messaging now process documents from a publisher publication order.	Previously, a webMethods messaging trigger with serial processing corresponded to a shared named object on a channel on Universal Messaging server. Now, a serial webMethods messaging trigger corresponds to a priority named object. This provides processing in publishing order for a serial trigger in a cluster or a non-clustered group of Integration Server
Transient error handling during trigger preprocessing.	<p>The following changes were made to improve transient error handling during trigger preprocessing:</p> <p>If the <code>ISRuntimeException</code> occurs during trigger preprocessing, Integration Server checks the values of the <code>watt.server.trigger.preprocess.suspendAndRetryOnError</code> parameter and the <code>On Retry Failure</code> trigger property. If <code>watt.server.trigger.preprocess.suspendAndRetryOnError</code> is set to <code>true</code> or <code>On Retry Failure</code> is set to "Suspend and retry later", then Integration Server suspends the trigger.</p> <p>Previously, Integration Server checked the <code>watt.server.trigger.preprocess.suspendAndRetryOnError</code> server configuration property only for webMethods messaging triggers that received messages from webMethods Broker. Also previously, Integration Server checked the <code>On retry failure</code> property only for JMS triggers and webMethods messaging triggers that received</p>

Changed Item	Description
	<p>messages from Universal Messaging.</p> <p>The <code>watt.server.trigger.preprocess.suspendAndRetryOnError</code> parameter now applies to all trigger types and messaging providers. Previously, the behavior provided by this property applied only to <code>webMethods</code> messaging triggers that received messages from <code>webMethods</code> Broker. Now, this property also works with <code>webMethods</code> messaging triggers that receive messages from Universal Messaging as well as JMS triggers.</p> <p>Integration Server now includes a server configuration property, <code>watt.server.trigger.preprocess.monitorDatabaseOnConnectionException</code>, that determines how Integration Server handles a <code>ConnectionException</code> that causes a transient error. A <code>ConnectionException</code> occurs when the document history database is not enabled or is configured incorrectly.</p> <p>Typically, transient errors that occur during preprocessing occur during exactly-once processing.</p>
<p>Naming convention for port aliases for migrated ports</p>	<p>To ensure that each port alias is unique, Integration Server now includes the host name for the port in the port alias for migrated ports. The new naming convention is: <code>protocolListener_portNumber_hostName_packageName</code></p>
<p>Port number used while generating XIDs</p>	<p>Previously, Integration Server used the default Integration Server primary port number while generating the unique XIDs that are used to persist the state of XA transactions. Now, Integration Server uses the configured primary port number while generating the XIDs.</p>
<p>URL Path field renamed to Default URL Path</p>	<p>On the Settings > URL Alias > Create and Settings > URL Alias page, the URL Path field had been renamed to Default URL Path.</p>

Release 9.8

Added Item	Description
Basic client authentication performed when connecting Integration Server to a Universal Messaging realm server	When Universal Messaging is the webMethods messaging provider, connections between Integration Server and Universal Messaging servers can now be authenticated using user names and passwords. This authentication is specified in the Client Authentication Settings section on the Settings > Messaging > webMethods Messaging Settings > Universal Messaging Connection Alias > Create screen.
Use Secure Socket Layers to connect to a port on a Universal Messaging realm server	When Universal Messaging is the webMethods messaging provider, connections between Integration Server and Universal Messaging servers can now use one-way or two-way SSL once client-side parameters are configured on Integration Server.
Action on Startup Error option for clustering configuration	When configuring a cluster, you can now specify what action Integration Server takes when an error prevents the server from joining a cluster at startup. Previously, Integration Server would shut down if it could not connect to the cluster at start up. Now, you can start Integration Server as a standalone Integration Server, shut down Integration Server, or start the Integration Server in quiesce mode.
Security > Ports > Enterprise Gateway Registration Port Connections screen	The new Enterprise Gateway Registration Port Connections screen lists the connections from Internal Servers to the Enterprise Gateway registration port. To access this screen, go to the Ports Screen and click the port number of an Enterprise Gateway registration port. On the Security > Ports > View Enterprise Gateway Server Details screen, click Display Connections to Enterprise Gateway Registration Port.

Removed Item	Replacement, if any
WmPKI package	The WmPKI package which provides functionality for using PKI profiles in conjunction with Integration Server has been removed from the product. There is no replacement for the WmPKI package.
Changed Item	Description
JMS Trigger Management page	In Integration Server Administrator, the Settings > Messaging > JMS Trigger Management page now expands the Individual Standard JMS Trigger Controls and Individual SOAP JMS Trigger Controls tables by default. Previously, the tables were collapsed. Additionally, the page now includes links that you can use to jump from the top of one table to the other. This improves page navigation when there are a large number of JMS triggers.
Handling of undeclared namespaces in the resulting document (IData), when an endpoint service is invoked.	When an endpoint service is invoked, Integration Server now includes the undeclared namespace definitions in the resulting document (IData).
Realm URL parameter	<p>The Realm URL parameter on the on Settings > Messaging > webMethods Messaging Settings > Universal Messaging Connection Alias > Create and Edit pages now accepts a semicolon-separated list of URLs to the servers in a Universal Messaging cluster. Previously, the field accepted a comma-separated list only. Use a comma-separated list for Integration Server to always attempt to connect to the first Universal Messaging server in the list, trying the second server and so forth only if the first server becomes unavailable. Use a semicolon-separated list for Integration Server to connect to a randomly chosen URL from the list.</p> <p>You can now specify realm URLs that use the protocol nsps or nhps.</p>
Handling of HTTP requests with no authentication scheme specified in the Authorization header	Integration Server treats HTTP requests with no authentication scheme in the Authorization header as BASIC authentication requests and performs authentication for valid credentials. Previously, Integration Server did not force requests to have an

Changed Item	Description
User Names parameter	authentication scheme in the Authorization header. The User Names field on the Security > User Management > Add and Remove Users screen no longer allows specifying a <i>username ;password</i> combination per line. Passwords cannot be supplied in the User Names field.
JSON text with an array at the root	Integration Server now accepts JSON text containing an array at its root. If the supplied JSON text contains an array at its root and that array does not have a name, Integration Server uses a fixed name of "\$rootArray" for the array value.

9.0 Added, Removed, Deprecated, or Changed Built-In Services

A release is listed in this section only if changes occurred in that release.

Release 10.2

Added Service	Description
pub.client:websocket	Establishes a WebSocket connection to the URL captured in the identified WebSocket client endpoint.
pub.oauth:getToken	Requests an access token from the Integration Server acting as the authorization server. This service replaces the pub.oauth:getAccessToken and pub.oauth:refreshAccessToken services. Providing a single OAuth token endpoint for requesting an access token conforms to the OAuth standard, RFC 6749.
pub.oauth:removeExpiredAccessTokens	Removes expired access tokens from the database.
pub.websocket:close	Closes the WebSocket connection.
pub.websocket:disconnect	Closes the WebSocket connection abruptly.
pub.websocket:onBinary	Specification to use as the signature for a callback service related to WebSocket server or client endpoint that handles a binary payload.
pub.websocket:onClose	Specification to use as the signature for a callback service that gets invoked when a WebSocket session is closed gracefully or disconnected.

Added Service	Description
pub.websocket:onConnect	Specification to use as the signature for a callback service that gets invoked when a WebSocket session is successfully established.
pub.websocket:onError	Specification to use as the signature for a callback service that gets invoked when an error is encountered, resulting in WebSocket session termination.
pub.websocket:onText	Specification to use as the signature for a callback service that handles a text payload.
pub.websocket:send	Sends a message on a connected WebSocket session.

Deprecated Service	Replacement, if any
pub.jms.wmjms:receiveStream	None. webMethods Broker is deprecated, resulting in the deprecation of this service.
pub.jms.wmjms:sendStream	None. webMethods Broker is deprecated, resulting in the deprecation of this service.
pub.oauth:getAccessToken	pub.oauth:getToken
pub.oauth:refreshAccessToken	pub.oauth:getToken

Changed Service	Description
pub.client:http handling of a 307 or 308 status code	When the pub.client:http service sends a POST, PUT, DELETE, or PATCH request and receives a 307 or 308 response status code from an HTTP server, the service no longer changes the request method to GET when redirecting.
pub.publish:envelope	The <i>uuid</i> field can now be set in the document envelope when publishing a document to Universal Messaging using the pub.publish:publish service. Previously, the <i>uuid</i> field was read-only.

Release 10.1

Added Service	Description
pub.json:validate	Validates JSON content against a JSON schema.

Added Service	Description
pub.oauth:introspectToken	Checks if an access token or refresh token generated by an Integration Server acting as an authorization server is active.
pub.oauth:revokeToken	Revokes a token issued by an Integration Server acting as an authorization server
pub.xmldata:getGroupObjects	Returns information about the objects and their tags in a group value. This service is useful primarily when the structure of data is not known in advanced and must be discovered. Examples include any wildcard content, anyType content, substitution groups, and XmlData that was created without using an XML document type.
pub.xmldata:getGroupValues	Returns information about the values and their tags in a group value. This service is useful primarily when the structure of data is not known in advanced and must be discovered. Examples include any wildcard content, anyType content, substitution groups, and XmlData that was created without using n XML document type.
pub.xmldata:queryXMLNode	Queries an XML node and returns the results as XMLData.

Removed Service	Replacement, if any
pub.restV2:listAllRESTResources	None. REST V2 resources are visible in the Package Navigator view of Designer.

Changed Service	Description
pub.jms:sendBatch	The pub.jms:sendBatch service can now be used with Universal Messaging as the JMS provider.
pub.report*	Integration Server now executes the template file and any services therein, using the credentials of the user who invoked the pub.report* service.
pub.xml:loadEnhancedXMLNode	<p>The inputProcessing\supportDTD input parameter now has a default value of false.</p> <p>The inputProcessing\isSupportingExternalEntities input parameter now has a default value of false.</p>

Changed Service	Description
pub.xml:XMLStringToEnhancedXMLNode	The inputProcessing\supportDTD input parameter now has a default value of false. The inputProcessing\isSupportingExternalEntities input parameter now has a default value of false

Release 10.0

Added Service	Description
pub.flow:HTTPRequest	Document type that represents information about the HTTP request received by Integration Server.
pub.flow:HTTPResponse	Document type that specifies the HTTP response information to be returned by Integration Server to the client.
pub.restV2:listAllRESTResources	Lists the REST resources configured using the URL template-based approach for a specified Integration Server service or for all the Integration Server services in a specified package. These REST resources are invoked using the restv2 directive.

Removed Service	Replacement, if any
pub.event.nerv:eventToDocument	No replacement service because the pub.event.routing:subscribe service, which replaces the deprecated pub.event.nerv:subscribe service, handles the conversion of the incoming JMS message into an IS document (IData).
pub.event.nerv:send	pub.event.routing:send
pub.event.nerv:subscribe	pub.event.routing:subscribe
pub.event.nerv:unsubscribe	pub.event.routing:unsubscribe

Changed Service	Description
pub.client:ftp	Updated the input parameter <i>secure</i> to include <i>useJSSE</i> as a child parameter.
pub.client.ftp:login	Updated the input parameter <i>secure</i> to include <i>useJSSE</i> as a child parameter.

Changed Service	Description
pub.client:http	You can now supply a value in the data/bytes input variable when the method input variable is DELETE.
pub.security.xml:signXML	<p>Added input parameter isDetached to indicate whether the signature should be detached or not.</p> <p>Added input parameter keyName which is the name that is used by the signer to communicate a key identifier to the recipient.</p> <p>Added input parameter keyNameValue to indicate whether to include the key value (RSAKeyValue or DSAKeyValue), based on the certificate used for signing. The KeyValue element contains a single public key that can be used in validating the signature.</p> <p>The certData input parameter now includes X509_CRL as an option.</p>

Release 9.12

Added Service	Description
pub.cache.serviceResults:addCacheEntry	Adds cached entry into service results for the service without executing the actual service. This service can be useful to perform bulk load of service results at Integration Server startup.
pub.cache.serviceResults:listServiceCache	Returns a list of the cached service results for a particular service.
pub.cache.serviceResults:resetServerCache	Resets the cache for all services in the service results cache, resulting in the removal of all cached service results for all services from the service results cache.
pub.cache.serviceResults:resetServiceCache	Resets the cache for specific service, resulting in the removal of cached service results for the service.

Changed Service	Description
pub.client:http	<p>The default value of the <i>useJSSE</i> parameter is now null, which allows the <i>watt.net.ssl.client.useJSSE</i> server configuration property to determine the default behavior for outbound HTTPS connections.</p> <p>Added input parameter <i>delegation</i> to allow a user to execute a service on behalf of other user.</p> <p>Added input parameter <i>requestDelegatableToken</i> to specify if you want to request for a forwardable ticket granting ticket to send to the intermediary. The intermediary can use this forwardable ticket granting ticket for Kerberos delegation.Integration Server, which the server can use for delegation</p>
pub.client.ldap:search	<p>Added input parameter <i>pageSize</i> to set the number of entries to return in a page. Added input and output parameter <i>ldapCookie</i> which contains the index of the page count for the search.</p>
pub.client.ftp:login	<p>Added input parameter <i>useJSSE</i> to allow use of JSSE for creating outbound FTPS connections.</p>
pub.client:soapClient	<p>Added input parameter <i>useJSSE</i> to allow use of JSSE for outbound web service invocation.</p>
pub.date*	<p>Previously, if a value was not specified for the locale input parameter, Integration Server used the locale from the session used by the client that invoked the service. Now, Integration Server uses the value of the <i>watt.server.session.locale.ignore</i> server configuration parameter to determine whether the locale is obtained from the session used by the client that invoked the service or if the locale is the locale of Integration Server.</p>
pub.flow:setResponse2	<p>Added input variable <i>responseStreamTransferEncoding</i> to use chunked transfer encoding for the response and include the "Transfer-Encoding: chunked" response header.</p>
pub.json:jsonStreamToDocument	<p>Added input parameter <i>decodeRealAsString</i> to converts real numbers in the <i>jsonStream</i> to String.</p>
pub.json:jsonStringToDocument	<p>Added input parameter <i>decodeRealAsString</i> to converts real numbers in the <i>jsonStream</i> to String.</p>

Changed Service

pub.security:userInfoSpec

Description

The error input parameter is now named `userInfoError`.

Release 9.10**Added Service**

pub.date:compareDates

Compares two dates.

pub.date:incrementDate

Increments a date by a specified period.

pub.event.routing:eventAcknowledgement

Defines the input signature for a callback service that processes acknowledgments sent by the Event Routing framework.

pub.security:userInfoSpec

Specification for UserInfo service that performs custom processing based on the personally identifiable information in the OpenID Connect UserInfo token.

pub.string:compareStrings

Performs a case-sensitive comparison of two strings, and indicates whether the strings are identical.

pub.string:isAlphanumeric

Determines whether a string consists entirely of alphanumeric characters (in the ranges A–Z, a–z, or 0–9).

pub.string:isDate

Determines whether a string follows a specified date pattern.

pub.string:isNullOrBlank

Checks a string for a null or a blank value.

pub.string:isNumber

Determines whether the contents of a string can be converted to a float value.

pub.string:substitutePipelineVariables

Replaces a pipeline variable with its corresponding value.

Changed Service	Description
pub.client:http	<p>The <i>method</i> input parameter now supports the patch method.</p> <p>The pub.client:http service now includes fields for supplying Kerberos information that Integration Server uses to acquire a Kerberos ticket to include in the outbound service request. Specifically, the <i>auth</i> input parameter now contains a <i>kerberos</i> document in which you can specify <i>jaasContext</i>, <i>clientPrincipal</i>, <i>clientPassword</i>, <i>servicePrincipal</i>, and <i>servicePrincipalForm</i>.</p> <p>If the remote server redirected pub.client:http to a different location, <i>encodedURL</i> contain the URL that pub.client:http submitted to the server to which it was redirected.</p>
pub.event.routing:send	New input parameters <i>isAsync</i> , <i>service Name</i> , and <i>runAsUser</i> .
pub.jms:send	Added the <i>JMSMessage/header/replyTo</i> input parameter which can be used to specify a destination for replies without actually needing to wait for replies.
pub.security.keystore.pkcs7:sign	New input parameters <i>dataAsStream</i> and <i>signatureAsStream</i> .
pub.security.pkcs7:verify	New input parameters <i>signatureAsStream</i> and <i>dataAsStream</i> .
	New output parameter <i>contentAsStream</i> .
pub.security.util:createMessageDigest	New input parameter <i>inputAsStream</i> .
	New output parameter <i>outputAsStream</i>

Release 9.9

Added Service	Description
pub.assets:getChecksums	Retrieves the checksums for assets, specifically namespace elements, on Integration Server.
pub.event.routing:send	Sends events to the messaging provider using Software AG Event Routing.
pub.event.routing:subscribe	Creates a subscription to a particular type of event using Software AG Event Routing.

Added Service	Description
pub.event.routing:unsubscribe	Removes a subscription to an event type previously subscribed to using the pub.event.routing:subscribe service.
pub.security.enterpriseGateway:customFilterSpec	Specification for services that can be invoked by the custom filter in Enterprise Gateway rules.
pub.xmldata:domNodeToXMLData	Converts a DOM node to a document (XMLData object).
pub.xmldata:getAttributes	Retrieves the attribute values from an XMLData object.
pub.xmldata:getInstanceTag	Retrieves the instance (xsi) tag values from an XMLData object.
pub.xmldata:getNamespaceTag	Retrieves the namespace tags values from an XMLData object.
pub.xmldata:setAttribute	Sets the value of an attribute in an XMLData object.
pub.xmldata:setInstanceTag	Sets an instance tag value in an XMLData object.
pub.xmldata.setNamespaceTag	Sets a namespace tag value in an XMLData object.
pub.xmldata:xmlDataToXMLString	Converts a document (XMLData object) to an XML String.
Deprecated Service	Replacement, if any
pub.event.nerv:eventToDocument	No replacement service because the pub.event.routing:subscribe service, which replaces the deprecated pub.event.nerv:subscribe service, handles the conversion of the incoming JMS message into an IS document (IData) now.
pub.event.nerv:send	pub.event.routing:send
pub.event.nerv:subscribe	pub.event.routing:subscribe
pub.event.nerv:unsubscribe	pub.event.routing:unsubscribe
Changed Service	Description
pub.client.ftp:getCompletedNotification	The publishable document type now specifies the IS_LOCAL_CONNECTION alias as the connection alias used to publish and retrieve instances of this document type.

Changed Service	Description
pub.client.ftp:putCompletedNotification	The publishable document type now specifies the IS_LOCAL_CONNECTION alias as the connection alias used to publish and retrieve instances of this document type.
pub.schema:validate	The service can now validate XMLData against an XML document type.
pub.remote:invoke	Integration Server now ignores the value of the \$close input parameter when a trigger service for a webMethods messaging trigger executes the pub.remote:invoke service, if the watt.server.trigger.reuseSession server configuration parameter is set to true.
pub.trigger:resumeProcessing	The <i>applyChangeAcrossCluster</i> input parameter now applies to webMethods messaging triggers that receive documents from Universal Messaging. Previously, the <i>applyChangeAcrossCluster</i> input parameter applied only to triggers that received documents from Broker.
pub.trigger:resumeRetrieval	The <i>applyChangeAcrossCluster</i> input parameter now applies to webMethods messaging triggers that receive documents from Universal Messaging. Previously, the <i>applyChangeAcrossCluster</i> input parameter applied only to triggers that received documents from Broker.
pub.trigger:suspendProcessing	The <i>applyChangeAcrossCluster</i> input parameter now applies to webMethods messaging triggers that receive documents from Universal Messaging. Previously, the <i>applyChangeAcrossCluster</i> input parameter applied only to triggers that received documents from Broker.
pub.trigger:suspendRetrieval	The <i>applyChangeAcrossCluster</i> input parameter now applies to webMethods messaging triggers that receive documents from Universal Messaging. Previously, the <i>applyChangeAcrossCluster</i> input parameter applied only to triggers that received documents from Broker.

Release 9.8

Added Service	Description
pub.cache:getAll	Retrieves the values of the cached elements for the specified keys.
pub.cache:putAll	Populates a collection of elements in the cache with the specified key-value pairs.
pub.cache:removeAll	Removes the cached elements associated with a list of keys or, if no keys are specified, removes all elements from the cache.
pub.flow:getCallingService	Retrieves the service name and package name of the parent of the calling child service.
pub.event.nerv:subscribe	Subscribes to a particular type of event on the Network for Event Routing and Variation (NERV) and identifies the service that will act as the event handler.
pub.event.nerv:unsubscribe	Unsubscribes from an event type previously subscribed to using the pub.event.nerv:subscribe service.
pub.utils.messaging:migrateDocTypesTriggersToUM	Migrates the messaging provider for publishable document types and webMethods messaging triggers from webMethods Broker to webMethods Universal Messaging.
pub.utils:transcode	Transcodes data from one encoding to another.

Removed Service	Replacement, if any
pub.pki.pkcs7:sign	
pub.pki.pkcs7:verify	
pub.pki.smime:createSignedAndEncryptedData	
pub.pki.smime:createSignedData	
pub.pki.smime:processEncryptedData	
pub.pki.smime:processSignedData	

Changed Service	Description
pub.cache:remove	Added <i>useWriter</i> input parameter.

Changed Service	Description
pub.flow:savePipelineToFile	Before writing the pipeline to a file, Integration Server verifies whether the provided path is specified in the allowedWritePaths parameter of the file access control configuration file (fileAccessControl.cnf). If it is not, Integration Server does not write the pipeline to a file.
pub.publish:deliver	Added the ability to deliver documents with webMethods Universal Messaging.
pub.publish:deliverAndWait	Added the ability to publish documents with webMethods Universal Messaging.
pub.publish:envelope	webMethods Universal Messaging now supports the read-only parameter <i>destID</i> .
pub.string:bytesToString	Added <i>ignoreBOMChars</i> input parameter.

10.0 Added, Removed, Deprecated, or Changed Parameters

A release is listed in this section only if changes occurred in that release.

Release 10.2

Added Parameter	Description
watt.adapters.withOnlineHelp	Specifies a comma-separated list of the adapter names that have online help content that needs to be accessible through the Help link on Integration Server Administrator.
watt.core.xsd.useKnownSchemaLocation	Specifies whether Integration Server includes a “known schema location” and the schemaLocation attribute for an import statement in the WSDL document generated for a provider web service descriptor. When the schemaLocation attribute is not present in the WSDL document, an Integration Server that consumes the WSDL document will not dereference the schemaLocation attribute value during web service processing. If the known schema locations are the only import statements that required namespace resolution, the absence of the schemaLocation attribute allows web service development to occur while not connected to the Internet.

Added Parameter	Description
watt.server.checkWhitelist	Specifies whether Integration Server uses a whitelist to filter the list of classes than can be used for deserialization.
watt.server.circuitBreaker.threadPoolMax	<p>Specifies the maximum number of threads that the server maintains in the circuit breaker thread pool.</p> <p>Note: The circuit breaker feature is available by default for a service that resides in a Microservices Runtime. To use the circuit breaker feature with Integration Server, your Integration Server must have additional licensing.</p>
watt.server.circuitBreaker.threadPoolMin	<p>Specifies the minimum number of threads that the server maintains in the circuit breaker thread pool.</p> <p>Note: The circuit breaker feature is available by default for a service that resides in a Microservices Runtime. To use the circuit breaker feature with Integration Server, your Integration Server must have additional licensing.</p>
watt.server.dumpWhitelist	When whitelist class filtering is enabled, indicates whether or not Integration Server generates a log file that lists any classes for which deserialization was attempted but are not part of a whiteslistclasses.xml file.
watt.server.http.Strict-Transport-Security	Specifies whether Integration Server includes Strict-Transport-Security header in response headers
watt.server.http.X-Content-Type-Options	Specifies whether Integration Server includes the X-Content-Type-Options header in response headers.
watt.server.http.X-XSS-Protection	Specifies whether Integration Server includes X-XSS-Protection in the response headers.
watt.server.jms.csq.publishDelayWhileDraining	Specifies the number of seconds that Integration Server waits before publishing a JMS message to the client side queue (CSQ) while the CSQ drains.
watt.server.messaging.csq.publishDelayWhileDraining	Specifies the number of seconds that Integration Server waits before publishing a message to the client side queue (CSQ) while the CSQ drains.

Added Parameter	Description
watt.server.oauth.disableClient.disableTokens	Specifies whether a client access token issued to a client account that is disabled can be used to access resources
watt.server.password.historyLength	Specifies the maximum number of previously set passwords that Integration Server saves in the memory for a user.
watt.server.password.maxIdenticalCharsInARow	Specifies the maximum number of identical characters in a row a password can contain for non-Administrator users.
watt.server.password.maxLength	Specifies the maximum number of characters (alphabetic characters, digits, and special characters combined) the password must contain for non-Administrator users.
watt.server.serverlogMaskBearerToken	Specifies whether Integration Server masks the value of a bearer token when writing to the server log.
watt.server.SOAP.preserveCDATA	Specifies whether Integration Server encodes CDATA blocks in outbound messages.
watt.ssh.jsch.mac_c2s	Specifies the order of message authentication code (MAC) algorithms for client to server transmission.
watt.ssh.jsch.mac_s2c	Specifies the order of message authentication code (MAC) algorithms for server to client transmission.

Changed Parameter	Description
watt.server.password.minDigits	The default value of this parameter is now 0. Integration Server uses the new default value for new Integration Server instances only.
watt.server.password.minLowerChars	The default value of this parameter is now 0. Integration Server uses the new default value for new Integration Server instances only.
watt.server.password.minSpecialChars	The default value of this parameter is now 0. Integration Server uses the new default value for new Integration Server instances only.
watt.server.password.minUpperChars	The default value of this parameter is now 0. Integration Server uses the new default value for new Integration Server instances only.

Removed Parameter	Replacement, if any
watt.server.jms.debugTrace	Use enhanced logging for the JMS connection aliases used by the JMS triggers for which you want Integration Server to generate additional logging.
watt.server.SOAP.HTTP.useMailWriter	None. This parameter was for use with webMethods Mediator which reached end-of-life in 10.1.
watt.xslt.debug.facList	This parameter is obsolete. Use Integration Server Administrator to set the Server Logger configuration for the facilities in the WmXSLT package.
watt.xslt.debug.level	None. Use Integration Server Administrator to set the server logging level for the facilities in the WmXSLT package.
watt.xslt.debug.logfile	None. Integration Server writes log messages for the WmXSLT package to the server.log.

Deprecated Parameter	Replacement, if any
watt.broker.sync.enableBrokerSync	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.broker.sync.forceDispatcherInit	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.brokerCoder.verbose	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.core.brokerTypeCoder.verbose	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.core.brokerCoder.wireFormat	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.auditDocIdField	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.broker.producer.multiclient	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.broker.replyConsumer.fetchSize	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.

Deprecated Parameter	Replacement, if any
watt.server.broker.replyConsumer.multiclient	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.broker.replyConsumer.sweeperInterval	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.brokerTransport.dur	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.brokerTransport.max	None. WebMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.brokerTransport.ret	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.control.maxPersist	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.control.triggerInputControl.delayIncrementInterval	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.control.triggerInputControl.delays	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.dispatcher.comms.brokerPing	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.dispatcher.comms.connectionShareLimit	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.jms.trigger.maxPrefetchSize	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.jms.trigger.wmjms.clientIDSharing	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.jms.wmjms.lms.readTimeout	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.publish.useCSQ	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.publish.drainCSQInOrder	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.publish.usePipelineBrokerEvent	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.
watt.server.trigger.keepAsBrokerEvent	None. webMethods Broker is deprecated, resulting in the deprecation of this parameter.

Changed Parameter	Description
watt.server.content.type.mappings	Specifies a one-to-one mapping of content types for inbound and outbound requests. Previously, this parameter only mapped wildcards found in the Accept header field to a specific content type for the outbound content handler.
watt.server.http.interceptor.preprocess.sizeLimit	Integration Server now includes support for size units with this server configuration parameter. Set this property to N [KB MB GB], where N is any valid integer. Do not include any spaces between the integer and the unit of measure. If you do not specify a unit of measure, Integration Server treats the supplied N value as bytes.
watt.server.http.preserveUriReservedChars	The server configuration property watt.server.http.preserveUriReservedChars has been modified to include "space" as one of the reserved characters as well.

Release 10.1

Added Parameter	Description
watt.art.notification.publishToJMS.useCSQ	Specifies whether the asynchronous notification of Adapter Runtime can use CSQ (Client side Queueing) when publishing to JMS provider.
watt.server.http.interceptor.enabled	Specifies whether an HTTP interceptor is enabled for Integration Server.
watt.server.http.interceptor.impl	Specifies the fully qualified name of the java class for the HTTP interceptor implementation.
watt.server.http.interceptor.preprocess.sizeLimit	Specifies the maximum number of bytes that an HTTP interceptor reads during preprocessing.
watt.server.http.uriPath.decodePlus	Specifies whether or not the '+' character is interpreted as a ' ' when it appears in the path component of the request URI .
watt.server.jms.trigger.startupFailure.retryCount	Determines the maximum number of retry attempts that Integration Server makes to start a JMS trigger after the trigger fails to start.
watt.server.oauth.requirePost	Indicates whether Integration Server requires that client invoke the pub.oauth services via an HTTP POST request.

Added Parameter	Description
watt.server.rg.internalsocket.timeout	Specifies the length of time, in milliseconds, that Enterprise Gateway Server allows a client request to wait for a connection to the Internal Server before terminating the request with an HTTP 500 Internal Server Error.
watt.server.service.list.treatEmptyAsNull	Specifies whether Integration Server assigns a null value to all list data types like Document List, String List, Document Reference List, and Object list during the execution of a flow service without an input value.
watt.server.stats.logFilesToKeep	Specifies the number of stats.log files that Integration Server keeps on the file system, including the current log file. When Integration Server reaches the limit for the number of stats.log files, each time Integration Server rotates the stats.log, Integration Server deletes the oldest archived stats.log file.
watt.server.stats.logRotateSize	Specifies the file size at which Integration Server rolls over the stats.log.
watt.server.uri.decodePath	Specifies whether or not the '+' character is interpreted as a ' ' when it appears in the path component of the request URI.

Removed Parameter	Replacement, if any
watt.net.defaultBufferSize	None. This parameter specifies the maximum content length of an HTTP request that the WmTomcat package will process. The WmTomcat package has been removed from the product.
watt.net.webapp.cookies.useRelevantPath	None. This parameter specifies how WmTomcat can create fewer cookies to prevent the web application from logging out because of exceeding the browser cookie limit. The WmTomcat package has been removed from the product.

Changed Parameter	Description
watt.net.maxClientKeepaliveConns	The default value of this parameter is now 0. Integration Server uses the new default value for new Integration Server instances only.

Changed Parameter	Description
watt.server.http.preserveUriReservedChars	Changes to this property no longer require a restart of Integration Server to take effect. The changes take effect immediately.
watt.server.oauth.requireHTTPS	If you set the value of this property to true and the client application accesses any of the pub.oauth services over HTTP, Integration Server issues an HTTP 400 error response to the client and writes a service exception to the error log. An HTTP 400 error indicates a client error. Previously, Integration Server issued an HTTP 500 error, which indicates a server error. According to the OAuth Framework , it is a client error when the client does not use a secure channel.

Release 10.0

Added Parameter	Description
watt.core.schema.useUnboundedForMaxOccurs	Specifies the number at which a maxOccurs value greater than this number is treated as unbounded.
watt.net.ssl.client.ftps.useJSSE	<p>Controls the use of JSSE for all of the outbound FTPS connections from Integration Server. Set this parameter to true to use JSSE for all of the outbound FTPS connections. Set this property to false to indicate that JSSE is not used for outbound FTPS connections. The default is false.</p> <p>When executing the pub.client:ftp service or the pub.client:ftp:login service, the value of the useJSSE input parameter overrides the value of the watt.net.ssl.client.ftps.useJSSE server configuration parameter.</p>
watt.server.audit.logFilesToKeep	Specifies the number of audit log files, including the current log file for the audit logger, that Integration Server keeps on the file system for an audit logger that writes to a file. When Integration Server reaches the limit for the number of log files for the audit logger, each time Integration Server rotates the audit log, Integration Server deletes the oldest archived audit log file.

Added Parameter	Description
watt.server.audit.logRotateSize	Specifies the file size at which Integration Server rolls over the audit log for a logger that writes to a file.
watt.server.cache.maxEntriesInCache	Specifies the default value for the Maximum Entries in Cache property for a distributed system cache used in an Integration Server cluster.
watt.server.diagnostic.logFiles.maxMB	Specifies the maximum number of megabytes of data that Integration Server reads from the file system for an audit log while collecting diagnostic data. This parameter affects the collection of audit log files only. It does not affect audit log data read from the database, nor does it affect other log files such as the server log, stats log or terracotta-client logs. The default is 250 megabytes. You do not need to restart Integration Server for changes to this parameter to take effect.
watt.server.jms.trigger.groupTag	Specifies the group tag used in the names of JMS triggers that belong to a trigger group. Integration Server treats JMS triggers with the specified group tag in the name as members of a trigger group.
watt.server.remoteInvoke.queryCSRFToken	Indicates if, during remote service invocation, Integration Server queries the remote server for the CSRF token for the current session and then includes the token in the service request. If an Integration Server uses Cross-Site Request Forgery (CSRF) guard, requests sent to the server must include a CSRF token. If the request does not include a CSRF token, the server rejects the request. When an Integration Server performs a remote invoke to execute a service on another Integration Server that uses CSRF guard, the request needs to include the CSRF token. To ensure that the request includes a CSRF token, the requesting Integration Server obtains the CSRF token for the current session from the remote Integration Server. The requesting Integration Server then modifies the request to include the CSRF token. However, this is only necessary if the remote Integration Server uses CSRF guard.

Added Parameter	Description
watt.server.response.displayISErrorCode	<p>Indicates whether the response from Integration Server to a client application in the case of an error situation includes the Integration Server error code in the header and the body of the response.</p> <p>If you set the value of this parameter to false, the response during error situations includes the error message text without the Integration Server error code in both the header and the body of the response.</p> <p>If you set the value to true, the response during error situations includes the Integration Server error code and the corresponding error message text in the header and body of the response.</p> <p>The default value of the parameter is true.</p>
watt.server.SOAP.default.endpointHTTP	Specifies the default provider web service endpoint alias for the HTTP protocol.
watt.server.SOAP.default.endpointHTTPS	Specifies the default provider web service endpoint alias for the HTTPS protocol.
watt.server.SOAP.HTTP.useMailWriter	Configures Axis stack to use an alternate multipart writer implementation based on the JavaMail AP

Removed Parameter	Replacement, if any
watt.server.event.nerv.subscribeService.user	None. The pub.event.nerv:subscribe service has been removed from the product as of Integration Server version 10.0.

Changed Parameter	Description
watt.net.ssl.client.useJSSE	<p>Controls the use of JSSE only for the outbound HTTPS connections from Integration Server.</p> <p>Note: This change does not impact any FTP connections created using the pub.client:ftp and the pub.client.ftp:login services.</p>

Release 9.12

Added Parameter	Description
watt.core.validation.skipAbsentStarBody	Specifies whether to skip a validation that Integration Server performs when decoding mixed content elements that have an enumeration restriction.
watt.server.apiportal.url	Specifies the URL for establishing a connection to API Portal and publishing a REST API descriptor. This parameter derives the API Portal URL from the following: <ul style="list-style-type: none">• Host name and port number for the API Portal connection.• Tenant for which the REST API descriptor is to be published.• Specifications of the REST API descriptor.
watt.server.jca.connectionPool.threadInterrupter.waitTime	Specifies the maximum number of milliseconds that a thread can take while creating or closing a connection before the pool interrupter thread interrupts the thread. After the specified time elapses, the pool interrupter thread considers the thread to be blocked and interrupts it.
watt.server.jca.connectionPool.threadInterrupter.sleepTime	Specifies the number of milliseconds the pool interrupter thread sleeps between sweeps for server threads that became blocked while creating or closing a connection.
watt.server.json.decodeRealAsString	Converts a real number that Integration Server retrieves from JSON content to String.
watt.server.log.maxEntries	Specifies the default number of log entries to be displayed in the log viewing utility.
watt.server.mediator.directives	Specifies a comma-separated list of Mediator directives.

Added Parameter	Description
watt.server.messaging.debugTrace	Enables an extra level of verbose logging for webMethods messaging triggers that receive messages from Universal Messaging or through Digital Event Services. You can configure the additional logging globally or at the individual webMethods messaging trigger level.
watt.server.ns.decodeJavaService	Enables Integration Server to display non-ASCII Unicode characters in the body of a Java service.
watt.server.ns.logDuplicateDocTypeRegistrationAsError	<p>Indicates whether to suppress or continue logging of error messages related to registration of duplicate universal names for a document type. Set the value of the parameter to false to suppress the logging of the error messages and true to resume the logging. The default value of this parameter is true.</p> <p>Note: Setting the parameter value to false suppresses the error messages about duplicate universal names only. It does not resolve the duplicate names.</p>
watt.server.returnCurrentDateTimeString	Returns the current date and time for an Object field that is not of type java.util.Date and has the format "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'".
watt.server.serverlogFilesToKeep	Specifies the number of server log files that Integration Server keeps on the file system, including the current server log file.
watt.server.session.locale.ignore	Specifies whether the default locale for the pub.date* services is the server locale or the locale from the session used by the client that invoked the service.
watt.server.SOAP.pre82WSD.ignoreVersionMismatch	For a web service provider that was created in an Integration Server release earlier than 8.2.2 and for which the Pre-8.2 compatibility mode property is set to true, specifies whether to emulate pre-8.2 behavior for process SOAP requests.

Added Parameter	Description
watt.server.sftp.dateStampFmt	Specifies the format of date to be used in the SFTP client public services, specifically the pub.client.sftp* services in the WmPublic package. To specify the date format to use, you can use any format that is supported by the Java class java.text.SimpleDateFormat. For example, to display the date with the format 08-12-02 14:44:33:1235, specify dd-MM-yy HH:mm:ss:SSSS. If the watt.server.sftp.dateStampFmt property is not set, Integration Server uses the default format, which is yyyy-MM-dd HH:mm:ss z.
watt.server.transaction.xastore.maxTxnPerFile	Specifies the maximum number of unique XA transactions in an XA recovery log file. When the XA recovery log file reaches the maximum number of transactions, Integration Server creates a new file.
watt.server.transaction.xastore.performXALogging	Specifies whether or not Integration Server writes transaction information to the XA recovery store. Set to true to instruct Integration Server to log information about the state and progress of each XA transaction. Set to false to instruct Integration Server to skip logging XA transaction information. The default is true.
watt.server.um.producer.transaction.commitRetryCount	Specifies the number of attempts made by Integration Server in publishing a guaranteed document to a Universal Messaging server, after the initial attempt at publishing fails because of a transaction failure.
watt.um.clientLog.level	Determines the information that is written to the Universal Messaging client log file, called umClient.log. Each level outputs log entries with that level or higher. Valid values are trace, debug, info, warn, error, fatal, and off. The default is error.

Added Parameter	Description
watt.um.clientLog.size	Maximum size, in MB, of the Universal Messaging client log file. When this size is reached, Integration Server rolls the file over to a backup called umClient (number).log and creates a new file.
watt.um.clientLog.fileDepth	Number of backup log files to keep on disk when using log rolling for the Universal Messaging client log file.

Changed Parameter	Description
watt.net.ftpDataConn	Now, when this parameter is set to true, Integration Server allows parallel downloads from multiple FTP sessions. When this parameter is set to false, Integration Server does not allow parallel downloads and reuses the same FTP session. The default is false.
watt.net.jsse.client.enabledCipherSuiteList	To make it easier to specify a long list of allowed cipher suites, Integration Server now allows specifying a file as the value for this parameter.
watt.net.jsse.server.enabledCipherSuiteList	To make it easier to specify a long list of allowed cipher suites, Integration Server now allows specifying a file as the value for this parameter.
watt.net.ssl.client.cipherSuiteList	To make it easier to specify a long list of allowed cipher suites, Integration Server now allows specifying a file as the value for this parameter.
watt.net.ssl.client.useJSSE	The default value of this parameter is now true, indicating that global default for outbound HTTP requests is to use JSSE. Individual outbound requests can override this setting.
watt.net.ssl.server.cipherSuiteList	To make it easier to specify a long list of allowed cipher suites, Integration Server now allows specifying a file as the value for this parameter.
watt.server.cors.allowedOrigins	Now supports the use of regular expressions in the comma-separated list of allowed origin servers. Integration Server treats any value in the comma-separated list that begins with "r:" as a regular expression.

Removed Parameter**Replacement, if any**

watt.ssh.jsch.kex

Release 9.10**Added Parameter****Description**

watt.net.http401.throwException

Specifies whether the pub.client:http service throws a NetException when receiving a 401 error response or, instead, places the HTTP response header and body in the pipeline.

watt.net.http501-599.throwException

Specifies whether the pub.client:http service throws a ServiceException or returns response headers and response body when receiving a 501 to 599 level response from a remote HTTP server.

watt.security.openid.logExceptions

Specifies whether Integration Server writes OpenID errors to the error log.

watt.security.session.forceReauthOnExpiration

Specifies whether Integration Server accepts or rejects a request that includes an expired or invalid session.

watt.server.audit.um.sessionPool.max

Specifies the maximum number of sessions in the Universal Messaging session pool for each Universal Messaging connection alias used by an audit logger.

watt.server.audit.um.sessionPool.min

Specifies the minimum number of sessions in the Universal Messaging session pool for each Universal Messaging connection alias used by an audit logger.

watt.server.audit.um.sessionPool.retryInterval

Specifies the number of seconds Integration Server waits between attempts to re-establish a session on the Universal Messaging server after an audit logger enters queue fail-fast mode.

watt.server.jca.connectionPool.
createConnection.interrupt.waitTime

Specifies the wait time, measured in milliseconds, that elapses before Integration Server interrupts a connection creation thread that is in a wait state

Added Parameter	Description
watt.server.package.pre82WSD.loadExternalResources	Specifies whether, at package load time, Integration Server loads external resources for a consumer web service descriptor or a WSDL first web descriptor created on a version of Integration Server prior to version 8.2 and for which the Pre-8.2 compatibility mode property is set to true.

Changed Parameter	Description
watt.server.oauth.custom.responseHeader	Specifies whether the OpenID redirection endpoint for Integration Server include a brief description of the error in the response header when an OpenID error or Integration Server exception occurs during authentication. Previously, this parameter applied to OAuth authorization server only.
watt.server.stats.pollTime	Restrictions for the value for the parameter. For a stand-alone Integration Server, the watt.server.stats.pollTime must be an integer greater than or equal to 0 (zero). For an Integration Server in a cluster, the watt.server.stats.pollTime must be an integer greater than 0 (zero) but less than or equal to 60.

Release 9.9

Added Parameter	Description
watt.core.validation. skipNoNamespaceReference	Indicates whether or not Integration Server skips validation of references to elements that are not namespace qualified made from within a namespace qualified element during XML validation against a schema.
watt.net.jsse.client.enabled CipherSuiteList	Specifies, using a comma-separated list, the cipher suites used on JSSE sockets that are used while making outbound HTTPS requests.
watt.net.ssl.client.useJSSE	Controls the use of JSSE for all of the outbound HTTPS connections from Integration Server.

Added Parameter	Description
watt.net.jsse.server.enabled CipherSuiteList	Specifies, using a comma-separated list, the cipher suites used on Integration Server ports that use JSSE and handle inbound requests.
watt.server.audit.failFastLoggers	Specifies a comma-separated list of the loggers that can enter into fail-fast mode. If this parameter is empty and fail-fast is enabled in the ISCoreAudit functional alias, all synchronous audit loggers with a database destination will have fail-fast capability.
watt.server.serverlogRotateSize	Specifies the file size at which Integration Server rolls over the server.log file.
watt.server.SOAP.ignoreMissing ResponseHeader	Determines whether missing required headers in a SOAP response results in an error.
watt.server.SOAP.treatNilAsNull	Indicates whether or not Integration Server generates an @xsi:nil field for an element for which the xsi:nil attribute is set to true.
watt.server.trigger.preprocess. monitorDatabaseOnConnectionException	Indicates whether Integration Server schedules a system task to monitor the document history database when a ConnectionException causes a transient error occurs during trigger preprocessing.

Changed Parameter	Description
watt.net.jsse.server.enabledProtocols	Modified to include "SSLv2Hello" as a default value.
watt.server.soap.decodeElementWithPrefix	Extended to work with the decoding of SOAP response messages received by a web service connector and with the pub.xml:xmlNodeToDocument service Previously, this parameter affected the decoding of SOAP request message received by a web service provider only.
watt.server.trigger.preprocess. suspendAndRetryOnError	This server configuration parameter now applies to all trigger types and messaging providers. Previously, the behavior provided by this property applied only to webMethods messaging triggers that received messages from webMethods Broker. Now, this property also works with webMethods messaging triggers that receive messages from Universal Messaging as well as JMS triggers.

Release 9.8

Added Parameter	Description
watt.cachedirective.exclude.packages	Specifies a comma-separated list of packages whose Dynamic Server Pages you want the browser to cache.
watt.debug.warnOnClasspathError	Specifies whether or not a warning message about a missing classpath entry will be written to standard out.
watt.net.jsse.client.enabledProtocols	Specifies the SSL protocol versions that Integration Server supports when acting as a client making outbound requests.
watt.net.jsse.server.enabledProtocols	Specifies the SSL protocol versions that Integration Server supports when acting as a server handling inbound requests.
watt.net.ssl.randomAlgorithm	Identifies the random algorithm name used by Integration Server on HP-UX.
watt.net.ssl.server.handshake. maxVersion	Specifies the maximum version of the SSL protocol that Integration Server supports when acting as the server handling inbound requests.
watt.net.ssl.server.handshake. minVersion	Specifies the minimum version of the SSL protocol that Integration Server supports when acting as the server handling inbound requests.
watt.server.cluster.action.errorOnStartup	Specifies how Integration Server responds when an error at start up prevents Integration Server from joining the cluster.
watt.server.auth.oauth.accessToken. useHeaderFields	Specifies whether Integration Server performs OAuth authentication when an inbound HTTP/S request includes an access_token in the header fields.
watt.server.auth.oauth.accessToken. useQueryParameters	Specifies whether Integration Server performs OAuth authentication when an inbound HTTP/S request includes an access_token in the query parameter
watt.server.enableHotDeployment	This is an internal property.
watt.server.event.nerv. subscribeService.user	Specifies the default user who will invoke the service specified in the pub.event.nerv:subscribe built-in service when an event is received for a subscribed event type.
watt.server.hotDeploymentTimeout	This is an internal property.
watt.server.hotDeployment AutoRecover	This is an internal property.

Added Parameter	Description
watt.server.json.allowUnquotedFieldNames	Specifies whether or not unquoted field names are allowed in JSON text sent to the pub.json:jsonStringToDocument and pub.json:jsonStreamToDocument services. This parameter also determines whether document types created from JSON text contains fields that correspond to unquoted fields as well as quoted fields
watt.server.json.quoteFieldNames	Specifies whether or not the pub.json:documentToJSONString service encloses all generated JSON field names in double quotes.
watt.server.SOAP.generateRequiredTags	Specifies whether or not a SOAP message generated by Integration Server includes empty element tags for required parameters for which a value was not supplied at run time.
watt.server.SOAP.hideEPRHostInFault	Hides the endpoint reference host name and IP address details in the SOAP fault.
watt.server.xml.encoding	Specifies the encoding that Integration Server must use when processing incoming XML files.
watt.server.xml.xmlNodeToDocument.makeArrayforWS	Specifies how Integration Server decodes duplicate elements contained in an anyType element.

Removed Parameter	Replacement, if any
watt.security.pki.jnditimeout	
watt.security.pki.performSync	

11.0 Added, Removed, Deprecated, or Changed APIs

A release is listed in this section only if changes occurred in that release.

Release 10.2

Added API	Description
com.wm.app.b2b.server.ServerAPI.registerCoderForMultipart	Registers a service and associated coder for a multipart content handler (ContentHandler_Multipart). The specified coder is the content handler for REST requests for that service.

Added API	Description
com.wm.app.b2b.server.ServerAPI. removeCoderForMultipart	Unregisters the coder for the specified service from multipart content handler.

Deprecated API	Replacement, if any
com.wm.app.b2b.server.jms.producer. ProducerFacace.sendLargeMessageStream	None. webMethods Broker is deprecated, resulting in the deprecation of this method.
com.wm.app.b2b.server.jms.consumer. JMSTriggerFacade.getPrefetchSize	None. webMethods Broker is deprecated, resulting in the deprecation of this method.

Release 10.1

Added API	Description
com.softwareag.is.interceptor.HttpInterceptor Exception	Exception that can be thrown from an HttpInterceptorIFC implementation of the preProcess() method to interrupt normal processing of the HTTP request.
com.softwareag.is.interceptor.HttpInterceptorIFC	Defines a set of contracts that any HttpInterceptor must implement.
com.wm.xmldata.XmlData	Class for creating, getting, and setting XmlData content.
com.wm.xmldata.XmlDataAnyTag	Class for creating keys that are used as XmlData *any element wildcards.
com.wm.xmldata.XmlDataAttributeTag	Class for creating XmlDataTags for use as attribute names.
com.wm.xmldata.XmlDataConstants	Class implements static methods that server as shortcuts for various XmlData method names. By using a static import of this class, a program can significantly reduce the syntax clutter resulting from long class names and long method names.
com.wm.xmldata.XmlDataContentTag	Class that is a common superclass for the XmlDataTags that may appear in content models.
com.wm.xmldata.XmlDataCopy	Class that implements a structured copying utility for XmlData infosets.
com.wm.xmldata.XmlDataCursor	Class for navigating and manipulating the XMLData element content.

Added API	Description
com.wm.xmldata.XmlDataElementTag	Class for creating keys that are used as XmlData element names
com.wm.xmldata.XmlDataException	Class that serves as the base class for all exceptions thrown by the XMLData feature.
com.wm.xmldata.XmlDataGroupTag	Class for constructing an XmlData key for group models.
com.wm.xmldata.XmlDataIteratorTag	Class that provides a standard implementation for creating XmlData iterator-tags.
com.wm.xmldata.XmlDataMap	Class that provides the direct manipulation of fields in an XmlData object without the use of cursors.
com.wm.xmldata.XmlDataPath	Class to facilitate the encoding or decoding of an XmlData path as a String literal.
com.wm.xmldata.XmlDataSubstitutionTag	Class for creating specialized element tags that are the heads of substitution groups.
com.wm.xmldata.XmlDataTreeCursor	Class that provides additional capabilities beyond those of the XmlDataCursor. In particular, repeating values can be presented as if flattened so that nextItem() and previousItem() position to individual items rather than arrays of items.
com.wm.xmldata.XmlDataTypeTag	Class for creating an XmlDataTypeTag and returning the value of an XMLDataTag.
com.wm.xmldata.XmlNamespaceMap	Class that maintains a mapping between prefixes and namespace URIs.
com.wm.xmldata.xmldocumenttype.SchemaConverterException	Class for throwing a SchemaConverterException.
com.wm.xmldata.xmldocumenttype.XmlDataSchemaConverter	Class that converts the scripted form of an XmlData document type specification to actual XmlData Document Type namespace nodes.
com.wm.xmldata.xmldocumenttype.XmlDataWorkspace	Class that provides a 'local' workspace in which XmlData document types and fields can be created on an ad hoc basis. This class also allows read-only access to the XmlData document types in the Integration Server namespace.

Removed API

com.wm.data.XmlData
 com.wm.data.XmlDataAnyTag
 com.wm.data.XmlDataAttributeTag
 com.wm.data.XmlDataContentTag
 com.wm.data.XmlDataCursor
 com.wm.data.XmlDataElementTag
 com.wm.data.XmlDataException
 com.wm.data.XmlDataGroupTag
 com.wm.data.XmlDataSubstitutionTag
 com.wm.data.obsolete.XmlData
 com.wm.data.obsolete.XmlDataCursor

Replacement, if any

com.wm.xmldata.XmlData
 com.wm.xmldata.XmlDataAnyTag
 com.wm.xmldata.XmlDataAttributeTag
 com.wm.xmldata.XmlDataContentTag
 com.wm.xmldata.XmlDataCursor
 com.wm.xmldata.XmlDataElementTag
 com.wm.xmldata.XmlDataException
 com.wm.xmldata.XmlDataGroupTag
 com.wm.xmldata.XmlDataSubstitutionTag
 com.wm.xmldata.XmlData
 com.wm.xmldata.XmlDataCursor

Release 10.0**Added API**

com.wm.app.b2b.server.jms.consumer.
 JMSTriggerGroupFacade

Description

Class containing methods for creating, viewing, and deleting JMS trigger groups.

Release 9.12**Added API**

com.wm.data.obsolete.XmlData

Description

Contains the 9.10 version of the com.wm.data.XmlData class. The com.wm.data.XmlData has been revised and simplified in 9.12.

com.wm.data.obsolete.XmlDataCursor

Contains the 9.10 version of the com.wm.data.XmlDataCursor class. The com.wm.data.XmlDataCursor has been revised and simplified in 9.12.

Changed API

Description

`com.wm.data.XmlData`

Revised to present a more consistent programming interface in which there are clearer distinctions between the underlying IData implementation of XmlData and the logical XmlData model as viewed by a typical application. Additionally, the API formalizes aspects of the IData implementation of XmlData to reduce the knowledge of the underlying XmlData implementation that is required to use the methods in the class.

For a detailed list of changes to this class, see the [webMethods Integration Server Java API Reference](#).

Note: If you have any existing Java classes that use this class, you must modify your Java classes to either use new APIs introduced in `com.wm.data.XmlData` or change the Java imports in to reference the 9.10 version of the class which is now contained in `com.wm.data.obsolete.XmlData`.

`com.wm.data.XmlDataCursor`

Revised to present a more consistent programming interface in which there are clearer distinctions between the underlying IData implementation of XmlData and the logical XmlData model as viewed by a typical application. Additionally, the API formalizes aspects of the IData implementation of XmlData to reduce the knowledge of the underlying XmlData implementation that is required to use the methods in the class.

For a detailed list of changes to this class, see the [webMethods Integration Server Java API Reference](#).

Note: If you have any existing Java classes that use this class, you must modify your Java classes to either use new APIs introduced in `com.wm.data.XmlDataCursor` or change the Java imports in to reference the 9.10 version of the class which is now contained in `com.wm.data.obsolete.XmlDataCursor`.

Release 9.10

Added API	Description
com.wm.data.XmlData	Class for creating, getting, and setting XmlData content.
com.wm.data.XmlDataAnyTag	Class for creating keys that are used as XmlData *any element wildcards.
com.wm.data.XmlDataAttributeTag	Class for creating XmlDataTags for use as attribute names.
com.wm.data.XmlDataContentTag	Class that is a common superclass for the XmlDataTags that may appear in content models.
com.wm.data.XmlDataCursor	Class for navigating and manipulating the XmlData element content.
com.wm.data.XmlDataElementTag	Class for creating keys that are used as XmlData element names
com.wm.data.XmlDataException	Class that serves as the base class for all exceptions thrown by the XmlData feature.
com.wm.data.XmlDataGroupTag	Class for constructing an XmlData key for group models.
com.wm.data.XmlDataSubstitutionTag	Class for creating specialized element tags that are the heads of substitution groups.

12.0 Copyright Information

Copyright © 2018 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses>. <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

13.0 Support

Visit the [Empower website](#) to learn about support policies and critical alerts, read technical articles and papers, download products and fixes, submit feature/enhancement requests, and more.

Visit the [TECHcommunity website](#) to access additional articles, demos, and tutorials, technical information, samples, useful resources, online discussion forums, and more.

IS-RM-102-20180417