webMethods Monitor Built-In Services Reference

Version 9.10

April 2016

WEBMETHODS

This document applies to webMethods Monitor Version 9.10 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: MON-BIS-910-20160415

Table of Contents

About this Guide....7
Document Conventions....7
Online Information....8

pub.monitor.archive Folder....9
Summary of Elements in This Folder....10
pub.monitor.archive:documentArchive....11
pub.monitor.archive:getArchiveInfo....12
pub.monitor.archive:processArchive....13
pub.monitor.archive:serverArchive....15
pub.monitor.archive:serviceArchive....16
pub.monitor.archive:setOperationParameters....18

pub.monitor.document Folder....23
Summary of Elements in This Folder....24
pub.monitor.document:exists....24
pub.monitor.document:getActions....25
pub.monitor.document:getDetails....26
pub.monitor.document:getDocument....27
pub.monitor.document:getList....28

pub.monitor.documentControl Folder....33
Summary of Elements in This Folder....34
pub.monitor.documentControl:resubmit....34

pub.monitor.integrationProcessLogging Folder....37
Summary of Elements in This Folder....38
pub.monitor.integrationProcessLogging:createLoggedFieldInstance....39
pub.monitor.integrationProcessLogging:createLoggedFieldMetadata....41
pub.monitor.integrationProcessLogging:createProcessError....42
pub.monitor.integrationProcessLogging:createProcessInstance....44
pub.monitor.integrationProcessLogging:createProcessMetadata....45
pub.monitor.integrationProcessLogging:createStepInstance....46
pub.monitor.integrationProcessLogging:createStepMetadata....47
pub.monitor.integrationProcessLogging:generateProcessInstanceID....48
pub.monitor.integrationProcessLogging:updateProcessInstanceStatus....48
pub.monitor.integrationProcessLogging:updateStepInstanceStatus....49

pub.monitor.process.actions Folder....53
Summary of Elements in This Folder....54
pub.monitor.process.actions:CustomImplOutputDoc....54

pub.monitor.process.instance Folder....55
Summary of Elements in This Folder....56
pub.monitor.process.instance:generateInstanceImage....58
pub.monitor.process.instance:getDocumentNames....60
pub.monitor.process.instance:getFieldNames....60
pub.monitor.process.instance:getInstance....61
pub.monitor.process.instance:getInstanceActivityLogs....63
pub.monitor.process.instance:getInstanceControl....64
pub.monitor.process.instance:getInstanceConversationID....66
pub.monitor.process.instance:getInstanceCustomData....66
pub.monitor.process.instance:getInstanceErrors....69
pub.monitor.process.instance:getInstanceList....70
pub.monitor.process.instance:getInstanceListByQueryName....74
pub.monitor.process.instance:getInstanceListCustomData....75
pub.monitor.process.instance:getInstanceListCustomData Set....79
pub.monitor.process.instance:getInstanceListWithFilter....84
pub.monitor.process.instance:getInstanceModelInfo....90
pub.monitor.process.instance:getInstanceSteps....90
pub.monitor.process.instance:getInstanceTransitions....92
pub.monitor.process.instance:getPagedInstanceList....93
pub.monitor.process.instance:getProcessList....97
pub.monitor.process.instance:getRecentlyCompleted....98
pub.monitor.process.instance:getRecentlyCreated....99
pub.monitor.process.instance:getRecentlyFailed....100
pub.monitor.process.instance:getRecentlySuspended....101

pub.monitor.process.instanceControl Folder....103
Summary of Elements in This Folder....104
pub.monitor.process.instanceControl:changeInstanceStatus....104
pub.monitor.process.instanceControl:changeUserTaskPriority....105
pub.monitor.process.instanceControl:resubmitAllFailed....106
pub.monitor.process.instanceControl:resubmitInstanceStep....107

pub.monitor.process.instanceSteps Folder....109
Summary of Elements in This Folder....110
pub.monitor.process.instanceSteps:getStepActivityLogs....110
pub.monitor.process.instanceSteps:getStepControl....112
pub.monitor.process.instanceSteps:getStepCustomData....113
pub.monitor.process.instanceSteps:getStepDetails....114
pub.monitor.process.instanceSteps:getStepErrors....117
pub.monitor.process.instanceSteps:getStepHistory....118
pub.monitor.process.instanceSteps:getStepPipeline....119

pub.monitor.process.model Folder....121
Summary of Elements in This Folder....122

pub.monitor.process.model:clearProcessFilter..... 123
pub.monitor.process.model:generateModelImage..... 124
pub.monitor.process.model:getCustomFieldDefinitions..... 125
pub.monitor.process.model:getCustomFields..... 126
pub.monitor.process.model:getModelDetails..... 127
pub.monitor.process.model:getModelImage..... 129
pub.monitor.process.model:getModelList..... 130
pub.monitor.process.model:getModelListWithFilter..... 134
pub.monitor.process.model:getModelNames..... 138
pub.monitor.process.model:getModelNamesWithFilter..... 138
pub.monitor.process.model:getProcessFilter..... 140
pub.monitor.process.model:getUnusedModels..... 140
pub.monitor.process.model:setProcessFilter..... 144

pub.monitor.process.modelControl Folder..... 145
Summary of Elements in This Folder..... 146
pub.monitor.process.modelControl:changeModelEnabled..... 146
pub.monitor.process.modelControl:deleteUnusedModel..... 147
pub.monitor.process.modelControl:getProcessLogical Servers..... 148
pub.monitor.process.modelControl:isModelEnabled..... 149
pub.monitor.process.modelControl:refreshModelNames..... 150

pub.monitor.process.modelSteps Folder..... 151
Summary of Elements in This Folder..... 152
pub.monitor.process.modelSteps:getModelSteps..... 152
pub.monitor.process.modelSteps:getModelTransitions..... 155
pub.monitor.process.modelSteps:getStepIDNames..... 156

pub.monitor.service Folder..... 159
Summary of Elements in This Folder..... 160
pub.monitor.service:exists..... 161
pub.monitor.service:getActions..... 162
pub.monitor.service:getActivityLogs..... 163
pub.monitor.service:getChildDetails..... 164
pub.monitor.service:getCustomData..... 165
pub.monitor.service:getDetails..... 167
pub.monitor.service:getErrors..... 168
pub.monitor.service:getHistory..... 169
pub.monitor.service:getList..... 171
pub.monitor.service:getListCustomData..... 175
pub.monitor.service:getListCustomDataSet..... 179
pub.monitor.service:getPipeline..... 183
pub.monitor.service:isResubmittable..... 184

pub.monitor.serviceControl Folder..... 185
Summary of Elements in This Folder..... 186

pub.monitor.serviceControl:resubmit....186

Status Reference....189
Statuses....190

About this Guide

This guide describes the built in services provided in the WmMonitor package.

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies storage locations for services on webMethods Integration Server, using the convention *folder.subfolder:service* .
UPPERCASE	Identifies keyboard keys. Keys you must press simultaneously are joined with a plus sign (+).
Italic	Identifies variables for which you must supply values specific to your own situation or environment. Identifies new terms the first time they occur in the text.
`Monospace font`	Identifies text you must type or messages displayed by the system.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at http://documentation.softwareag.com. The site requires Empower credentials. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

You can find product information on the Software AG Empower Product Support website at https://empower.softwareag.com.

To submit feature/enhancement requests, get information about product availability, and download products, go to Products.

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the Knowledge Center.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at http://techcommunity.softwareag.com. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

1 pub.monitor.archive Folder

- Summary of Elements in This Folder 10
- pub.monitor.archive:documentArchive 11
- pub.monitor.archive:getArchiveInfo 12
- pub.monitor.archive:processArchive 13
- pub.monitor.archive:serverArchive 15
- pub.monitor.archive:serviceArchive 16
- pub.monitor.archive:setOperationParameters 18

Summary of Elements in This Folder

Important: Before you can archive data, you must set up Monitor to enable archiving. For instructions, see the configuration chapter in *webMethods Monitor User's Guide*.

Note: For Oracle, SQL Server, and DB2, the default values for all archive and delete service parameters are stored in the OPERATION_PARAMETER in the Archive database component. You can change the defaults in the table by running the pub.monitor.archive:setOperationParameters service. You can override the defaults for specific archive or delete actions by specifying values on the relevant parameters when you run the archive and delete services.

Note: If you enable JDBC archive and execute any of the `pub.monitor.archive` services, even though the AUDITTIMESTAMP fields of the relevant runtime tables use the local time, the AUDITTIMESTAMP fields are stored in the UTC time in the corresponding archive tables.

Service and Description
pub.monitor.archive:documentArchive Archives or deletes logged document data. The logged data is archived or deleted from the WMDOCUMENT table in the IS Core Audit Log database component and document control data (such as resubmit actions) from the PRA_PROCESS_ACTION table in the IS Core Audit Log database component.
pub.monitor.archive:getArchiveInfo Provides information about the components, tables, number of rows in each table, and the date and time of the oldest and newest records in the tables that are available for archive in the database. The service output is useful to understand the current status of the database before you execute the archive services.
pub.monitor.archive:processArchive Archives or deletes process data and control data (such as resubmit actions) from the PRA_PROCESS_ACTION, WMCUSTOMFIELDDEFINITION, PRA_CUSTOM_PROCESS_DATA, PRA_PROCESS, PRA_CUSTOM_PROCESS, WMPROCESSDEFINITION, WMPROCESSIMAGE, PRA_PROCESS_RECENT, PRA_PROCESS_STEP, WMSTEPDEFINITION, and WMSTEPTRANSITIONDEFINITION tables in the Process Audit Log database component, and archives or deletes process-related service data from the PRA_SERVICE_ACTIVITY and PRA_ERROR tables in the IS Core Audit Log database component.

Service and Description
pub.monitor.archive:serverArchive Archives or deletes server data from the PRA_ERROR, WMSESSION, WMTXIN, and WMTXOUT tables in the IS Core Audit Log database component. Server data includes Integration Server server, session, and guaranteed delivery log entries, and error log entries that are not associated with logged processes, services, or documents (for example, errors that occur during startup or during the run of unlogged processes, services, activations, and documents).
pub.monitor.archive:serviceArchive Archives or deletes service log entries, input pipelines, error data, and user-defined messages from the PRA_ERROR and PRA_SERVICE_ACTIVITY tables in the IS Core Audit Log database component and service control data (such as resubmit actions) from the PRA_PROCESS_ACTION table in the Process Audit Log database component.
pub.monitor.archive:setOperationParameters Sets parameters in the OPERATION_PARAMETER table of the Archive database component if you are using Oracle, SQL Server, or DB2.

pub.monitor.archive:documentArchive

Archives or deletes logged document data. The logged data is archived or deleted from the WMDOCUMENT table in the IS Core Audit Log database component and document control data (such as resubmit actions) from the PRA_PROCESS_ACTION table in the IS Core Audit Log database component.

Input Parameters

todate	**String** Optional. Start date for the period to keep the data in the IS Core Audit Log database component. The period ends with and includes the current date. Use the format yyyy-mm-dd hh:mm:ss. Supply either *todate* or *days* . Do not supply both.
days	**String** Optional. Number of days to keep the data in the IS Core Audit Log database component, ending with and including the current date. Supply either *todate* or *days* . Do not supply both.
archiveAction	**String** Optional. Whether to archive or delete the data.

- `ARCHIVE` Default. Copy the data from the IS Core Audit Log database component to the Archive database component and delete the data from the IS Core Audit Log database component.
- `DELETE` Delete the data from the IS Core Audit Log database component without first archiving it.

batchSize **String** Optional. Number of documents to archive or delete at a time.

Output Parameters

message **String** Message that indicates the success of the archive or delete action.

errorMessage **String** Errors that occurred during the archive or delete action.

Usage Notes

You can use this service with Integration Server Administrator to archive or delete data automatically, at specific times or intervals. To do so, build a service that calls the archive or delete service and sets its input parameters; then run the wrapper service as a scheduled task from Integration Server Administrator. For instructions on scheduling services to run at specific times, see *webMethods Integration Server Administrator's Guide*.

Examples

- *todate* parameter: Suppose the current date is September 30 and you want to archive or delete data that was logged more than 15 days ago (that is, before September 15). You would specify this parameter as `2003-10-15 00:00:00`.
- *days* parameter: Suppose the current date is September 30 and you want to archive or delete data that was logged more than 15 days ago (that is, before September 15). You would specify this parameter as `15`.

pub.monitor.archive:getArchiveInfo

This service output provides information about the components, tables, number of rows in each table, and the date and time of the oldest and newest records in the tables that are available for archive in the database. The service output is useful to understand the current status of the database before you execute the archive services.

Input Parameters

None.

Output Parameters

archiveInfo **Document List** A table containing the following archive information:

- Component to be archived (document, server, service, or process).
- Tables that will be archived for each component.
- Number of rows available for archive in each table.
- Date and time of the oldest record in each table.
- Date and time of the newest record in each table.

pub.monitor.archive:processArchive

Archives or deletes process data and control data (such as resubmit actions) from the PRA_PROCESS_ACTION, WMCUSTOMFIELDDEFINITION, PRA_CUSTOM_PROCESS_DATA, PRA_PROCESS, PRA_CUSTOM_PROCESS, WMPROCESSDEFINITION, WMPROCESSIMAGE, PRA_PROCESS_RECENT, PRA_PROCESS_STEP, WMSTEPDEFINITION, and WMSTEPTRANSITIONDEFINITION tables in the Process Audit Log database component, and archives or deletes process-related service data from the PRA_SERVICE_ACTIVITY and PRA_ERROR tables in the IS Core Audit Log database component.

Specifically, this service archives or deletes the following:

- Process log entries, input pipelines, error data, and run-time values for user-specified input and output document fields.
- Referenced processes, process-related service data (services, service error data, and user-defined messages).
- Process control data (resubmit, suspend, and resume actions).

Input Parameters

todate **String** Optional. Start date for the period to keep the data in the IS Core Audit Log and Process Audit Log database components. The period ends with and includes the current date. Use the format yyyy-mm-dd hh:mm:ss.

Supply either *todate* or *days* . Do not supply both.

days **String** Optional. Number of days to keep the data in the IS Core Audit Log and Process Audit Log database components, ending with and including the current date.

Supply either *todate* or *days* . Do not supply both.

archiveAction **String** Optional. Indicates whether to archive or delete the data.

- `ARCHIVE` Default. Copy the data from the IS Core Audit Log and Process Audit Log database components to the Archive database component and delete the data from the IS Core Audit Log and Process Audit Log database components.
- `DELETE` Delete the data from the IS Core Audit Log and Process Audit Log database components without first archiving it.

status **String** Optional. Indicates the status on which to archive or delete the data.

- `COMPLETED` Default. Archive or delete data for processes with a status of Completed.
- `COMPLETED-FAILED` Archive or delete data for processes with a status of Completed, Failed, Stopped, or Resubmitted.

Data for processes with a status other than those specified is retained in the database.

batchSize **String** Optional. Number of processes to archive or delete at a time.

modelId **String array** Optional. Complete model ID of the model for the process instance(s) you want to archive. Use a comma to separate multiple *modelId* string values. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames /PROCESSKEY* output parameter. If *modelId* is null, all processes are archived.

Note: Whether Monitor treats *modelId* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

message **String** Message that indicates the success of the archive or delete action.

errorMessage **String** Errors that occurred during the archive or delete action.

Usage Notes

You can use this service with Integration Server Administrator to archive or delete data automatically, at specific times or intervals. To do so, build a service that calls the archive or delete service and sets its input parameters, then run the wrapper service as

a scheduled task from Integration Server Administrator. For instructions on scheduling services to run at specific times, see *webMethods Integration Server Administrator's Guide*.

Examples

- *todate* parameter: Suppose the current date is September 30 and you want to archive or delete data for processes that finished running more than 15 days ago (that is, before September 15). You would specify this parameter as `2003-10-15 00:00:00`.
- *days* parameter: Suppose the current date is September 30 and you want to archive or delete data for processes that finished running more than 15 days ago (that is, before September 15). You would specify this parameter as `15`.

pub.monitor.archive:serverArchive

Archives or deletes server data from the PRA_ERROR, WMSECURITY, WMSESSION, WMTXIN, and WMTXOUT tables in the IS Core Audit Log database component. Server data includes Integration Server server, session, and guaranteed delivery log entries, and error log entries that are not associated with logged processes, services, or documents (for example, errors that occur during startup or during the run of unlogged processes, services, activations, and documents).

Input Parameters

todate **String** Optional. Start date for the period to keep the data in the IS Core Audit Log database component. The period ends with and includes the current date. Use the format yyyy-mm-dd hh:mm:ss.

Supply either *todate* or *days* . Do not supply both.

days **String** Optional. Number of days to keep the data in the IS Core Audit Log database component, ending with and including the current date.

Supply either *todate* or *days* . Do not supply both.

archiveAction **String** Optional. Indicates whether to archive or delete the data.

- `ARCHIVE` Default. Copy the data from the IS Core Audit Log database component to the Archive database component and delete the data from the IS Core Audit Log database component.
- `DELETE` Delete the data from the IS Core Audit Log database component without first archiving it.

batchSize **String** Optional. This parameter is ignored.

Output Parameters

message	**String** Message that indicates the success of the archive or delete action.
errorMessage	**String** Errors that occurred during the archive or delete action.

Usage Notes

You can use this service with Integration Server Administrator to archive or delete data automatically, at specific times or intervals. To do so, build a service that calls the archive or delete service and sets its input parameters, then run the wrapper service as a scheduled task from Integration Server Administrator. For instructions on scheduling services to run at specific times, see *webMethods Integration Server Administrator's Guide*.

Examples

- *todate* parameter: Suppose the current date is September 30 and you want to archive or delete data that was logged more than 15 days ago (that is, before September 15). You would specify this parameter as `2003-10-15 00:00:00`.
- *days* parameter: Suppose the current date is September 30 and you want to archive or delete data that was logged more than 15 days ago (that is, before September 15). You would specify this parameter as `15`.

pub.monitor.archive:serviceArchive

Archives or deletes service log entries, input pipelines, error data, and user-defined messages from the PRA_ERROR and PRA_SERVICE_ACTIVITY tables in the IS Core Audit Log database component and service control data (such as resubmit actions) from the PRA_PROCESS_ACTION table in the Process Audit Log database component.

Note: Monitor can archive user-defined messages for a service only if customized logging is set up for the service in Designer. That is, if service logging is globally enabled in Integration Server but customized logging is not set up for the service in Designer, Monitor cannot archive used-defined messages that a service writes.

Input Parameters

todate	**String** Optional. Start date for the period to keep the data in the IS Core Audit Log and Process Audit Log database components. The period ends with and includes the current date. Use the format yyyy--mm-dd hh:mm:ss.

Supply either *todate* or *days* . Do not supply both.

days **String** Optional. Number of days to keep the data in the IS Core Audit Log and Process Audit Log database components, ending with and including the current date.

Supply either *todate* or *days* . Do not supply both.

archiveAction **String** Indicates whether to archive or delete the data.

- `ARCHIVE` Default. Copy the data from the IS Core Audit Log and Process Audit Log database components to the Archive database component and delete the data from the and Process Audit Log database components.
- `DELETE` Delete the data from the IS Core Audit Log and Process Audit Log database components without archiving it first.

status **String** Indicates the status for which to archive or delete the data.

- `COMPLETED` Default. Archive or delete data for services with a status of Completed. Data for services with other statuses is retained in the database.
- `COMPLETED-FAILED` Archive or delete data for services with a status of Completed, Failed, or Resubmitted. Data for services with a status of other than those specified is retained in the database.
- `ALL` Archive or delete data for services with any status.

batchSize **String** Optional. Number of services to archive or delete at a time.

Output Parameters

message **String** Message that indicates the success of the archive or delete action.

errorMessage **String** Errors that occurred during the archive or delete action.

Usage Notes

You can use this service with Integration Server Administrator to archive or delete data automatically, at specific times or intervals. To do so, build a wrapper service that calls the archive or delete service and sets its input parameters; then run the wrapper service as a scheduled task from Integration Server Administrator. For instructions on scheduling services to run at specific times, see *webMethods Integration Server Administrator's Guide*.

Examples

- *todate* parameter: Suppose the current date is September 30 and you want to archive or delete data for 4.x activations that finished running more than 15 days ago (that is, before September 15). You would specify this parameter as `2003-10-15 00:00:00`.
- *days* parameter: Suppose the current date is September 30 and you want to archive or delete data for 4.x activations that finished running more than 15 days ago (that is, before September 15). You would specify this parameter as `15`.

pub.monitor.archive:setOperationParameters

Sets parameters in the OPERATION_PARAMETER table of the Archive database component if you are using Oracle, SQL Server, or DB2.

The names of the parameters in the table below exactly match the column names in the table.

Input Parameters

COMMIT_SIZE

String Optional. Number of documents, processes, or services to archive or delete at a time.

Note: This parameter is ignored for server data.

PROCESSAUDIT_DBLINK

String Optional. If you installed the Archive database component in Oracle or SQL Server, you want to archive from the Process Audit Log, and the Archive and Process Audit Log database components are on different database servers, enter the DBlink (Oracle) or Linked Server (SQL Server) name to use to link the database components. If you do not want to archive from the Process Audit Log database component, or the database components are on the same database server, enter null.

Note: If you are using DB2, you were required to install the Archive database component in the same database as the Process Audit Log database components, so no linking is necessary; enter null.

ISCOREAUDIT_DBLINK

String Optional. If you installed the Archive database component in Oracle or SQL Server, you want to archive from the IS Core Audit Log, and the Archive and IS Core Audit Log database components are on different database servers, enter the DBlink (Oracle) or Linked Server (SQL Server) name to use to link the

database components. If you do not want to archive from the IS Core Audit Log database component, or the database components are on the same database server, enter null.

Note: If you are using DB2, you were required to install the Archive database component in the same database as the IS Core Audit Log database components, so no linking is necessary; enter null.

PROCESS_SCHEMA

String Optional. To archive from the Process Audit Log database component, enter the following for the Process Audit Log database component. If the RDBMS is:

- **Oracle** Enter the database user.
- **SQL Server** Enter the database name.
- **DB2** Enter the schema name.

If you do not want to archive from the Process Audit Log database component, enter null.

ISCORE_SCHEMA

String Optional. To archive from the IS Core Audit Log database component, enter the following for the IS Core Audit Log database component. If the RDBMS is:

- **Oracle** Enter the database user.
- **SQL Server** Enter the database name.
- **DB2** Enter the schema name.

If you do not want to archive from the IS Core Audit Log database component, enter null.

PROCESS_DAYS_TO_RETAIN

String Optional. Default number of days to keep the process data in the IS Core Audit Log and Process Audit Log database components, ending with and including the current date. The service archives or deletes process data that is older than the retention period.

PROCESS_ARCHIVE_ACTION

String Optional. Default action to indicate whether a service should archive or delete the process data.

- `ARCHIVE` Default. Copy the process data from the IS Core Audit Log and Process Audit Log database components to the Archive database component.
- `DELETE` Delete the process data from the IS Core Audit Log and Process Audit Log database components without archiving it first.

PROCESS_STATUS_CRITERIA

String Optional. Default statuses of the process data that services should use to archive or delete the process data. Process data in other statuses is retained in the IS Core Audit Log and Process Audit Log database components.

- `2` Default. Archive or delete data for processes with a status of Completed.
- `4` Archive or delete data for processes with a status of Failed.
- `1024` Archive or delete data for processes with a status of Stopped.
- `32768` Archive or delete data for processes with a status of Resubmitted.

SERVICE_DAYS_TO_RETAIN

String Optional. Default number of days to keep the service data in the or IS Core Audit Log database component, ending with and including the current date. The service archives or deletes service data that is older than the retention period.

SERVICE_ARCHIVE_ACTION

String Optional. Default action to indicate whether a service should archive or delete the service data.

- `ARCHIVE` Default. Copy the service data from the IS Core Audit Log database component to the Archive database component.
- `DELETE` Delete the service data from the IS Core Audit Log database component without archiving it first.

SERVICE_STATUS_CRITERIA

String Optional. Default statuses of the service data that services should use to archive or delete the service data. Service data in other statuses is retained in the IS Core Audit Log database component.

- `2` Default. Archive or delete data for services with status of Completed.
- `4` Archive or delete data for services with a status of Failed.
- `32768` Archive or delete data for services with a status of Resubmitted.
- `32776` Archive or delete data for services with a status of Activity.

ACTIVATION_DAYS_TO_RETAIN

String Optional. Default number of days to keep the 4.x activation data in the or IS Core Audit Log database component, ending with and including the current

date. The service archives or deletes 4.x activation data that is older than the retention period.

Note: Logging data for activations has been deprecated.

ACTIVATION_ARCHIVE_ACTION

String Optional. Default action to indicate whether a service should archive or delete the 4.x activation data.

Note: Logging data for activations has been deprecated.

- `ARCHIVE` Default. Copy the 4.x activation data from the IS Core Audit Log database component to the Archive database component.
- `DELETE` Delete the 4.x activation data from the IS Core Audit Log database component without archiving it first.

ACTIVATION_STATUS_CRITERIA

String Optional. Default statuses of the 4.x activation data that services should use to archive or delete the 4.x activation data. Data for 4.x activations in other statuses is retained in the IS Core Audit Log database component.

Note: Logging data for activations has been deprecated.

- `2` Default. Archive or delete data for 4.x activations with a status of Completed.
- `4` Archive or delete data for 4.x activations with a status of Failed.

DOCUMENT_DAYS_TO_RETAIN

String Optional. Default number of days to keep the document data in the or IS Core Audit Log database component, ending with and including the current date. The service archives or deletes document data that is older than the retention period.

DOCUMENT_ARCHIVE_ACTION

String Optional. Default action to indicate whether a service should archive or delete the document data.

- **ARCHIVE**. Default. Copy the document data from the IS Core Audit Log database component to the Archive database component.
- **DELETE**. Delete the document data from the IS Core Audit Log database component without archiving it first.

SERVER_DAYS_TO_RETAIN

String Optional. Default number of days to keep the server data in the or IS Core Audit Log database

component, ending with and including the current date. The service archives or deletes server data that is older than the retention period.

SERVER_ARCHIVE_ACTION **String** Optional. Default action to indicate whether a service should archive or delete the server data.

- **ARCHIVE**. Default. Copy the server data from the IS Core Audit Log database component to the Archive database component.
- **DELETE**. Delete the server data from the IS Core Audit Log database component without archiving it first.

Output Parameters

message **String** Message that indicates that the parameters have been set.

Usage Notes

You can specify more than one status on the *_STATUS_CRITERIA* parameters. To do so, separate the codes using commas.

Examples

- *PROCESS_STATUS_CRITERIA* parameter: Suppose you want to archive processes with a status of Completed or Failed, specify `2,4` for the *PROCESS_STATUS_CRITERIA* parameter.
- *SERVICE_DAYS_TO_RETAIN* parameter: Suppose the current date is September 30 and you want to archive or delete data for services that finished running more than 15 days ago (that is, before September 15); specify this parameter as `15`.

2 pub.monitor.document Folder

- Summary of Elements in This Folder 24
- pub.monitor.document:exists 24
- pub.monitor.document:getActions 25
- pub.monitor.document:getDetails 26
- pub.monitor.document:getDocument 27
- pub.monitor.document:getList 28

Summary of Elements in This Folder

Service and Description
pub.monitor.document:exists Determines whether a specified document exists in the logging database.
pub.monitor.document:getActions Gets all resubmit actions associated with a specified document.
pub.monitor.document:getDetails Gets the most recently logged details about a specified document.
pub.monitor.document:getDocument Retrieves a specified document from the logging database.
pub.monitor.document:getList Retrieves a list of documents that meet criteria that you specify.

pub.monitor.document:exists

Determines whether a specified document exists in the logging database.

Input Parameters

documentID **String** Document ID of the document to check for in the logging database. Specify the complete, exact ID.

> **Note:** Whether the *documentID* is treated as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

exists **String** Indicates whether the document exists in the logging database.

- `true` Document exists in the database.

- `false` Document does not exist in the database.

pub.monitor.document:getActions

Gets all resubmit actions associated with a specified document.

Input Parameters

documentID **String** Document ID of the document for which you want to obtain resubmit actions. Specify the complete, exact ID.

> **Note:** Whether the *documentID* is treated as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

actions **Document List** The retrieved resubmit actions; each row contains these fields:

- *ROOTCONTEXTID* **String** The internal identifier that Integration Server uses.
- *PARENTCONTEXTID* **String** The internal identifier that Integration Server uses.
- *CONTEXTID* **String** The internal identifier that Integration Server uses.
- *DOCUMENTID* **String** Document ID for which the resubmit action was logged.
- *DOCUMENTNAME* **String** Document name for which the resubmit action was logged.
- *ACTION* **Number** Numerical code that represents the resubmit action, that is, `2`.
- *ACTIONDECODE* **String** Localized keyword value for the resubmit action, that is (in English), "Document Resubmit."
- *USERNAME* **String** User that performed the resubmit action.
- *SERVERID* **String** Server ID associated with the resubmitted document.

- For webMethods Broker-logged documents, ID of the publishing webMethods Broker.
- For in doubt documents, documents that failed during delivery or retrieval, and documents where the retries were exceeded and could not be delivered, ID of the intended recipient.
- For documents that failed during publishing and for documents where the retries were exceeded and could not be published, no server ID is listed.

- *AUDITTIMESTAMP* **Number** Time the resubmit action was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the resubmit action was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

pub.monitor.document:getDetails

Gets the most recently logged details about a specified document.

Input Parameters

documentID **String** Document ID of the document for which to get logged details. Specify the complete, exact ID. If there are multiple entries with the same document ID, the service returns all the entries.

> **Note:** Whether the *documentID* is treated as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

details **Document List** The retrieved details. For each document that matches the specified document ID, the service returns the following logged details:

- *DOCUMENTID* **String** Document ID you specified; that is the document ID to which these details apply.
- *DOCUMENTNAME* **String** Document name of the document.

- *STATUS* **Number** Document type of the document. The service returns the numerical value that represents the type, as follows:
  - `0`Broker
  - `4` Failed
  - `512` Retries Exceeded
  - `32770` In question
- *STATUSDECODE* **String** The localized keyword value for the document type. The keyword (in English) will be one of: Broker, Retries Exceeded, In Doubt, or Failed.
- *ENQUEUETIMESTAMP* **String** If a webMethods Broker logged the document, the time Broker first enqueued the document; otherwise it is null. The timestamp is in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *SERVERID* **String** The server ID associated with the document:
  - For webMethods Broker-logged documents, ID of the publishing webMethods Broker.
  - For in doubt documents, documents that failed during delivery or retrieval, and documents where the retries were exceeded and could not be delivered, ID of the intended recipient.
  - For failed documents that failed during publishing and for retries exceeded documents that could not be published, no server ID is listed.
- *AUDITTIMESTAMP* **Number** Time this document was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time this document was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

pub.monitor.document:getDocument

Retrieves a specified document from the logging database.

Input Parameters

documentID **String** Document ID of the document you want to retrieve. Specify the complete, exact ID.

> **Note:** Whether the *documentID* is treated as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

document **Document** The retrieved document.

pub.monitor.document:getList

Retrieves a list of documents that meet criteria that you specify.

> **Note:** Whether the search is case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

documentID **String** Optional. A complete or partial document ID of the documents you want in the list.

type **String** Optional. The document type to list. Use *type* if you want the list to contain all documents of one document type. You can specify either a keyword (for example, `Broker`) or a numerical value that represents the type (for example, `0`). Use the value that corresponds to the type of document you want to retrieve.

- `Broker` or `0`Broker-logged documents
- `In Doubt` or `32770` In doubt documents
- `Retries Exceeded` or `512` Retries exceeded documents
- `Failed` or `4` Failed documents

typeSet **String List** Optional. A set of document types to retrieve. Use *typeSet* when you want the list to contain documents of more than one type. For each document type, you can specify either

a keyword (for example, `Broker`) or a numerical value that represents the type (for example, `0`). Valid values are:

- `Broker` or `0`Broker-logged documents
- `In Doub`t or `32770` In doubt documents
- `Retries Exceeded` or `512` Retries exceeded documents
- `Failed` or `4` Failed documents

documentName **String** Optional. A complete or partial document name on which to match. This works with the *documentNameExact* parameter.

documentNameExact **String** Optional. Whether the service should perform an exact match on the document name you specify in *documentName* .

- `true` Get documents whose document name exactly matches the *documentName* parameter.
- `false` Default. Get documents whose document name contains a substring that matches the *documentName* parameter.

clientID **String** Optional. A complete or partial client ID to match. This parameter works with the *clientIDExact* parameter. Supply the client ID that matches the document type you want to retrieve:

- For Broker-logged documents, the IDs of the Brokers that logged the documents.
- For in doubt documents, the IDs of the Broker clients associated with the triggers that processed the documents originally.
- For documents that failed and exceeded retries during delivery, the IDs of the original destination Broker clients.
- For documents that failed documents during retrieval, the IDs of the Broker clients associated with the triggers for which Integration Server tried to retrieve the documents originally.

Note: For documents that failed and exceeded retries during publication, there is no client ID on which to search.

The format for Broker IDs is *Broker@host:port* (for example, `CustOps@qatest07:6849`). The format for IDs of Broker clients is *clientprefix_folder1.folder2.foldern_trigger* (for example, `joesmith_documents.history.triggers_MsgHistoryTrigger`).

clientIDExact **String** Optional. Whether the service should perform an exact match on the client ID you specify in *clientID* . Valid values are:

- `true` Get documents where the document name exactly matches the *clientID* parameter.
- `false` Default. Get documents with a document name contains a substring that matches the *clientID* parameter.

range

String Optional. A date range for the documents you want in the list. The date range identifies the date when the documents were logged. If you use this parameter, do not use the *fromDate* or *toDate* parameter. A week is Sunday through Saturday.

- `Today` Current date.
- `Yesterday` Yesterday.
- `In the last 7 days` Within the last 7 days, including today.
- `Last week` Any day in the previous calendar last week.
- `This week` Any day in current calendar week.
- `Last month` Any day in the previous calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

fromDate

String Optional. The start date of when the documents were logged. The service will get all documents logged on or after this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *toDate* to specify an end date; do not use the *range* parameter when you use the *fromDate* and *toDate* parameters.

toDate

String Optional. The end date of when the documents were logged. The service will get all documents logged on or before this date. Use the format *YYYY-MM-DD HH:MM:SS* .

maxRows

String Optional. Maximum number of documents to find, starting with those most recently logged. By default, the service gets all documents.

sortColumn

String Optional. How to sort the returned list of documents. This parameter works with the *sortAscending* parameter.

- `DOCUMENTID` Document IDs of the returned documents.
- `DOCUMENTNAME` Document names of the returned documents.
- `STATUS` Document types of the returned documents, sorted based on the numerical order as follows:
  - `0` Broker
  - `4` Failed

- `512` Retries Exceeded
- `32770` In Doubt

- `SERVERID` Server ID associated with the returned documents.
- `ENQUEUETIMESTAMP` Time that Broker first enqueued the returned documents. This is only available for documents logged by Broker.
- `AUDITTIMESTAMP` Default. Time the returned documents were logged.

sortAscending **String** Optional. Whether to sort the returned list of documents in ascending or descending order. The documents are sorted by the field identified by the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

isAnd **String** Optional. Whether the service is to use an AND or an OR condition for the criteria specified in the input parameters.

- `true` Default. Use an AND condition. The service returns documents that match all of the criteria.
- `false` Use an OR condition. The service returns documents that match any of the criteria.

Output Parameters

documents **Document List** List of documents that match the specified criteria. For each document, the following fields are returned:

- *DOCUMENTID* **String** Document ID of a returned document.
- *DOCUMENTNAME* **String** Document name of a returned document.
- *STATUS* **Number** Document type of a returned document. The service returns the numerical value that represents the document type:
  - `0`Broker-logged documents.
  - `4` Failed documents.
  - `32770` In doubt documents.
  - `512` Retries exceeded documents.
- *STATUSDECODE* **String** The localized keyword value for the document type. The keyword (in English) is one of: Broker, Retries Exceeded, In Doubt, or Failed.

- *ENQUEUETIMESTAMP* **String** If the document was logged by a Broker, the time the Broker first enqueued the document. The timestamp is in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *SERVERID* **String** The server ID associated with the documents:
  - For Broker-logged documents, ID of the publishing Broker.
  - For in doubt documents, documents that failed during delivery or retrieval, and documents where the retries were exceeded, ID of the intended recipient.
  - For documents that failed and exceeded retries during publishing, no client ID is listed.
- *AUDITTIMESTAMP* **Number** Time the document was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time this document was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

3 pub.monitor.documentControl Folder

- Summary of Elements in This Folder 34
- pub.monitor.documentControl:resubmit 34

Summary of Elements in This Folder

Service and Description
pub.monitor.documentControl:resubmit Resubmits a document.

pub.monitor.documentControl:resubmit

Resubmits a document.

Note: Monitor handles two types of documents:

1. Documents sent directly from Broker using the WmLogUtil package.
2. Documents marked as in-doubt by Integration Server triggers.

In both cases, documents can be viewed and resubmitted from Monitor. However, use case 1 is not supported when Universal Messaging is in use. Use case 2 is supported with Universal Messaging.

Input Parameters

documentID **String** Document ID of the document to resubmit. Specify the complete, exact ID.

Note: Whether Monitor treats *documentID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

document **Document** Optional. Document to resubmit. To retrieve the document, use the pub.monitor.document:getDocument service. If you do not provide *document* or if you do not have the functional privilege to resubmit the document, the service retrieves the document logged in the database under the specified document ID.

Output Parameters

None.

Usage Notes

Monitor resubmits documents as follows:

For this type of document	Monitor...
Broker-logged documents	Publishes the documents to the webMethods Broker to which the Monitor-equipped Integration Server is connected.
In doubt documents	Delivers the documents to the triggers that processed the documents originally.
Failed documents that failed during delivery	Delivers the documents to the original destination webMethods Broker clients.
Failed documents that failed during publication	Publishes the documents to the webMethods Broker to which the Monitor-equipped Integration Server is connected.
Failed documents that failed during retrieval	Delivers the documents to the triggers for which Integration Server tried to retrieve the documents originally.
Retries exceeded documents that were exceeded during delivery	Delivers the documents to the original destination Broker clients.
Retries exceeded documents that were exceeded during publication	Publishes the documents to the webMethods Broker to which the Monitor-equipped Integration Server is connected.

4 pub.monitor.integrationProcessLogging Folder

- Summary of Elements in This Folder 38
- pub.monitor.integrationProcessLogging:createLoggedFieldInstance 39
- pub.monitor.integrationProcessLogging:createLoggedFieldMetadata 41
- pub.monitor.integrationProcessLogging:createProcessError 42
- pub.monitor.integrationProcessLogging:createProcessInstance 44
- pub.monitor.integrationProcessLogging:createProcessMetadata 45
- pub.monitor.integrationProcessLogging:createStepInstance 46
- pub.monitor.integrationProcessLogging:createStepMetadata 47
- pub.monitor.integrationProcessLogging:generateProcessInstanceID 48
- pub.monitor.integrationProcessLogging:updateProcessInstanceStatus 48
- pub.monitor.integrationProcessLogging:updateStepInstanceStatus 49

Summary of Elements in This Folder

Note: The pub.monitor.integrationProcessLoggingservices pass data to the database through the Integration Server audit service. The Integration Server audit service might take some time to store data in the database. Make sure you introduce a delay between each pub.monitor.integrationProcessLogging service execution to avoid any errors. For information about how to use these services, see information about integration process logging in *webMethods Monitor User's Guide*.

Service and Description
pub.monitor.integrationProcessLogging:createLoggedFieldInstance Creates an instance of a logged field, setting the field's value to the value that you specify and associating the logged field with the step in the process instance that you specify.
pub.monitor.integrationProcessLogging:createLoggedFieldMetadata Adds a definition for a logged field (known as logged field metadata) to the Process Audit Log database. The logged field definition is associated with a specific step definition that is part of a specific integration process definition.
pub.monitor.integrationProcessLogging:createProcessError Adds a process error message to an existing process instance in the Process Audit Log.
pub.monitor.integrationProcessLogging:createProcessInstance Creates a new process instance of a specified integration process.
pub.monitor.integrationProcessLogging:createProcessMetadata Adds a definition for an integration process (known as process metadata) to the Process Audit Log database.
pub.monitor.integrationProcessLogging:createStepInstance Creates a new step instance of the specified process step.
pub.monitor.integrationProcessLogging:createStepMetadata

Service and Description
Adds a definition for a step within an integration process (known as step metadata) to the Process Audit Log database.
pub.monitor.integrationProcessLogging:generateProcessInstanceID Generates a unique process instance ID.
pub.monitor.integrationProcessLogging:updateProcessInstanceStatus Updates the process status of an existing process instance.
pub.monitor.integrationProcessLogging:updateStepInstanceStatus Updates the step status of an existing step instance.

pub.monitor.integrationProcessLogging:createLoggedFieldInstance

Creates an instance of a logged field, setting the field's value to the value that you specify and associating the logged field with the step in the process instance that you specify.

Input Parameters

processKey

String The internal identifier (that is, process key) of the integration process definition with which the logged field is associated. You define the process key when you add the process definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createProcessMetadata service.

A process key is 1 to 64 alphanumeric characters.

processInstanceID

String The identifier of the process instance for which you are logging a field. You assign the process instance an identifier when you create the process instance using the pub.monitor.integrationProcessLogging:createProcessInstance service.

A process instance ID is 1 to 32 alphanumeric characters.

stepID

String The internal identifier of the process step with which the logged field is associated. You define the step identifier and associate it with a process step when you add the step definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createStepMetadata service.

A step identifier is 1 to 128 alphanumeric characters.

stepIteration	**Number** Optional. The iteration of the step. Specify an iteration count if the step is executed multiple times within a single process instance, for example, if the step is within a loop or is an error handling step that can be executed more than one time. By default, the service uses a value of 1 for *stepIteration* .
documentName	**String** Optional. The name of the document that contains the logged field. The value you specify for *documentName* must be 1 to 128 characters and can include characters that are valid in IS service names. See the *webMethods Service Development Help* for information about allowed characters.
fieldName	**String** The internal name of the logged field for which you want to create an instance. You define the internal name when you add the logged field definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createLoggedFieldMetadata service. A field name is 1 to 512 alphanumeric characters.
fieldValue	**Object** Optional. The BLOB value of the logged field. Use *fieldValue* if `Blob` is specified for the input parameter *fieldType* of the pub.monitor.integrationProcessLogging:createLoggedFieldMetadata service when the logged field definition is added.
stringValue	**String** Optional. The string value of the logged field. Use *stringValue* if `String` is specified for the input parameter *fieldType* of the pub.monitor.integrationProcessLogging:createLoggedFieldMetadata service when the logged field definition is added. This value can be 1 to 255 characters.
numberValue	**Number** Optional. The number value of the logged field. Use *numberValue* if `Number` is specified for the input parameter *fieldType* of the pub.monitor.integrationProcessLogging:createLoggedFieldMetadata service when the logged field definition is added.
dateValue	**Date** Optional. The date value of the logged field. Use *dateValue* if `Date` is specified for the input parameter *fieldType* of the pub.monitor.integrationProcessLogging:createLoggedFieldMetadata service when the logged field definition is added.

Output Parameters

result	**String** The outcome of creating the logged field instance. If the service successfully created the logged field instance, *result* is "success." If the service encountered an error attempting to create the logged field instance, *result* contains the exception.

Usage Notes

- This service uses a combination of the values you specify for *processInstanceID* , *stepID* , *stepIteration*, and *fieldName* to locate the logged field definition for which you want create a field instance and supply a value.
- The service requires that you specify a value for the logged field using one of *fieldValue* , *stringValue* , *numberValue* , or *dateValue* . Use the variable that matches the data type used in the logged field definition. The data type is defined using the *fieldType* input parameter of the pub.monitor.integrationProcessLogging:createLoggedFieldMetadata service.
- If you specify values in more than one parameters (for example, if you specify values for both *fieldValue* and *stringValue*), the service uses the value that is associated with the data type of the logged field definition.
- If the service encounters an error, it logs an error to the PRA_ERROR table. You can view the error using Monitor with the Step Instance Detail page for the step instance.

pub.monitor.integrationProcessLogging:createLoggedFieldMetadata

Adds a definition for a logged field (known as logged field metadata) to the Process Audit Log database. The logged field definition is associated with a specific step definition that is part of a specific integration process definition.

The logged field definition is comparable to custom data for a process model that was created with Designer. In the same way you need to create a field defined in a document before you can specify a value for that field, you need to define a logged field definition for an integration process before you can specify a value for the field.

Input Parameters

processKey	**String** The internal identifier (that is, process key) of the integration process definition with which the logged field is to be associated. You defined the process key when you added the process definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createProcessMetadata service. A process key is 1 to 64 alphanumeric characters.

stepID **String** The internal identifier of the process step with which the logged field is to be associated. You defined the step identifier and associated it with a process step when you added the step definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createStepMetadata service.

A step identifier is 1 to 128 alphanumeric characters.

documentType **String** A name that identifies the type of document with which the logged field is to be associated. For example, you might specify a name like "purchaseOrder" or you might specify the fully qualified name of an existing IS document type. The document type you specify does *not* have to be an existing document defined in a webMethods component, such as a webMethods Broker document type, IS document type, or TN document type. The value you specify for *documentType* must be 1 to 512 characters and can include characters that are valid in IS service names. See *webMethods Service Development Help* for information about allowed characters.

fieldName **String** The internal name you want to assign to the logged field. The value you specify for *fieldName* must be 1 to 512 alphanumeric characters. Use this name in subsequent services when you want to reference this logged field definition. It is an internal name only; Monitor does not display this value.

fieldAlias **String** The external name you want to assign to the logged field. The value you specify for *fieldAlias* must be 1 to 640 alphanumeric characters. Monitor displays this value when listing information about the logged field.

fieldType **String** The data type of the logged field. Specify one of the following: `String`, `Number`, `Date`, or `Blob`.

Output Parameters

result **String** The outcome of adding the logged field definition. If the service successfully added the logged field definition, *result* is "success." If the service encountered an error attempting to add the logged field definition, *result* contains the exception.

pub.monitor.integrationProcessLogging:createProcessError

Adds a process error message to an existing process instance in the Process Audit Log.

Note: This service does not cause the integration process to fail. It also leaves the process status unchanged; that is, the process status is *not* updated to Failed.

Input Parameters

processInstanceID	**String** The identifier of the process instance for which you want to add an error message. You assigned the process instance an identifier when you created the process instance using the pub.monitor.integrationProcessLogging:createProcessInstance service. A process instance ID is 1 to 32 alphanumeric characters.
serviceName	**String** Optional. The name of the service in which the error occurred. This is typically the fully qualified name of the IS service that caused the process error. The value you specify for *serviceName* can be 1 to 512 characters and can include characters that are valid in IS service names. See *webMethods Service Development Help* for information about allowed characters.
errorMsg	**String** The text of the error message that describes the error message. The value you specify for *errorMsg* can be 1 to 1024 characters.
errorStackTrace	**String** Optional. The stack trace associated with the error. The value you specify for *errorMsg* can be 1 to 2000 characters.

Output Parameters

result	**String** The outcome of adding the error message to the process instance. If the service successfully added the error message, *result* is "success." If the service encountered an error attempting to add the error message, *result* contains the exception.

Usage Notes

- You can view the error messages you add with this service using Monitor by viewing the Process Instance Detail page.
- If the service encounters an error, it logs an error to the PRA_ERROR table. You can view the error using Monitor with the Process Instance Detail page.

pub.monitor.integrationProcessLogging:createProcessInstance

Creates a new process instance of a specified integration process.

Input Parameters

processKey **String** The internal identifier (that is, process key) of the integration process for which you want to create an instance. The process key that you specify must already be defined in the Process Audit Log. You defined the process key and associated it with an integration process when you added the process definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createProcessMetadata service.

A process key is 1 to 64 alphanumeric characters.

processInstanceID **String** The process instance ID you want to assign to the new process instance you are creating. You can create your own process instance ID or use the pub.monitor.integrationProcessLogging:generateProcessInstanceID service to generate one.

If you create your own process instance ID, the ID must be 1 to 32 characters.

Output Parameters

result **String** The outcome of creating the process instance. If the service successfully created the process instance, *result* is "success." If the service encountered an error attempting to create the process instance, *result* contains the exception.

Usage Notes

- The new process instance is added to the Process Audit Log.
- After initially creating a new process instance, the process instance status is Started.
- If this service encounters errors, you can review the errors on the My webMethods pages you use to monitor services.

pub.monitor.integrationProcessLogging:createProcessMetadata

Adds a definition for an integration process (known as process metadata) to the Process Audit Log database.

The process definition is comparable to an entry in the Process Audit Log for a process model that was created with Designer. In the same way you need to create a process model before you can execute process instances that use the model, you need to define the integration process definition before you can create instances of the integration process.

When you add the definition, as part of the process metadata, the service includes a blank image because there is no model diagram for the process. You cannot specify an image to associate with the process definition.

Input Parameters

processKey **String** An identifier (that is, process key) that you assign to the integration process, for example, `OrderProcess`. The value you specify for *processKey* must be 1 to 64 alphanumeric characters. You use this identifier in subsequent services when you want to reference this process definition. It is an internal identifier only; Monitor does not display this value.

processLabel **String** A process name that you assign to the integration process. The value you specify for *processLabel* must be 1 to 1024 alphanumeric characters. Monitor displays this value when listing information about the integration process.

Output Parameters

result **String** The outcome of adding the process definition. If the service successfully added the process definition, *result* is "success." If the service encountered an error attempting to add the process definition, *result* contains the exception.

Usage Notes

If this service encounters errors, you can review the errors on the My webMethods pages you use to monitor services.

pub.monitor.integrationProcessLogging:createStepInstance

Creates a new step instance of the specified process step.

Input Parameters

processKey **String** The internal identifier (that is, process key) of the integration process definition that the step is a part of. You defined the process key when you added the process definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createProcessMetadata service.

A process key is 1 to 64 alphanumeric characters.

processInstanceID **String** The identifier of the process instance that the step instance is a part of. You assigned the process instance an identifier when you created the process instance using the pub.monitor.integrationProcessLogging:createProcessInstance service.

A process instance ID is 1 to 32 alphanumeric characters.

stepID **String** The internal identifier of the process step for which you want to create an instance. The step identifier that you specify must already be defined in the Process Audit Log. You defined the step identifier and associated it with a process step when you added the step definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createStepMetadata service.

A step identifier is 1 to 128 alphanumeric characters.

stepIteration **Number** Optional. The iteration of the step you are creating. Specify an iteration count if the step is executed multiple times within a single process instance, for example, if the step is within a loop or is an error handling step that can be executed more than one time. By default, the service uses a value of 1 for *stepIteration* .

Output Parameters

result **String** The outcome of creating the step instance. If the service successfully created the step instance, *result* is "success." If the service encountered an error attempting to create the step instance, *result* contains the exception.

Usage Notes

- The new step instance is added to the Process Audit Log.
- After initially creating a new step instance, the step instance status is Started.
- If this service encounters errors, you can review the errors on the My webMethods pages you use to monitor services.

pub.monitor.integrationProcessLogging:createStepMetadata

Adds a definition for a step within an integration process (known as step metadata) to the Process Audit Log database.

The step definition is comparable to an entry in the Process Audit log for a step in a process model that was created with Designer. In the same way you need to create steps within a process model before you can execute instances of those steps, you need to define the step definitions for steps in an integration process before you can create the step instances for the integration process.

When you add the step definition, as part of the step metadata, the service includes a blank image because there is no step icon for the process step. You cannot specify an image to associate with the step definition.

Input Parameters

processKey	**String** The internal identifier (that is, process key) of the integration process definition that the step you are defining is a part of. You defined the process key when you added the process definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createProcessMetadata service. A process key is 1 to 64 alphanumeric characters.
stepID	**String** An identifier that you assign to the process step. The value you specify for *stepID* must be 1 to 128 alphanumeric characters. You use this identifier in subsequent services when you want to reference this step definition. It is an internal identifier only; Monitor does not display this value.
stepLabel	**String** A step name that you assign to the process step. The value you specify for *stepLabel* must be 1 to 1024 alphanumeric characters. Monitor displays this value when listing information about the process step.

Output Parameters

result	**String** The outcome of adding the step definition. If the service successfully added the step definition, *result* is "success." If the service encountered an error attempting to add the step definition, *result* contains the exception.

Usage Note

If this service encounters errors, you can review the errors on the My webMethods pages you use to monitor services.

pub.monitor.integrationProcessLogging:generateProcessInstanceID

Generates a unique process instance ID.

Input Parameters

None.

Output Parameters

processInstanceID	**String** The process instance ID that the service generated.

Usage Notes

- This service does not save the generated process instance ID to the Process Audit Log database.
- Other services in the pub.monitor.integrationProcessLogging folder require a process instance ID as input. You can use this service to generate one rather than creating your own process instance ID.

pub.monitor.integrationProcessLogging:updateProcessInstanceStatus

Updates the process status of an existing process instance.

Input Parameters

processKey	**String** The internal identifier (that is, process key) of the integration process definition that the process instance uses. You defined the process key when you added the

process definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createProcessMetadata service.

A process key is 1 to 64 alphanumeric characters.

processInstanceID **String** The identifier of the process instance whose status you want to update. You assigned the process instance an identifier when you created the process instance using the pub.monitor.integrationProcessLogging:createProcessInstance service.

A process instance ID is 1 to 32 alphanumeric characters.

status **String** The status code number that represents the status you want to assign to the process instance. For a list of status codes you can use, see "Status Reference" on page 189.

Note: If you are setting the value of *status* using Designer, Designer displays the list of status values rather than listing the status code numbers.

Output Parameters

result **String** The outcome of updating the process status. If the service successfully updated the process status, *result* is "success." If the service encountered an error attempting to update the process status, *result* contains the exception.

Usage Notes

- If the service encounters an error, it logs an error to the PRA_ERROR table. You can view the error using Monitor with the Process Instance Detail page for the process instance.
- You can specify any valid status regardless of the current status. There are no restrictions for what statuses you can use based on the current status.

pub.monitor.integrationProcessLogging:updateStepInstanceStatus

Updates the step status of an existing step instance.

Input Parameters

processKey **String** The internal identifier (that is, process key) of the integration process definition that the step is a part of. You defined the process key when you added the

process definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createProcessMetadata service.

A process key is 1 to 64 alphanumeric characters.

processInstanceID **String** The identifier of the process instance that the step instance is a part of. You assigned the process instance an identifier when you created the process instance using the pub.monitor.integrationProcessLogging:createProcessInstance service.

A process instance ID is 1 to 32 alphanumeric characters.

stepID **String** The internal identifier of the process step whose status you want to update. The step identifier that you specify must already be defined in the Process Audit Log. You defined the step identifier and associated it with a process step when you added the step definition to the Process Audit Log using the pub.monitor.integrationProcessLogging:createStepMetadata service.

A step identifier is 1 to 128 alphanumeric characters.

stepIteration **Number** Optional. The iteration of the step. Specify an iteration count if the step is executed multiple times within a single process instance, for example, if the step is within a loop or is an error handling step that can be executed more than one time. By default, the service uses a value of 1 for *stepIteration* .

status **String** The status code number that represents the status you want to assign to the step instance. "Status Reference" on page 189.

Note: If you are setting the value of *status* using Designer, Designer displays the list of status values rather than listing the status code numbers.

Output Parameters

result **String** The outcome of updating the step status. If the service successfully updated the step status, *result* is "success". If the service encountered an error attempting to update the step status, *result* contains the exception.

Usage Notes

- If the service encounters an error, it logs an error to the PRA_ERROR table. You can view the error using Monitor via the Step Instance Detail page for the step instance.

- You can specify any valid status regardless of the current status. There are no restrictions for what statuses you can use based on the current step status.

5 pub.monitor.process.actions Folder

- Summary of Elements in This Folder 54
- pub.monitor.process.actions:CustomImplOutputDoc 54

Summary of Elements in This Folder

Service and Description
pub.monitor.process.actions:CustomImplOutputDoc A document type that describes the required output of a user-defined service used to determine the process instances to be targeted by a business rule process action.

pub.monitor.process.actions:CustomImplOutputDoc

A document type that describes the required output of a user-defined service used to determine the process instances to be targeted by a business rule process action.

Input Parameters

instances **Document list** A list of instances that are to be targeted by the business rule process action. For example, if you have 10 instances to process, there will be 10 items in this list.

instanceID **String** Specific process iteration that you want to target with the invoked business rule process action.

Output Parameters

None.

6 pub.monitor.process.instance Folder

- Summary of Elements in This Folder 56
- pub.monitor.process.instance:generateInstanceImage 58
- pub.monitor.process.instance:getDocumentNames 60
- pub.monitor.process.instance:getFieldNames 60
- pub.monitor.process.instance:getInstance 61
- pub.monitor.process.instance:getInstanceActivityLogs 63
- pub.monitor.process.instance:getInstanceControl 64
- pub.monitor.process.instance:getInstanceConversationID 66
- pub.monitor.process.instance:getInstanceCustomData 66
- pub.monitor.process.instance:getInstanceErrors 69
- pub.monitor.process.instance:getInstanceList 70
- pub.monitor.process.instance:getInstanceListByQueryName 74
- pub.monitor.process.instance:getInstanceListCustomData 75
- pub.monitor.process.instance:getInstanceListCustomData Set 79
- pub.monitor.process.instance:getInstanceListWithFilter 84
- pub.monitor.process.instance:getInstanceModelInfo 90
- pub.monitor.process.instance:getInstanceSteps 90
- pub.monitor.process.instance:getInstanceTransitions 92
- pub.monitor.process.instance:getPagedInstanceList 93
- pub.monitor.process.instance:getProcessList 97
- pub.monitor.process.instance:getRecentlyCompleted 98
- pub.monitor.process.instance:getRecentlyCreated 99
- pub.monitor.process.instance:getRecentlyFailed 100
- pub.monitor.process.instance:getRecentlySuspended 101

Summary of Elements in This Folder

Service and Description
pub.monitor.process.instance:generateInstanceImage Generates the run-time image for a specified process instance in JPG or SVG format.
pub.monitor.process.instance:getDocumentNames Retrieves the names of documents for which a process instance logged user-specified fields. Process instances log user-defined fields to the logging database.
pub.monitor.process.instance:getFieldNames Retrieves the names of user-specified document fields that a process instance logged to the logging database.
pub.monitor.process.instance:getInstance Retrieves details about a process instance from the logging database.
pub.monitor.process.instance:getInstanceActivityLogs Retrieves the user-defined messages that were the logged for the specified process instance.
pub.monitor.process.instance:getInstanceControl Retrieves the control actions (suspend, resume, resubmit, or stop) that have been executed against a specified webMethods-executed process instance.
pub.monitor.process.instance:getInstanceConversationID Retrieves the conversation ID for a process instance that was triggered by a Trading Networks document.
pub.monitor.process.instance:getInstanceCustomData Retrieves the user-specified document field values that a process instance logged to the logging database.
pub.monitor.process.instance:getInstanceErrors Retrieves the errors that were logged to the logging database for a process instance.

Service and Description
pub.monitor.process.instance:getInstanceList Retrieves process instances that meet specified criteria.
pub.monitor.process.instance:getInstanceListByQueryName Retrieves a list of process instances that meet the criteria specified by a saved query. This service can use the saved queries created in webMethods Monitor version 6.5.x or earlier and have been migrated to the current version. This service is deprecated.
pub.monitor.process.instance:getInstanceListWithFilter Retrieves process instances that meet specified criteria. In addition to the criteria that you can specify with the pub.monitor.process.instance:getInstanceList service, with this service you can also use filters to limit the returned instances based on pipeline values logged at run time and filter fields assigned to the process model on which the instance is based.
pub.monitor.process.instance:getInstanceListCustomData Retrieves process instances that meet specified criteria, including specifying the value of a single logged field, which instructs the service to return all instances where the value you specify was logged for a specified custom logged field.
pub.monitor.process.instance:getInstanceListCustomData Set Retrieves process instances that meet specified criteria, including specifying a set of values of multiple custom logged fields, which instructs the service to return all process instances where the values you specify were logged for the specified custom logged fields.
pub.monitor.process.instance:getInstanceSteps Retrieves information about all steps that were executed within a process instance.
pub.monitor.process.instance:getInstanceTransitions Retrieves information from the logging database about the transitions that were logged for the most recent iteration of a process instance.
pub.monitor.process.instance:getProcessList Retrieves the model names of all process instances that have run, successfully or otherwise, and for which logging data exists in the logging database.

Service and Description
pub.monitor.process.instance:getRecentlyCompleted Retrieves information for the most recently completed process instances. The service returns information for up to twenty process instances that completed in the last two weeks.
pub.monitor.process.instance:getRecentlyCreated Retrieves information for the most recently created process instances. The service returns information for up to twenty process instances that were created in the last two weeks.
pub.monitor.process.instance:getRecentlyFailed Retrieves information for the most recently failed process instances. The service returns information for up to twenty process instances that failed in the last two weeks.
pub.monitor.process.instance:getRecentlySuspended Retrieves information for the most recently suspended process instances. The service returns information for up to twenty process instances that were suspended in the last two weeks.

pub.monitor.process.instance:generateInstanceImage

Generates the run-time image for a specified process instance in JPG or SVG format. Run-time images show the process model image with icons that indicate each step's status.

This service returns the location of the generated image. To view the JPG image, open the image from the specified location. To view the SVG image, open an Internet browser and type the Integration Server host and port along with the location returned by the service as the URL. For example, if Integration Server is running on localhost:5555 and the service returned the location /WmMonitor/images/processes/process_image1352194153648935.svg, type the URL http://localhost:5555/WmMonitor/images/processes/process_image1352194153648935.svg.

Input Parameters

instanceID **String** Instance ID of the process instance for which to generate the image. Specify the complete, exact ID.

Note: Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2,

or SQL server) handles the queries that Monitor issues to obtain data.

nodeID **String** Optional. Node ID of a step within the process that represents an inline process (or subprocess) within the process; that is, a set of steps that have been collapsed into a single step. When you specify *nodeID* , the service generates the image for the subprocess only, not for the entire process. You can retrieve the node IDs for the subprocess within a process by executing the service and using the value returned in the *modelSteps/INLINESTEPID* field.

imageType **String** Requested image type, either JPG or SVG.

Output Parameters

imageData **Document** The generated image. The returned *imageData* document contains the following fields:

- *imageURL* **String** URL to the generated image file on the file system.
- *width* **String** Width of the image.
- *height* **String** Height of the image.
- *type* **String** Image type.
  - `1` Image is in JPG format.
  - `2` Image is in SVG format.
- *imageMap* **Document List** A list of the steps within the process instance. The following fields are returned for each step:
  - *key* **String** If the step represents a referenced process, *key* is the model ID of the model for the referenced process.
  - *stepid* **String** The step ID of the step.
  - *x* **String** The X coordinate of the top, left corner for where the icon for this step is placed within the image for the process instance.
  - *y* **String** The Y coordinate of the top, left corner for where the icon for this step is placed with in the image for the process instance.
  - *x2* **String** The X coordinate of the bottom, right corner for where the icon for this step is placed with in the image for the process instance.
  - *y2* **String** The Y coordinate of the bottom, right corner for where the icon for this step is placed with in the image for the process instance.

imageError **String** Errors that occurred during generation of the image.

pub.monitor.process.instance:getDocumentNames

Retrieves the names of documents for which a process instance logged user-specified fields. Process instances log user-defined fields to the logging database.

Note: Whether the search is case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID for the process instance for which to retrieve document names. Specify the complete, exact ID.

Output Parameters

documentNames **Document List** The retrieved document names. For each document, the following field is returned.

documentName **String** The name of a document for which the process instance logged user-specified fields.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getFieldNames

Retrieves the names of user-specified document fields that a process instance logged to the logging database.

Note: Whether the search is case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID for the process instance for which to retrieve the names of user-specified document fields. Specify the complete, exact ID.

Output Parameters

fieldNames **Document List** The retrieved field names. For each field name, the following field is returned.

fieldName **String** Name of the field name that was logged.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstance

Retrieves details about a process instance from the logging database.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID for the process instance for which you want to retrieve details. Specify the complete, exact ID.

Output Parameters

data **Document** The details for the process instance. The returned *data* parameter contains these fields:

- *instanceID* **String** The instance ID for the process instance that you specified in the input parameter.
- *customID* **String** An ID that the process instance logged during execution using the pub.prt.log:logCustomId service. If this service was not used to log a custom ID, *customID* and *instanceID* are the same value.

- *modelID* **String** The unique ID for the process model.
- *modelName* **String** The name of the model.
- *modelVersion* **String** The version of the model.
- *instanceIteration* **Number** Most recent instance iteration count for the process instance.

  The first time a process instance is executed, the iteration count is 1. Each time a process instance is resubmitted, the iteration count is incremented.
- *parentInstanceID* **String** Instance ID for parent process instance, if any. If the process instance does not have a parent process, *parentInstanceID* is null.
- *parentInstanceIteration* **Number** Iteration count of the parent process instance, if any. If the process instance does not have a parent process, *parentInstanceIteration* is 0.
- *statusDecode* **String** A keyword value for the most recent status of the process instance. For the list of keyword values, for example "Started" or "Completed," see "Status Reference" on page 189.
- *status* **String** A numerical value that represents the status of the process instance. For a list of values you can specify, see "Status Reference" on page 189.
- *rootContextID* **String** Context ID of the root process instance that called this process instance.
- *parentContextID* **String** The parent context ID is an internal identifier that Integration Server uses.
- *timeStamp* **String** Time data was last logged for the process instance in the string format YYYY-MM-DD hh:mm:ss.SSS zzz.
- *errorMessage* **String** Error that occurred during the execution of this service if this service encountered an error.

contextID **String** Context IDs that enable you retrieve errors associated with this process instance.

processControl **String** Whether you have the functional privileges to suspend/resume, stop, *and* resubmit the process.

- `true` You have all three functional privileges for this process instance.

- `false` You do not have all three functional privileges to control this process instance.

pub.monitor.process.instance:getInstanceActivityLogs

Retrieves the user-defined messages that were the logged for the specified process instance.

The process instance logged these messages using the pub.prt.log:logActivityMessages service.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID for the process instance for which to retrieve user-defined messages. Specify the complete, exact ID.

stepNames **HashMap** Optional. A hash map that provides the names of the steps within the process instance. The hash map is a key value pair, where the key is the step ID and the value is the name of the step. To obtain a hash map, use the pub.monitor.process.instance:getInstanceSteps service.

stepCidList **HashMap** Optional. A hash map that provides the context IDs of the services that were executed within the process instance. The hash map is a key value pair, where the key is the step ID of a step that executed a service and the value is the context ID of the service that was executed. To obtain a hash map, use the pub.monitor.process.instance:getInstanceSteps service.

cidList **HashSet** Optional. Context IDs for the steps that the process ran. You can retrieve the context IDs using the pub.monitor.process.instance:getInstanceSteps service.

Output Parameters

activityLog **Document List** A list of logged activity messages. For each activity message, the following fields are returned.

- *PARENTCONTEXTID* **String** The parent context ID is an internal identifier that Integration Server uses.

- *CONTEXTID* **String** The context ID is an internal identifier that Integration Server uses.
- *PROCESSTEPCONTEXT* **String** The internal identifier (i.e., the step ID) of the step that logged activity message.
- *ENTRYTYPE* **String** The type of user-defined message, that is error, warning, or message.
- *FULLMESSAGE* **String** The text of the user-defined message. It contains up to 1024 characters.
- *BRIEFMESSAGE* **String** A brief version of the text of the message that contains only up to 240 characters.
- *B2BUSER* **String** The Integration Server user name of the user that invoked the service that logged the user-defined message.
- *SERVERID* **String** ID of the server where the service that logged the user-defined message ran.
- *AUDITTIMESTAMP* **Number** Time the message was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the message was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstanceControl

Retrieves the control actions (suspend, resume, resubmit, or stop) that have been executed against a specified webMethods-executed process instance.

Note: This service only reports on control actions that you can perform using Monitor. Using Monitor, you can only perform control actions against webMethods-executed process instances, which are the only type of process instances that the Process Engine executes. Because the Process Engine does not execute externally executed and integration processes, you cannot use Monitor to perform control actions against them, and consequently, this service does not return information about them.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID of the webMethods-executed process instance for which to retrieve control actions. Specify the complete, exact ID.

stepNames **HashMap** Optional. A hash map that provides the names of the steps within the process instance. The hash map is a key value pair, where the key is the step ID and the value is the name of the step. To obtain a hash map you can use as input, use the pub.monitor.process.instance:getInstanceSteps service.

Output Parameters

instanceControl **Document List** The list of control actions that have been executed. For each control action that was taken against a step in the process instance, the following fields are returned:

- *ACTION* **String** Type of action. The service returns a numerical value that represents the type of action as follows.
  - `1` Service Resubmit
  - `2` Document Resubmit
  - `3` Process Resubmit
  - `4` Process Suspend
  - `5` Process Resume
  - `6` Process Stop
- *ACTIONDECODE* **String** The keyword value that represents the action. The keyword is: Service Resubmit, Document Resubmit, Process Resubmit, Process Suspend, Process Resume, or Process Stop.
- *USERNAME* **String** User who initiated the control action.
- *SERVERID* **String** Server where the action was initiated.
- *AUDITTIMESTRING* **String** Time the action was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

- *STEPID* **String** The ID of the step against which the control action was executed. The step ID uniquely identifies a specific step within a specific process instance.
- *INSTANCEITERATION* **Number** The process instance iteration that contains the step against which the control action was executed.
- *STEPITERATION* **Number** The iteration of step against which the control action was executed.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstanceConversationID

Retrieves the conversation ID for a process instance that was triggered by a Trading Networks document.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID for the process instance for which to retrieve the conversation ID. Specify the complete, exact ID.

Output Parameters

conversationID **String** The retrieved conversation ID.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstanceCustomData

Retrieves the user-specified document field values that a process instance logged to the logging database.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID of the process instance for which to retrieve the user-specified document field values. Specify the complete, exact ID.

recent **String** The iterations of the process instance for which to retrieve the user-specified document field values. The code value indicates whether to retrieve document field values logged for all process instances or the most recent process instance iterations:

- `0` Default. Retrieve field values only for most recent process instance iteration.
- `1` Retrieve field values logged for all process instance iterations.

documentName **String** Optional. If values were logged for more than one document, complete name of the document whose logged values you want retrieved. By default, the service returns logged values for all documents. You can retrieve document names using the pub.monitor.process.instance:getDocumentNames service.

Note: Specify the name of the document that was assigned through Designer, not the document type name (for example, not the fully qualified name of an IS document type).

fieldName **String** Optional. Complete name of the field whose value you want retrieved. By default, the service returns logged values for all fields. You can retrieve field names using pub.monitor.process.instance:getFieldNames service.

sortBy **String** Optional. Value to use to sort the returned list of document field value. This parameter works with the *sortOrder* parameter.

- `STEPLABEL` Default. Names of the steps for which the field value was logged.
- `INSTANCEITERATION` Process instance iteration when the field value was logged.
- `STEPITERATION` Step iteration when the field value was logged.
- `DOCUMENTNAME` Name of the document for which values were logged.

- `FIELDNAME` Name of the document field for which a value was logged.
- `STRINGVALUE` Values of String type document fields.
- `NUMBERVALUE` Values of Number type document fields.
- `DATEVALUE` Values of Date type document fields.

sortOrder **String** Optional. Whether to sort the returned list of documents in ascending or descending order. The documents are sorted by the field identified by the *sortBy* parameter.

- `0` Default. Sort in ascending order.
- `1` Sort in descending order.

Output Parameters

customData **Document List** The list of returned document field values. For each document field value, the following fields are returned:

- *STEPLABEL* **String** Step name of the step that logged the field value.
- *INSTANCEITERATION* **Number** Process instance iteration when the field value was logged.
- *STEPITERATION* **Number** Step iteration when the field value was logged.
- *DOCUMENTNAME* **String** Name of the document that contains the field value that was logged.
- *FIELDNAME* **String** Name of the field within the document that was logged.
- *STRINGVALUE* **String** The value of the field if the logged field value is a string.
- *NUMBERVALUE* **String** The value of the field if the logged field value is a number; otherwise the service returns 0.0 in this field.
- *DATEVALUE* **String** The value of the field if the logged field value is date.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstanceErrors

Retrieves the errors that were logged to the logging database for a process instance.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID of the webMethods-executed process instance for which to retrieve control actions. Specify the complete, exact ID.

Output Parameters

instanceErrors **Document List** The retrieved errors. The following fields are returned for each error.

- *ROOTCONTEXTID* **String** Root context ID of the process instance.
- *PARENTCONTEXTID* **String** Parent context ID of the process instance.
- *CONTEXTID* **String** Context ID of the process instance.
- *SERVERID* **String** ID of the server where the error occurred.
- *SERVICENAME* **String** Name of the service that logged the error.
- *ERRORMSG* **String** The logged error.
- *ERRORSTACKTRACE* **String** Any stack trace logged for this error.
- *AUDITTIMESTAMP* **Number** Time the action was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the action was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstanceList

Retrieves process instances that meet specified criteria.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

modelID **String** Optional. Complete model ID used for the process instances you want retrieved. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.

modelIDSet **String List** Optional. A set of complete model IDs used for the process instances you want retrieved. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.

modelNameInput **String** Optional. Complete or partial model name used for the process instances you want retrieved. You can retrieve model names by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEYDECODE* output parameter. When you specify *modelNameInput* , do not use the *modelID* parameter.

instanceID **String** Optional. Complete or partial instance ID for the process instances you want retrieved.

status **String** Optional. The status of the process instances you want retrieved. The status you specify is matched against the most recently logged status. Use *status* if you want the list to contain all process instances of a single specified status. Specify the numerical value that represents the status. For a list of values you can specify, see "Status Reference" on page 189.

Note: If you do not specify *status* or *statusSet* , the service returns process instances of all statuses.

statusSet **String List** Optional. A set of statuses for the process instances you want retrieved. Use *statusSet* when you want the list to contain process instances of more than one specified status. For each status that you specify in *statusSet* , specify the numerical value that represents the status, as described above for the *status* parameter.

parentInstanceID **String** Optional. Complete or partial instance ID for the parent process of the process instances that you want retrieved.

customID **String** Optional. The full, user-defined ID for the process instances that you want retrieved. User-defined IDs are assigned by executing the pub.prt.log:logCustomID service.

dateCreated **String** Optional. A date range for the process instances you want retrieved. The date range identifies the date of the most recent log entry for the process instances. If you use this parameter, do not use the *fromDate* or *toDate* parameter.

- `Today` Today
- `Yesterday` Yesterday
- `In the last 7 days` Within the last 7 days, including today.
- `Last week` Any day in the previous calendar last week (A week is Sunday through Saturday.)
- `This week` Any day in the current calendar week. (A week is Sunday through Saturday.)
- `Last month` Any day in the previous calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

fromDate **String** Optional. The start date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or after this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *toDate* to specify the end date; do not use the *dateCreated* parameter when you use the *fromDate* and *toDate* parameters.

toDate **String** Optional. The end date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or before this date. Use the format *YYYY-MM-DD HH:MM:SS* .

stepID **String** Optional. The full ID of a step that was executed in the process instances you want retrieved. You can retrieve step IDs using the service. When you use this *stepID* parameter, you must also specify:

- The *modelID* parameter to identify the model in which to search for the specified step.
- The *user* or *role* parameters.

user **String** Optional. To retrieve information about tasks, user on which to match (that is, user that performed a task). When you use this *user* parameter, you must also specify the *modelID* parameter.

role **String** Optional. To retrieve information about tasks, role on which to match (that is, role that performed a task). When you use this *role* parameter, you must also specify the *modelID* parameter.

maxRows **String** Optional. Maximum number of process instances to return, starting with those most recently logged. By default, the service gets all process instances.

isAnd **String** Optional. Whether the service is to use an `AND` or an `OR` condition for the criteria specified in the input parameters.

- `true` Default. Use an `AND` condition. The service returns process instances that match all the criteria you specify.
- `false` Use an `OR` condition. The service returns process instances that match any of the criteria you specify.

sortColumn **String** Optional. How to sort the returned list of process instances. By default, the service sorts the returned data using `AUDITTIMESTAMP`. This parameter works with the *sortAscending* parameter. Sort by:

- `PROCESSLABEL` Model name.
- `INSTANCEID` Process instance ID.
- `PARENTINSTANCEID` Parent process instance ID, if any.
- `STATUS` Most recent status; the sort order is based on the numerical values associated with statuses. For a list of the status values, see "Status Reference" on page 189.
- `AUDITTIMESTAMP` The last time data was logged for the process instance.

sortAscending **String** Optional. Whether to sort the returned list of process instances in ascending or descending order. The entries are sorted by the field identified by the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

Output Parameters

instances **Document List** List of process instances that match the specified criteria. For each process instance, the following fields are returned:

- *ROOTCONTEXTID* **String** Root context ID of the process instance.
- *PARENTCONTEXTID* **String** Parent context ID of the process instance.
- *CONTEXTID* **String** Context ID of the process instance.
- *AUDITTIMESTAMP* **Number** Time data was last logged for the process instance, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time data was last logged for the process instance in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *PARENTINSTANCEID* **String** Instance ID of the process instance's parent instance, if any.
- *PARENTINSTANCEITERATION* **Number** Instance iteration of the process instance's parent instance, if any.
- *INSTANCEID* **String** Instance ID of the process instance.
- *INSTANCEITERATION* **Number** Instance iteration of the process instance.
- *STATUS* **String** Status of the process instance. The service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value for this process instance. The service returns the keyword value that represents the status. For the list of keyword values, for

example, "Started" or "Completed", see "Status Reference" on page 189.

- *PROCESSKEY* **String** Unique model ID of the process instance.
- *PROCESSKEYDECODE* **String** Process name of the process instance.
- *CUSTOMID* **String** The user-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.
- *PROCESSLABEL* **String** Process name of the process instance.

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

Usage Notes

If you want to improve performance and limit the number of results returned, use the related service, pub.monitor.process.instance:getPagedInstanceList. This service limits the number of results returned in the query by the *fromIndex* and *toIndex* .

pub.monitor.process.instance:getInstanceListByQueryName

Deprecated. Retrieves a list of process instances that meet the criteria specified by a saved query. This service can use the saved queries created in webMethods Monitor version 6.5.x or earlier and have been migrated to the current version. This service is deprecated.

Input Parameters

loadQuery **String** Name of the saved query that specifies the criteria for retrieving the process instances.

Output Parameters

instances **Document List** List of process instances that match the criteria specified in the query.

pub.monitor.process.instance:getInstanceListCustomData

Retrieves process instances that meet specified criteria, including specifying the value of a single logged field, which instructs the service to return all instances where the value you specify was logged for a specified custom logged field.

Note: Whether the search is case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

modelID **String** Optional. Complete model ID used for the process instances you want retrieved. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.

modelIDSet **String List** Optional. A set of complete model IDs used for the process instances you want retrieved. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.

modelNameInput **String** Optional. Complete or partial model name used for the process instances you want retrieved. You can retrieve model names by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEYDECODE* output parameter. When you specify *modelNameInput* , do not use the *modelID* parameter.

instanceID **String** Optional. Complete or partial instance ID for the process instances you want retrieved.

status **String** Optional. The status of the process instances you want retrieved. The status you specify is matched against the most recently logged status. Use *status* if you want the list to contain all process instances of a single specified status. Specify the numerical value that represents the status. For a list of values you can specify, see "Status Reference" on page 189.

Note: If you do not specify *status* or *statusSet* , the service returns process instances of all statuses.

statusSet **String List** Optional. A set of statuses for the of the process instances you want retrieved. Use *statusSet* when you want the list to contain process instances of more than one specified status. For each status that you specify in *statusSet* , specify the numerical value that represents the status, as described above for the *status* parameter.

parentInstanceID **String** Optional. Complete or partial instance ID for the parent process of the process instances that you want retrieved.

dateCreated **String** Optional. A date range for the process instances you want to retrieve. The date range identifies the date of the most recent log entry for the process instances. If you use this parameter, do not use the *fromDate* or *toDate* parameter. A week is Sunday through Saturday.

- `Today` Today.
- `Yesterday` Yesterday.
- `In the last 7 days` Within the last 7 days, including the current date.
- `Last week` Any day in the previous calendar last week.
- `This week` Any day in the current calendar week.
- `Last month` Any day in the previous calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

fromDate **String** Optional. The start date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or after this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *toDate* to specify the end date; do not use the *dateCreated* parameter when you use the *fromDate* and *toDate* parameters.

toDate **String** Optional. The end date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or before this date. Use the format *YYYY-MM-DD HH:MM:SS* .

stepID **String** Optional. The full ID of a step that was executed in the process instances you want retrieved. You can retrieve step IDs using the service. When you use this *stepID* parameter, you must also specify:

- The *modelID* parameter to identify the model in which to search for the specified step.
- The *user* or *role* parameters.

user **String** Optional. To retrieve information about tasks, user on which to match (that is, user that performed a task). When you use this *user* parameter, you must also specify the *modelID* parameter.

role **String** Optional. To retrieve information about tasks, role on which to match (that is, role that performed a task). When you use this *role* parameter, you must also specify the *modelID* parameter.

documentName **String** The complete name of the document that contains the custom field that you want to use to search for process instances. Specify the name as it was defined in Designer.

fieldName **String** Complete name of the logged field that you want to use to search for process instances. The logged field should be a field within the document identified by *documentName* .

fieldType **String** The data type of *fieldname* . Specify either `String` or `Number`.

comparator **String** A comparator indicates how the service should compare the value you specify in *value* to the actual value logged for the custom field specified by *fieldName* . Specify one of the following: = , Contains, Not Contains, != , < , >, <= , >=

value **String** The value that you want the service to use to compare with the actual value stored for the custom field specified by *fieldName* .

maxRows **String** Optional. Maximum number of process instances to return, starting with those most recently logged. By default, the service gets all process instances.

isAnd **String** Optional. Whether to use an `AND` or an `OR` condition for the criteria specified in the input parameters.

- `true` Default. Use an `AND` condition. The service returns process instances that match all the criteria you specify.
- `false` Use an `Or` condition. The service returns process instances that match any of the criteria you specify.

sortColumn **String** Optional. How to sort the returned list of process instances. By default, the service sorts the returned data using `AUDITTIMESTAMP`. This parameter works with the *sortAscending* parameter.

- `PROCESSLABEL` Model name.
- `INSTANCEID` Process instance ID.
- `PARENTINSTANCEID` Parent process instance ID, if any.
- `STATUS` Most recent status; the sort order is based on the numerical values associated with statuses. For a list of the status values, see "Status Reference" on page 189.
- `AUDITTIMESTAMP` The last time data was logged for the process instance.

sortAscending **String** Optional. Whether to sort the returned list of process# instances in ascending or descending order. The entries are sorted# by the field identified by the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

Output Parameters

instances **Document List** List of process instances that match the specified criteria. For each process instance, the following fields are returned:

- *ROOTCONTEXTID* **String** Root context ID of the process instance.
- *PARENTCONTEXTID* **String** Parent context ID of the process instance.
- *CONTEXTID* **String** Context ID of the process instance.
- *AUDITTIMESTAMP* **Number** Time data was last logged for the process instance, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time data was last logged for the process instance in string format, *YYYY-MM-DD hh:mm:ss.SSS zzz*, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *PARENTINSTANCEID* **String** Instance ID of the process instance's parent instance, if any.

- *PARENTINSTANCEITERATION* **Number** Instance iteration of the process instance's parent instance, if any.
- *INSTANCEID* **String** Instance ID of the process instance.
- *INSTANCEITERATION* **Number** Instance iteration of the process instance.
- *STATUS* **String** Status of the process instance. The service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value for this process instance. The service returns the keyword value that represents the status. For the list of keyword values, for example, “Started” or “Completed,” see "Status Reference" on page 189.
- *PROCESSKEY* **String** Unique model ID of the process instance.
- *PROCESSKEYDECODE* **String** Process name of the process instance.
- *CUSTOMID* **String** The user-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.
- *PROCESSLABEL* **String** Process name of the process instance.

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstanceListCustomDataSet

Retrieves process instances that meet specified criteria, including specifying a set of values of multiple custom logged fields, which instructs the service to return all process instances where the values you specify were logged for the specified custom logged fields.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

modelID	**String** Optional. Complete model ID used for the process instances you want retrieved. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.
modelIDSet	**String List** Optional. A set of complete model IDs used for the process instances you want retrieved. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.
modelNameInput	**String** Optional. Complete or Partial model name used for the process instances you want retrieved. You can retrieve model names by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEYDECODE* output parameter. When you specify *modelNameInput* , do not use the *modelID* parameter.
instanceID	**String** Optional. Complete or partial instance ID for the process instances you want retrieved.
status	**String** Optional. The status of the process instances you want retrieved. The status you specify is matched against the most recently logged status. Use *status* if you want the list to contain all process instances of a single specified status. Specify the numerical value that represents the status. For a list of values you can specify, see "Status Reference" on page 189. **Note:** If you do not specify *status* or *statusSet* , the service returns process instances of all statuses.
statusSet	**String List** Optional. A set of statuses for the of the process instances you want retrieved. Use *statusSet* when you want the list to contain process instances of more than one specified status. For each status that you specify in *statusSet* , specify the numerical value that represents the status, as described above for the *status* parameter.

parentInstanceID **String** Optional. Complete or partial instance ID for the parent process of the process instances that you want retrieved.

customID **String** Optional. The full, user-defined ID for the process instances that you want retrieved. User-defined IDs are assigned by executing the pub.prt.log:logCustomID service.

dateCreated **String** Optional. A date range for the process instances you want retrieved. The date range identifies the date of the most recent log entry for the process instances. If you use this parameter, do not use the *fromDate* or *toDate* parameter. Set the value to the code to retrieve process instances that match the most recent entry:

- `Today` Today
- `Yesterday` Yesterday
- `In the last 7 days` Within the last 7 days, including current date.
- `Last week` Any day in the previous calendar last week. (A week is Sunday through Saturday.)
- `This week` Any day in the current calendar week (A week is Sunday through Saturday.)
- `Last month` Any day in the previous calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

fromDate **String** Optional. The start date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or after this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *toDate* to specify the end date; do not use the *dateCreated* parameter when you use the *fromDate* and *toDate* parameters.

toDate **String** Optional. The end date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or before this date. Use the format *YYYY-MM-DD HH:MM:SS* .

stepID **String** Optional. The full ID of a step that was executed in the process instances you want retrieved. You can retrieve step IDs using the service. When you use this *stepID* parameter, you must also specify:

- The *modelID* parameter to identify the model in which to search for the specified step.
- The *user* or *role* parameters.

user **String** Optional. To retrieve information about tasks, user on which to match (that is, user that performed a task). When you use this *user* parameter, you must also specify the *modelID* parameter.

role **String** Optional. To retrieve information about tasks, role on which to match (that is, role that performed a task). When you use this *role* parameter, you must also specify the *modelID* parameter.

sortAscending **String** Optional. Whether to sort the returned list of process instances in ascending or descending order. The entries are sorted by the field identified by the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

customTable **String Table** Optional. The custom fields and their values that you want to use to search for process instances. The service returns process instances that match all the custom field data you specify. For each custom field, specify the following fields.

- *documentName* **String** The complete name of the document that contains the custom field that you want to use to search for process instances. Specify the name as it was defined in Designer.
- *fieldName* **String** The complete name of the logged field that you want to use to search for process instances. The logged field should be a field within the document identified by *documentName* .
- *fieldType* **String** The data type of *fieldname* . Specify either `String` or `Number`.
- *comparator* **String** A comparator indicates how the service should compare the value you specify in *value* to the actual value logged for the custom field specified by *fieldName* . Specify one of the following: = , Contains, Not Contains, != , < , >, <= , >=
- *value* **String** The value that you want the service to use to compare with the actual value stored for the custom field specified by *fieldName* .

maxRows **String** Optional. Maximum number of process instances to return, starting with those most recently logged. By default, the service gets all process instances.

isAnd **String** Optional. Whether the service uses an `AND` or an `OR` condition for the criteria specified in the input parameters.

- `true` Default. Use an `AND` condition. The service returns process instances that match all the criteria you specify.
- `false` Use an `Or` condition. The service returns process instances that match any of the criteria you specify.

sortColumn **String** Optional. How to sort the returned list of process instances. By default, the service sorts the returned data using `AUDITTIMESTAMP`. This parameter works with the *sortAscending* parameter.

- `PROCESSLABEL` Model name.
- `INSTANCEID` Process instance ID.
- `PARENTINSTANCEID` Parent process instance ID, if any.
- `STATUS` Most recent status; the sort order is based on the numerical values associated with statuses. For a list of the status values, see "Status Reference" on page 189.
- `AUDITTIMESTAMP` The last time data was logged for the process instance.

Output Parameters

instances **Document List** List of process instances that match the specified criteria. For each process instance, the following fields are returned:

- *ROOTCONTEXTID* **String** Root context ID.
- *PARENTCONTEXTID* **String** Parent context ID.
- *CONTEXTID* **String** Context ID.
- *AUDITTIMESTAMP* **Number** Time data was last logged for the process instance, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time data was last logged for the process instance in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *PARENTINSTANCEID* **String** Instance ID of the parent instance, if any.

- *PARENTINSTANCEITERATION* **Number** Instance iteration of the parent instance, if any.
- *INSTANCEID* **String** Instance ID.
- *INSTANCEITERATION* **Number** Instance iteration.
- *STATUS* **String** Instance status. The service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value for this process instance. The service returns the keyword value that represents the status. For the list of keyword values, for example “Started” or “Completed,” see "Status Reference" on page 189.
- *PROCESSKEY* **String** Unique model ID of the process instance.
- *PROCESSKEYDECODE* **String** Process name of the process instance.
- *CUSTOMID* **String** User-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.
- *PROCESSLABEL* **String** Process name of the process instance.

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstanceListWithFilter

Retrieves process instances that meet specified criteria. In addition to the criteria that you can specify with the pub.monitor.process.instance:getInstanceList service, with this service you can also use filters to limit the returned instances based on pipeline values logged at run time and filter fields assigned to the process model on which the instance is based.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

modelID **String** Optional. Complete model ID used for the process instances you want retrieved. You can retrieve model IDs

	by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.
modelIDSet	**String List** Optional. A set of complete model IDs used for the process instances you want retrieved. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.
modelNameInput	**String** Optional. Complete or partial model name used for the process instances you want retrieved. You can retrieve model names by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEYDECODE* output parameter. When you specify *modelNameInput* , do not use the *modelID* parameter.
instanceID	**String** Optional. Complete or partial instance ID for the process instances you want retrieved.
status	**String** Optional. The status of the process instances you want retrieved. The status you specify is matched against the most recently logged status. Use *status* if you want the list to contain all process instances of a single specified status. Specify the numerical value that represents the status. For a list of values you can specify, see "Status Reference" on page 189. **Note:** If you do not specify *status* or *statusSet* , the service returns process instances of all statuses.
statusSet	**String List** Optional. A set of statuses for the process instances you want retrieved. Use *statusSet* when you want the list to contain process instances of more than one specified status. For each status that you specify in *statusSet* , specify the numerical value that represents the status, as described above for the *status* parameter.
parentInstanceID	**String** Optional. Complete or partial instance ID for the parent process of the process instances that you want retrieved.

customID **String** Optional. The full, user-defined ID for the process instances that you want retrieved. User-defined IDs are assigned by executing the pub.prt.log:logCustomID service.

dateCreated **String** Optional. A date range for the process instances you want retrieved. The date range identifies the date of the most recent log entry for the process instances. If you use this parameter, do not use the *fromDate* or *toDate* parameter.

- `Today` Today
- `Yesterday` Yesterday
- `In the last 7 days` Within the last 7 days, including the current date.
- `Last week` Any day in the previous calendar last week (A week is Sunday through Saturday.)
- `This week` Any day in the current calendar week (A week is Sunday through Saturday.)
- `Last month` Any day in the previous calendar month
- `This month` Any day in the current calendar month

fromDate **String** Optional. The start date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or after this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *toDate* to specify the end date; do not use the *dateCreated* parameter when you use the *fromDate* and *toDate* parameters.

toDate **String** Optional. The end date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or before this date. Use the format *YYYY-MM-DD HH:MM:SS* .

stepID **String** Optional. The full ID of a step that was executed in the process instances you want retrieved. You can retrieve step IDs using the service. When you use this *stepID* parameter, you must also specify:

- The *modelID* parameter to identify the model in which to search for the specified step.
- The *user* or *role* parameters.

user **String** Optional. To retrieve information about tasks, user on which to match (that is, user that performed a task). When you use this *user* parameter, you must also specify the *modelID* parameter.

role **String** Optional. To retrieve information about tasks, role on which to match (that is, role that performed a task). When you use this *role* parameter, you must also specify the *modelID* parameter.

maxRows **String** Optional. Maximum number of process instances to return, starting with those most recently logged. By default, the service gets all process instances.

isAnd **String** Optional. Whether to use an `AND` or an `OR` condition for the criteria specified in the input parameters.

- `true` Default. Use an `AND` condition. The service returns process instances that match all the criteria you specify.
- `false` Use an `OR` condition. The service returns process instances that match any of the criteria you specify.

sortColumn **String** Optional. How to sort the returned list of process instances. By default, the service sorts the returned data using `AUDITTIMESTAMP`. This parameter works with the *sortAscending* parameter.

- `PROCESSLABEL` Model name.
- `INSTANCEID` Process instance ID.
- `PARENTINSTANCEID` Parent process instance ID, if any.
- `STATUS` Most recent status; the sort order is based on the numerical values associated with statuses. For a list of the status values, see "Status Reference" on page 189.
- `AUDITTIMESTAMP` The last time data was logged for the process instance.

sortAscending **String** Optional. Whether to sort the returned list of process instances in ascending or descending order. The entries are sorted by the field identified by the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

criteriaFilter **String** Optional. A filter that limits the returned instances based on pipeline values that are logged at run time. Define the pipeline variables to log when creating the process model and defining the step properties in Software AG Designer. For more information, see the *Software AG Designer Online Help*.

Use the following format to specify the filter:
variable1 =value1 , value2 , ... , valueN

For example, if a pipeline variable named *city* is logged and you want to return only those instances for which the value

of the pipeline variable *city* is `Paris`, use the following filter: `city=Paris`

If you want to return instances for which *city* is `Paris` or `Madrid`, use the following filter: `city=Paris, Madrid`

To use multiple logged pipeline variables, specify `and` between the variables/values. For example, to return instances for which *city* is `Paris` and *manager* is `Mercier`, use the following filter: `city=Paris and manager=Mercier`

modelCriteriaFilter **String** Optional. A filter that limits the returned instances based on filter fields and values that are associated with the process model on which the instance is based. Assign filter fields and values to process models using the service.

Use the following format to specify the filter: *field1 =value1 , value2 , ... , valueN*

For example, if you have set a filter field named *countryCode* and want to return only those instances that use models for which *countryCode* is `es`, use the following filter: `countryCode=es`

If you want to return instances that use models for which the *countryCode* is `es` or `fr`, use the following filter: `countryCode=es, fr`

To use multiple filter fields, specify `and` between the fields/values. For example, to return instances of models for which the *countryCode* is `es` or `fr` and the *department* is `AP`, use the following filter: `countryCode=es, fr and department=AP`

isCustomIDExact **Boolean** Optional. If is *CustomIDExact* is set to true, it matches the exact *customID* in the query; otherwise it uses the LIKE condition.

Output Parameters

instances **Document List** List of process instances that match the specified criteria. For each process instance, the following fields are returned:

- *ROOTCONTEXTID* **String** Root context ID of the process instance.
- *PARENTCONTEXTID* **String** Parent context ID of the process instance.
- *CONTEXTID* **String** Context ID of the process instance.

- *AUDITTIMESTAMP* **Number** Time data was last logged for the process instance, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time data was last logged for the process instance in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *PARENTINSTANCEID* **String** Instance ID of the process instance's parent instance, if any.
- *PARENTINSTANCEITERATION* **Number** Instance iteration of the process instance's parent instance, if any.
- *INSTANCEID* **String** Instance ID of the process instance.
- *INSTANCEITERATION* **Number** Instance iteration of the process instance.
- *STATUS* **String** Status of the process instance. The service returns the numerical value that represents the status. For a list of the status values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value for this process instance. The service returns the keyword value that represents the status. For the list of keyword values, for example, “Started” or “Completed,” see "Status Reference" on page 189.
- *PROCESSKEY* **String** Unique model ID of the process instance.
- *PROCESSKEYDECODE* **String** Process name of the process instance.
- *CUSTOMID* **String** The user-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.
- *PROCESSLABEL* **String** Process name of the process instance.

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

Usage Notes

The following are services related to the *modelCriteriaFilter* input field:

- To assign a filter field that you can reference in *modelCriteriaFilter* , use the pub.monitor.process.model:setProcessFilter service.
- To retrieve a list of filter fields that are already set, use the pub.monitor.process.model:getProcessFilter service.
- To clear a filter field that you no longer need, use the pub.monitor.process.model:clearProcessFilter service.

pub.monitor.process.instance:getInstanceModelInfo

Retrieves information about the models that were invoked by a specified process instance.

Input Parameters

instanceID	**String** Process instance ID corresponding to the process model information you want to retrieve. Specify the complete, exact ID.

Output Parameters

modelID	**String** Complete model ID corresponding to the process instance. To retrieve model IDs, invoke the service, pub.monitor.process.instance:getProcessList, and use the value returned in the *processNames/PROCESSKEY* output parameter.
modelVersion	**String** The version of the process model to retrieve.
deployment Version	**String** The deployment version of the process model to retrieve.

pub.monitor.process.instance:getInstanceSteps

Retrieves information about all steps that were executed within a process instance.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID for the process instance for which to retrieve steps. Specify the complete, exact ID.

Output Parameters

instanceSteps **Document List** List of the steps that the specified process executed. For each step, the following fields are returned:

- *ROOTCONTEXTID* **String** Root context ID.
- *PARENTCONTEXTID* **String** Parent context ID.
- *CONTEXTID* **String** Context ID for the process step.
- *AUDITTIMESTAMP* **Number** Time the step was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the step was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *INSTANCEID* **String** Instance ID of the process instance in which the step was executed.
- *INSTANCEITERATION* **Number** Instance iteration for the process instance when the step was executed.
- *STEPID* **String** ID of the process step.
- *STEPITERATION* **Number** Iteration of the process step.
- *STATUS* **String** Status of the process step. The service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.

- *USERNAME* **String** If the step is a Task step, the user name associated with the task.
- *ROLENAME* **String** If the step is a Task step, the role name associated with the task.
- *SYSTEM* **String** System identified with this step. The service returns a numerical value to represent the system, as follows:
  - 1Integration Server
  - 2Task Engine
- *STEPLABEL* **String** Name of the step.
- *STATUSDECODE* **String** The keyword value for the process step's status. For the list of keyword values, see "Status Reference" on page 189.
- *SYSTEMDECODE* **String** The name of the system upon which the step was executed. If the *SYSTEM* code is 1, the service returns “Integration Server;” if the *SYSTEM* code is 2, the service returns “Workflow.”

stepNames **HashMap** List of the returned steps executed in the specified process instance. This is a hash map of key/value pairs, where the keys are the step IDs of the steps and values are the names of the steps.

stepCidList **HashMap** List of the context IDs for the steps executed in the specified process instance. This is a hash map of key/value pairs, where the keys are the step IDs of the steps and the values are the context IDs for the steps.

cidList **HashSet** A list of context IDs that correlate to the steps executed for the specified process instance.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getInstanceTransitions

Retrieves information from the logging database about the transitions that were logged for the most recent iteration of a process instance.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

instanceID **String** Instance ID for the process instance for which to retrieve logged transitions. Specify the complete, exact ID.

Output Parameters

transitions **Document List** List of the logged transitions for the specified process instance. For each transition, the following fields are returned:

- *INSTANCEID* **String** Instance ID.
- *INSTANCEITERATION* **Number** Instance iteration.
- *SOURCESTEPID* **String** Step ID of the source step for this transition.
- *SOURCESTEPITERATION* **Number** Iteration of the source step for this transition.
- *TARGETSTEPID* **String** Step ID of the target step.
- *AUDITTIMESTAMP* **Number** Time the transition was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the transition was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getPagedInstanceList

Retrieves process instances between the *fromIndex* and *toIndex* , inclusive. Use this service instead of pub.monitor.process.instance:getInstanceList for better performance when retrieving records.

Input Parameters

modelID	**String** Optional. Complete model ID used for the process instances you want retrieved. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID* , do not use the *modelNameInput* parameter.
modelNameInput	**String** Optional. Complete or partial model name used for the process instances you want retrieved. You can retrieve model names by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEYDECODE* output parameter. When you specify *modelNameInput* , do not use the *modelID* parameter.
instanceID	**String** Optional. Complete or partial instance ID for the process instances you want retrieved.
customID	**String** Optional. The full, user-defined ID for the process instances that you want retrieved. User-defined IDs are assigned by executing the pub.prt.log:logCustomID service.
parentInstanceID	**String** Optional. Complete or partial instance ID for the parent process of the process instances that you want retrieved.
status	**String** Optional. The status of the process instances you want retrieved. The status you specify is matched against the most recently logged status. Use *status* if you want the list to contain all process instances of a single specified status. Specify the numerical value that represents the status. For a list of values you can specify, see "Status Reference" on page 189. **Note:** If you do not specify *status* or *statusSet* , the service returns process instances of all statuses.
statusSet	**String List** Optional. A set of statuses for the process instances you want retrieved. Use *statusSet* when you want the list to contain process instances of more than one specified status. For each status that you specify in *statusSet* , specify the numerical value that represents the status, as described above for the *status* parameter.

dateCreated

String Optional. A date range for the process instances you want retrieved. The date range identifies the date of the most recent log entry for the process instances. If you use this parameter, do not use the *fromDate* or *toDate* parameter.

- `Today` Today
- `Yesterday` Yesterday
- `In the last 7 days` Within the last 7 days, including today.
- `Last week` Any day in the previous calendar last week (A week is Sunday through Saturday.)
- `This week` Any day in the current calendar week. (A week is Sunday through Saturday.)
- `Last month` Any day in the previous calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

fromDate

String Optional. The start date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or after this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *toDate* to specify the end date; do not use the *dateCreated* parameter when you use the *fromDate* and *toDate* parameters.

toDate

String Optional. The end date of when data was logged for the process instances you want retrieved. The service retrieves process instances with entries logged on or before this date. Use the format *YYYY-MM-DD HH:MM:SS* .

stepID

String Optional. The full ID of a step that was executed in the process instances you want retrieved. You can retrieve step IDs using the service. When you use this *stepID* parameter, you must also specify:

- The *modelID* parameter to identify the model in which to search for the specified step.
- The *user* or *role* parameters.

user

String Optional. To retrieve information about tasks, user on which to match (that is, user that performed a task). When you use this *user* parameter, you must also specify the *modelID* parameter.

role

String Optional. To retrieve information about tasks, role on which to match (that is, role that performed a task). When you

use this *role* parameter, you must also specify the *modelID* parameter.

maxRows **String** Optional. Maximum number of process instances to return, starting with those most recently logged. By default, the service gets all process instances.

sortColumn **String** Optional. How to sort the returned list of process instances. By default, the service sorts the returned data using `AUDITTIMESTAMP`. This parameter works with the *sortAscending* parameter. Sort by:

- `PROCESSLABEL` Model name.
- `INSTANCEID` Process instance ID.
- `PARENTINSTANCEID` Parent process instance ID, if any.
- `STATUS` Most recent status; the sort order is based on the numerical values associated with statuses. For a list of the status values, see "Status Reference" on page 189.
- `AUDITTIMESTAMP` The last time data was logged for the process instance.

sortAscending **String** Optional. Whether to sort the returned list of process instances in ascending or descending order. The entries are sorted by the field identified by the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

isAnd **String** Optional. Whether the service is to use an `AND` or an `OR` condition for the criteria specified in the input parameters.

- `true` Default. Use an `AND` condition. The service returns process instances that match all the criteria you specify.
- `false` Use an `OR` condition. The service returns process instances that match any of the criteria you specify.

fromIndex **String** The starting start row number of the data to retrieve.

toIndex **String** The ending row number of the data to retrieve.

Output Parameters

instances **Document List** List of process instances that match the specified criteria. For each process instance, the following fields are returned:

- *instanceId* **String** Instance ID of the process instance.

- *parentInstanceId* **String** Instance ID of the process instance's parent instance, if any.
- *instanceIteration* **Number** Instance iteration of the process instance.
- *customId* **String** The user-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.
- *modelId* **String** Unique model ID of the process instance.
- *modelName* **String** Name of the process instance.
- *modelVersion* **String** The version of the model.
- *firstStatus* **Number** First status of the process instance.
- *firstTime* **String** Time that data was first logged for the process instance, in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where: YYYY-MM-DD is the date, hh:mm:ss:SSS is the time and zzz is the time in milliseconds.
- *lastStatus* **Number** Latest status of the process instance.
- *lastTime* **String** Time that data was last logged for the process instance, in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where: YYYY-MM-DD is the date, hh:mm:ss:SSS is the time and zzz is the time in milliseconds.

message **String** Optional. Error that occurred during the execution of this service, if this service encountered an error.

Usage Notes

This service is related to the pub.monitor.process.instance:getInstanceList service but limits the process models returned by the *fromIndex* and *toIndex* . This query performs best if there is not a large difference between the input parameters.

pub.monitor.process.instance:getProcessList

Retrieves the model names of all process instances that have run, successfully or otherwise, and for which logging data exists in the logging database.

Input Parameters

None.

Output Parameters

processNames **Document List** List of the process models for process instances that have run and for which logging data exists in the logging database. For each process model, the following fields are returned:

- *PROCESSKEY* **String** Internal model ID that Designer assigned to the process model.
- *PROCESSKEYDECODE* **String** User-defined name assigned to the process model.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getRecentlyCompleted

Retrieves information for the most recently completed process instances. The service returns information for up to twenty process instances that completed in the last two weeks.

Input Parameters

None.

Output Parameters

recentlyCompleted **Document List** List of the most recently completed process instances. For each process instance, the following fields are returned:

- *AUDITTIMESTAMP* **Number** Latest logged timestamp in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Latest logged timestamp in string format, YYYY-MM-DDhh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *PARENTINSTANCEID* **String** Parent instance ID, if any.
- *PARENTINSTANCEITERATION* **Number** Parent instance iteration, if any.

- *INSTANCEID* **String** Instance ID.
- *INSTANCEITERATION* **Number** Instance iteration.
- *STATUS* **String** Status code, which is `2` for completed instances.
- *STATUSDECODE* **String** Status value, which is `Completed`.
- *PROCESSKEY* **String** Internal model ID that Designer assigned to the process model used for the process instance.
- *PROCESSKEYDECODE* **String** User-defined name of the process model that was used for the process instance.
- *CUSTOMID* **String** The user-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instance:getRecentlyCreated

Retrieves information for the most recently created process instances. The service returns information for up to twenty process instances that were created in the last two weeks.

Input Parameters

None.

Output Parameters

recentlyCreated **Document List** List of the most recently created process instances. For each process instance, the following fields are returned:

- *AUDITTIMESTAMP* **Number** Latest logged timestamp in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Latest logged timestamp in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds

- zzz is the time zone

- *PARENTINSTANCEID* **String** Parent instance ID, if any.
- *PARENTINSTANCEITERATION* **Number** Parent instance iteration, if any.
- *INSTANCEID* **String** Instance ID.
- *INSTANCEITERATION* **Number** Instance iteration.
- *STATUS* **String** Status code, which is `1` for Started.
- *STATUSDECODE* **String** Status value, which is `Started`.
- *PROCESSKEY* **String** Internal model ID that Designer assigned to the process model used for the process instance.
- *PROCESSKEYDECODE* **String** User-defined name of the process model that was used for the process instance.
- *CUSTOMID* **String** The user-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.

pub.monitor.process.instance:getRecentlyFailed

Retrieves information for the most recently failed process instances. The service returns information for up to twenty process instances that failed in the last two weeks.

Input Parameters

None.

Output Parameters

recentlyFailed **Document List** List of the most recently failed process instances. For each process instance, the following fields are returned:

- *AUDITTIMESTAMP* **Number** Latest logged timestamp in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Latest logged timestamp in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds

- zzz is the time zone

- *PARENTINSTANCEID* **String** Parent instance ID, if any.
- *PARENTINSTANCEITERATION* **Number** Parent instance iteration, if any.
- *INSTANCEID* **String** Instance ID.
- *INSTANCEITERATION* **Number** Instance iteration.
- *STATUS* **String** Status code, which is `4` for failed instances.
- *STATUSDECODE* **String** Status value, which is `Failed`.
- *PROCESSKEY* **String** Internal model ID that Designer assigned to the process model used for the process instance.
- *PROCESSKEYDECODE* **String** User-defined name of the process model that was used for the process instance.
- *CUSTOMID* **String** The user-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.

pub.monitor.process.instance:getRecentlySuspended

Retrieves information for the most recently suspended process instances. The service returns information for up to twenty process instances that were suspended in the last two weeks.

Input Parameters

None.

Output Parameters

recentlySuspended **Document List** List of the most recently suspended process instances. For each process instance, the following fields are returned. For the process instance:

- *AUDITTIMESTAMP* **Number** Latest logged timestamp in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Latest logged timestamp in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:

  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

- *PARENTINSTANCEID* **String** Parent instance ID, if any.
- *PARENTINSTANCEITERATION* **Number** Parent instance iteration, if any.
- *INSTANCEID* **String** Instance ID.
- *INSTANCEITERATION* **Number** Instance iteration.
- *STATUS* **String** Status code, which is `8` for suspended instances.
- *STATUSDECODE* **String** Status value, which is `Suspended`.
- *PROCESSKEY* **String** Internal model ID that Designer assigned to the process model used for the process instance.
- *PROCESSKEYDECODE* **String** User-defined name of the process model that was used for the process instance.
- *CUSTOMID* **String** The user-defined ID for the process instance that was assigned by executing the pub.prt.log:logCustomID service.

7 pub.monitor.process.instanceControl Folder

- Summary of Elements in This Folder 104
- pub.monitor.process.instanceControl:changeInstanceStatus 104
- pub.monitor.process.instanceControl:changeUserTaskPriority 105
- pub.monitor.process.instanceControl:resubmitAllFailed 106
- pub.monitor.process.instanceControl:resubmitInstanceStep 107

Summary of Elements in This Folder

Service and Description
pub.monitor.process.instanceControl:changeInstanceStatus Stops, suspends, or resumes a specified process instance.
pub.monitor.process.instanceControl:changeUserTaskPriority Changes the priority of the user tasks in a specified process instance.
pub.monitor.process.instanceControl:resubmitAllFailed Resubmits all iterations of the specified failed process instances at the failed step.
pub.monitor.process.instanceControl:resubmitInstanceStep Resubmits a specified iteration of a specified completed or failed process instance at a specified iteration of a process step.

pub.monitor.process.instanceControl:changeInstanceStatus

Stops, suspends, or resumes a specified process instance.

Input Parameters

instanceID **String** Instance ID for the process instance that you want to stop, suspend, or resume. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

instanceIteration **Number** Specific process iteration that you want to stop, suspend, or resume.

modelID **String** Complete model ID of the model that the process instance uses. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

controlAction **String** Control action that you want to take against the specified process instance.

- `CANCEL` Stop the running process iteration.
- `SUSPEND` Suspend the running process iteration.
- `RESUME` Resume the suspended process iteration.
- String x

modelVersion **String** The version of the process model that the process instance uses. The model version for externally executed process models is always 1.

Output Parameters

controlMessage **String** Message that indicates the success of the status change.

pub.monitor.process.instanceControl:changeUserTaskPriority

Changes the priority of the user tasks in a specified process instance.

Input Parameters

ProcessInstanceID **String** Instance ID for the process instance containing the user tasks.

Priority **String** The priority of the user tasks in the specified process instance. Valid values are:

- critical
- high
- medium
- low
- none

Output Parameters

EscalatedTasks	**String** The IDs of the user tasks that had their priority changed.
message	**String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instanceControl:resubmitAllFailed

Resubmits all iterations of the specified failed process instances at the failed step.

Input Parameters

modelID	**String** Optional. Complete model ID of the model used for the process instances whose iterations you want to resubmit. You can retrieve model IDs by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEY* output parameter. When you specify *modelID*, do not use the *modelNameInput* parameter. **Note:** Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.
modelNameInput	**String** Optional. Partial model name of the model used for the process instances whose iterations you want to resubmit. You can retrieve model names by invoking the pub.monitor.process.instance:getProcessList service and using the value returned in the *processNames/PROCESSKEYDECODE* output parameter. When you specify *modelNameInput*, do not use the *modelID* parameter. **Note:** Whether Monitor treats *modelNameInput* as case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.
dateCreated	**String** Optional. A date range for the process iterations you want to resubmit. The date range identifies the date of the most recent log entry for the process iterations. To select processes

whose most recent entry was logged, set the parameters as follows:

- `Today` The current date.
- `Yesterday` The day before the current date.
- `In the last 7 days` Within the last 7 days, including the current date.
- `Last week` Any day in the last calendar last week. (A week is Sunday through Saturday.)
- `This week` Any day in this calendar week. (A week is Sunday through Saturday.)
- `Last month` Any day in the last calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

maxRows **String** Optional. Maximum number of process iterations to resubmit, starting with those most recently logged. By default, the service resubmits the 100 most recent process iterations.

Output Parameters

controlMessage **String** Message that indicates the success of the resubmission.

Usage Notes

You must have the proper functional privileges to resubmit processes. For more information, see information about granting users the ability to perform Monitor actions in *webMethods Monitor User's Guide*.

pub.monitor.process.instanceControl:resubmitInstanceStep

Resubmits a specified iteration of a specified completed or failed process instance at a specified iteration of a process step.

Input Parameters

instanceID **String** Instance ID for the process instance to resubmit. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

instanceIteration **Number** Iteration of the process instance to resubmit.

stepID	**String** The full step ID for the step at which to resubmit the process instance.
stepIteration	**Number** Iteration of the step to resubmit.
stepPipeline	**Document** Optional. Input pipeline for the step.
globalData	**Document** Optional. Global data for this step.

Output Parameters

controlMessage	**String** Message that indicates the success of the resubmission.

Usage Notes

You can retrieve all input data needed to run this service using the pub.monitor.process.instanceSteps:getStepDetails service.

You must have the proper functional privileges to resubmit processes. If you have functional privileges to resubmit but not to modify the input pipeline or global data, the service retrieves that information from the logging database even if you try to supply them on the *stepPipeline* and *globalData* parameters. For more information, see information about granting users the ability to perform Monitor actions in *webMethods Monitor User's Guide*.

8 pub.monitor.process.instanceSteps Folder

- Summary of Elements in This Folder 110
- pub.monitor.process.instanceSteps:getStepActivityLogs 110
- pub.monitor.process.instanceSteps:getStepControl 112
- pub.monitor.process.instanceSteps:getStepCustomData 113
- pub.monitor.process.instanceSteps:getStepDetails 114
- pub.monitor.process.instanceSteps:getStepErrors 117
- pub.monitor.process.instanceSteps:getStepHistory 118
- pub.monitor.process.instanceSteps:getStepPipeline 119

Summary of Elements in This Folder

Service and Description
pub.monitor.process.instanceSteps:getStepActivityLogs Retrieves all user-defined messages that were logged by the process instance that contains the specified process step. This includes all messages that any service, invoked in any iteration of any step logged.
pub.monitor.process.instanceSteps:getStepControl Retrieves all control actions for all iterations of a specified process step.
pub.monitor.process.instanceSteps:getStepCustomData Retrieves the user-specified document field values, which were logged for a specified iteration of either a specified Invoke step or Task step.
pub.monitor.process.instanceSteps:getStepDetails Retrieves details about a step that has a specified status and that was executed in the specified process iteration and step iteration.
pub.monitor.process.instanceSteps:getStepErrors Retrieves the errors associated with a specified process step.
pub.monitor.process.instanceSteps:getStepHistory Retrieves all log entries from the logging database for all iterations of the specified step.
pub.monitor.process.instanceSteps:getStepPipeline Retrieves the pipeline details of the specified process step executed in the specified process iteration and step iteration.

pub.monitor.process.instanceSteps:getStepActivityLogs

Retrieves all user-defined messages that were logged by the process instance that contains the specified process step. This includes all messages that any service, invoked in any iteration of any step logged.

Input Parameters

instanceID **String** Instance ID of the process instance in which the step exists. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

instanceIteration **String** A number that indicates the iteration of the process instance that contains the step.

stepID **String** Optional. The full step ID for the step. You can retrieve step IDs using the pub.monitor.process.modelSteps:getStepIDNames service.

stepIteration **String** Optional. A number that indicates the iteration of the step for which to retrieve document field values.

Output Parameters

activityLogData **Document List** List of the retrieved user-defined messages. For each user-defined message, the following fields are returned:

- *STEPID* **String** Step ID of the step that logged the document field value.
- *ENTRYTYPE* **String** The type of user-defined message, that is error, warning, or message.
- *FULLMESSAGE* **String** The text of the user-defined message. It will contain up to 1024 characters.
- *BRIEFMESSAGE* **String** A brief version of the text of the message that contains only up to 240 characters.
- *B2BUSER* **String** The Integration Server user that invoked the service that logged the user-defined message.
- *SERVERID* **String** ID of server where service that logged the user-defined message ran.
- *AUDITTIMESTAMP* **Number** Time the message was logged, in epoch time; that is, the number of seconds since January 1, 1970.

- *AUDITTIMESTRING* **String** Time the message was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instanceSteps:getStepControl

Retrieves all control actions for all iterations of a specified process step.

Input Parameters

instanceID **String** Instance ID of the process instance in which the step exists. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

instanceIteration **String** A number that indicates the iteration of the process instance that contains the step.

stepID **String** Optional. The full step ID for the step. You can retrieve step IDs using the pub.monitor.process.modelSteps:getStepIDNames service.

stepIteration **String** Optional. A number that indicates the iteration of the step for which to retrieve document field values.

Output Parameters

stepControl **Document List** List of the retrieved control actions. For each control action, the following fields are returned:

- *ACTION* **String** The type of control action. The service returns the numerical value that represents the type, as follows:
  - `1` Service Resubmit
  - `2` Document Resubmit
  - `3` Process Resubmit
  - `4` Process Suspend

- 5 Process Resume
- 6 Process Stop

- *ACTIONDECODE* **String** The keyword value for the type of control action; that is, one of the following:
  - Service Resubmit
  - Document Resubmit
  - Process Resubmit
  - Process Suspend
  - Process Resume
  - Process Stop
- *INSTANCEITERATION* **Number** The step iteration of the step when the control action was taken.
- *USERNAME* **String** User who initiated the control action.
- *SERVERID* **String** Server where the action was initiated.
- *AUDITTIMESTRING* **String** Time the action was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instanceSteps:getStepCustomData

Retrieves the user-specified document field values, which were logged for a specified iteration of either a specified Invoke step or Task step.

Input Parameters

instanceID **String** Instance ID of the process instance that contains the step whose user-specified document field values to retrieve. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

instanceIteration **Number** Iteration of the process instance that contains the step.

stepID **String** The full step ID for the step. You can retrieve step IDs using the pub.monitor.process.modelSteps:getStepIDNames service.

stepIteration **Number** Iteration of the step for which to retrieve document field values.

Output Parameters

stepCustomData **Document List** List of the retrieved user-specified document field values. For each document field, the following fields are returned:

- *STEPID* **String** Step ID of the step that logged the document field value.
- *STEPITERATION* **Number** Step iteration of the step that logged the document field value.
- *INSTANCEITERATION* **Number** Iteration of the process instance when the step logged the document field value.
- *DOCUMENTNAME* **String** The name of the user-defined document that contains the field that was logged.
- *FIELDNAME* **String** Name of the logged field.
- *STRINGVALUE* **String** The value of the field if the logged field value is a String.
- *NUMBERVALUE* **String** The value of the field if the logged field value is a number.
- *DATEVALUE* **String** The value of the logged field if the logged field value is a date.
- *AUDITTIMESTAMP* **Number** Timestamp that the field was logged, in epoch time; that is, the number of seconds since January 1, 1970.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instanceSteps:getStepDetails

Retrieves details for a specific step, with a specified status that was executed in the specified process iteration and step iteration.

Input Parameters

stepID **String** The full step ID of the step whose details you want to retrieve. You can retrieve step IDs using the pub.monitor.process.instance:getInstanceSteps service.

instanceID **String** Instance ID for the process instance. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues.

status **String** Status of the step for which to retrieve information. Specify the numerical value that represents the status. For a list of valid values, see "Status Reference" on page 189.

instanceIteration **Number** Iteration of the process instance that contains the step.

stepIteration **Number** Iteration of the step for which to retrieve information.

Output Parameters

stepData **Document** The details for the step. *stepData* returns the following:

- *stepLabel* **String** Name of the step.
- *timestamp* **String** Timestamp for the retrieved status, in string format YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *description* **String** Description of the step.
- *user* **String** For task steps, the user assigned to perform the task.
- *roleName* **String** For task steps, the roles assigned to perform the task.
- *rootContextID* **String** The root context ID is an internal identifier that Integration Server uses.
- *parentContextID* **String** The parent context ID is an internal identifier that Integration Server uses.

- *contextID* **String** Context ID to retrieve errors associated with the step.
- *instanceID* **String** Instance ID for the process instance.
- *instanceIteration* **Number** Iteration of the process instance that contained the step.
- *instanceStatus* **String** Status of the process instance that contains the step. The service returns a numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *modelID* **String** Model ID for the model that contains the step and that the process instance uses.
- *modelName* **String** Name of the model that contains the step and that the process instance uses.
- *stepID* **String** Step ID of the step.
- *stepIteration* **Number** Step iteration of the step.
- *status* **String** Status of the step. The service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *statusDecode* **String** Status value of the step. The service returns the localized keyword value that represents the status. For the list of status values, for example, “Started” or “Completed” (in English), see "Status Reference" on page 189.
- *serverID* **String** Host name and port of the Integration Server on which the step ran.
- *pipeNull* **String** Whether the step's input pipeline was logged.
  - `true` Input pipeline was not logged.
  - `false` Input pipeline was logged and is available for viewing or resubmission.
- *globalDataNull* **String** Whether the step's input global data was logged.
  - `true` Input global data was not logged.
  - `false` Input global data was logged and is available for viewing or resubmission.
- *icon* **String** URL for the step's status icon in the file system.
- *stepPipeline* **Document** Input pipeline for the step, if logged.
- *globalData* **Document** Global data for the step, if logged.

message **String** Error that occurred during the execution of this service if this service encountered an error.

processResubmit **String** Indicates if the user who invoked the pub.monitor.process.instanceSteps:getStepDetails service belongs to a My webMethods role that has the functional privilege to resubmit the process instance:

- `true` User has the functional privilege to resubmit the process.
- `false` User does not have the functional privilege.

pipelineEdit **String** Whether the user who invoked the pub.monitor.process.instanceSteps:getStepDetails service belongs to a My webMethods role that has the functional privilege to modify the pipeline for a process step.

- `true` User has the privilege to modify the step.
- `false` User does not have the privilege to modify the step.

Usage Notes

The *stepPipeline* and *globalData* parameters are encoded as IData. You can modify and resubmit them (if you have the proper functional privileges) with the pub.monitor.process.instanceControl:resubmitInstanceStep service.

pub.monitor.process.instanceSteps:getStepErrors

Retrieves the errors associated with a specified process step.

Input Parameters

instanceID **String** Instance ID of the process instance in which the step exists. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

instanceIteration **String** A number that indicates the iteration of the process instance that contains the step.

stepID **String** Optional. The full step ID for the step. You can retrieve step IDs using the pub.monitor.process.modelSteps:getStepIDNames service.

stepIteration **String** Optional. A number that indicates the iteration of the step for which to retrieve document field values.

Output Parameters

stepErrors **Document List** The retrieved list of errors. If there are no errors associated with the step, the service returns a null value. If there are errors, the following fields are returned for each error:

- *AUDITTIMESTAMP* **Number** Time the error was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *ERRORSTACKTRACE* **String** Stack trace for the error.
- *SERVERID* **String** Server ID of the server where the error occurred.

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instanceSteps:getStepHistory

Retrieves all log entries from the logging database for all iterations of the specified step.

Input Parameters

instanceID **String** Instance ID of the process instance in which the step exists. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

stepID **String** The full step ID of the step. To retrieve step IDs, use the pub.monitor.process.instance:getInstanceSteps service.

Output Parameters

stepHistory **com.wm.util.Table** Logging information for the specified step.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.instanceSteps:getStepPipeline

Retrieves the pipeline details of the specified process step executed in the specified process iteration and step iteration.

Input Parameters

instanceId **String** Optional. Instance ID for the process instance. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *instanceID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

instanceIteration **Number** Optional. Iteration of the process instance that contains the step.

stepId **String** Optional. The full step ID for the step details to retrieve. To retrieve step IDs, use the pub.monitor.process.instance:getInstanceSteps service.

stepIteration **Number** Optional. The step iteration for which to retrieve information.

Output Parameters

pipeXML **String**. Contains the pipeline information of the specified step of the process. If there is no data in the pipeline for the process step, the string contains `No Data` message.

9 pub.monitor.process.model Folder

- Summary of Elements in This Folder 122
- pub.monitor.process.model:clearProcessFilter 123
- pub.monitor.process.model:generateModelImage 124
- pub.monitor.process.model:getCustomFieldDefinitions 125
- pub.monitor.process.model:getCustomFields 126
- pub.monitor.process.model:getModelDetails 127
- pub.monitor.process.model:getModelImage 129
- pub.monitor.process.model:getModelList 130
- pub.monitor.process.model:getModelListWithFilter 134
- pub.monitor.process.model:getModelNames 138
- pub.monitor.process.model:getModelNamesWithFilter 138
- pub.monitor.process.model:getProcessFilter 140
- pub.monitor.process.model:getUnusedModels 140
- pub.monitor.process.model:setProcessFilter 144

Summary of Elements in This Folder

Service and Description
pub.monitor.process.model:clearProcessFilter Removes a filter field and its value that was previously assigned to a process model.
pub.monitor.process.model:generateModelImage Generates the model image for a specified process model in SVG format.
pub.monitor.process.model:getCustomFieldDefinitions Retrieves the custom field definitions for the specified model.
pub.monitor.process.model:getCustomFields Retrieves all custom document fields that were defined for a specified process model.
pub.monitor.process.model:getModelDetails Retrieves the detailed information for a specified process model.
pub.monitor.process.model:getModelImage Retrieves the model image for a specified process model.
pub.monitor.process.model:getModelList Retrieves a list of process models for which at least one process instance of the model has run.
pub.monitor.process.model:getModelListWithFilter Retrieves a list of process models that match a specified filter and for which at least one process instance of the model has run.
pub.monitor.process.model:getModelNames Retrieves the names and IDs of all process models for which information is stored in the logging database.
pub.monitor.process.model:getModelNamesWithFilter

Service and Description
Retrieves names and IDs of process models that match a specified filter and for which information is stored in the logging database.
pub.monitor.process.model:getProcessFilter Retrieves the list of all filter fields and values that are assigned to all process models.
pub.monitor.process.model:getUnusedModels Retrieves process models that are available for execution, but for which no logging data exists in the Process Audit Log database component.
pub.monitor.process.model:setProcessFilter Assigns a filter field and its value to a process model. You can then use the filter field and value in a filter that limits the items returned when you execute services to retrieve a list of process models, list of process model names and IDs, or a list of process instances.

pub.monitor.process.model:clearProcessFilter

Removes a filter field and its value that was previously assigned to a process model.

Note: The service uses the input parameters to search for the filter field/value to remove. Whether the search is case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to locate the field/value to remove. If the search is case-sensitive, be sure to specify the field name and value using the exact case used when the field was assigned.

Input Parameters

processKey **String** The internal identifier (that is, process key) of the process model from which to remove a field and value.

fieldName **String** The name of the field to remove from the process model.

stringValue **String** The value of the field that you want to remove.

Output Parameters

result **String** The result of the service. If the service completes successfully, the *result* returns `success`. Otherwise, *result* returns the failure message.

Usage Notes

To retrieve a list of filter fields that are already set, use the pub.monitor.process.model:getProcessFilter service. Use this service if you need to determine the exact combination of upper- and lowercase characters used for a field and/or value.

pub.monitor.process.model:generateModelImage

Generates the model image for a specified process model in SVG format. The service writes the generated image to the file system in IData format.

Input Parameters

modelID **String** Model ID of the process model image to create. To retrieve model IDs, invoke the pub.monitor.process.model:getModelNames service and use the value returned in the *modelNames/PROCESSKEY* output parameter.

> **Note:** Whether Monitor treats *modelID* as case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

subNodeID **String** Optional. Node ID of a step within the process model that represents an inline process (or subprocess) within the process; that is a set of steps that are collapsed into a single step. When you specify *subnodeID*, the service generates the image for the subprocess only, not for the entire process. To retrieve the node IDs for the subprocesses within a process, execute the pub.monitor.process.modelSteps:getModelSteps service and use the value returned in the *modelSteps/INLINESTEPID* field.

deployVersion **String** The version of the process model for which you want to generate a model image. The model version for externally executed process models is always 1. There is no image associated with integration processes.

Output Parameters

imageData **Document** The generated image. The returned *imageData* document contains the following fields:

- `imageURL`**String** URL of the image in the file system.
- `width`**String** Width of the image.
- `height`**String** Height of the image.
- `type`**String** Image type. The value is `2` for images in SVG format.
- `imageMap`**Document List** A list of IDs for the child processes (that is, referenced processes and subprocesses) contained in the model. To obtain the details for one of these child processes, you can re-execute this service specifying its ID. For each child process in the mode, the following fields are returned:
  - *key* **String** If the child process is a referenced process, *key* contains the model ID of the referenced process model.
  - *stepid* **String** If the child process is a subprocess, *stepid* contains the ID of the step that represents the inline subprocess.
  - *x***String** The top x coordinate of the step icon for the child process.
  - *y* **String** The left y coordinate of the step icon for the child process.
  - *x2* **String** The bottom x coordinate of the step icon for the child process.
  - *y2* **String** The right y coordinate of the step icon for the child process.

imageError **String** Errors that occurred while generating the image.

pub.monitor.process.model:getCustomFieldDefinitions

Retrieves the custom field definitions for the specified model.

Input Parameters

modelID **String** Model ID of the process model from which to retrieve custom field definitions. To retrieve model IDs, invoke the pub.monitor.process.model:getModelNames service and use the value returned in the *modelNames*/*PROCESSKEY* output parameter.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

deployVersion **String** The version of the process model. The model version for externally executed and integration processes models is always 1.

Output Parameters

customFields **Document List** List of the retrieved custom field definitions. For each custom field definition, the following fields are returned:

- *PROCESSKEY* **String** The model ID of the process model that contains the custom field definition.
- *STEPID* **String** Step ID of the step that is set up to log a field for a custom field definition.
- *DOCUMENTTYPE* **String** Fully-qualified name of the IS document type for the document that contains the custom field.
- *DOCUMENTNAME* **String** User-defined name of the document as it was specified in Designer.
- *FIELDNAME* **String** The field name as defined in the IS document type.
- *FIELDTYPE* **String** Data type for the field, which will be one of String, Number, or Date.
- *TYPE* **String** Whether the custom field is in the pipeline data or the global data.
  - `1` The custom field is in the pipeline data.
  - `2` The custom field is in the global data.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.model:getCustomFields

Retrieves all custom document fields that were defined for a specified process model.

Input Parameters

modelID **String** Complete model ID of the process model for which you want to retrieve user-specified document fields. You can retrieve model IDs by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames*/*PROCESSKEY* output parameter.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

deployVersion **String** The version of the process model. The model version for externally executed and integration processes models is always 1.

Output Parameters

customFields **Document List** List of the retrieved user-defined document fields. For each user-defined document field, the following fields are returned:

- *FIELDNAME* **String** The field name as defined in the IS document type.
- *FIELDTYPE* **String** Data type of the field, which will be one of String, Number, or Date.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.model:getModelDetails

Retrieves the detailed information for a specified process model.

Input Parameters

modelID **String** Complete model ID of the process model. You can retrieve model IDs by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames*/*PROCESSKEY* output parameter.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or

SQL server) handles the queries that Monitor issues to obtain data.

deployVersion **String** The version of the process model. The model version for externally executed and integration processes models is always 1.

Output Parameters

modelDetails **Document** The retrieved detail information for the specified process model. The returned IS document contains the following fields:

- *modelName* **String** Name of the process model.
- *deployTime* **String** Date and time the process model was last added to the Process Audit Log:
  - For a webMethods-executed process model, this is the last date and time the process model version was built and uploaded for execution.
  - For an externally executed process model, this is the last date and time the process model was uploaded for analysis.
  - For an integration process, this is when the pub.monitor.integrationProcessLogging:createProcessMetadata service was used to log information about the process.
- *description* **String** Description of the process model.
- *createdBy* **String** User who created the process model.
- *enabled* **String** Whether the model is enabled or disabled (webMethods-executed process models only).
  - `0` Disabled.
  - `1` Enabled
  - `-1` Disabled and process model generation failed.
  - `-2` Enabled and process model generation failed.
- *started* **String** Number of process instances in Started status.
- *ended* **String** Number of process instances in Completed status.
- *failed* **String** Number of process instances in Failed status.
- *suspended* **String** Number of process instances in Suspended status.
- *canceled* **String** Number of process instances in Stopped status.
- *resumed* **String** Number of process instances in Resumed status.

- *resubmitted* **String** Number of process instances in Resubmitted status.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.model:getModelImage

Retrieves the model image for a specified process model.

Input Parameters

modelID **String** Complete model ID of the process model for which you want to obtain the model image. You can retrieve model IDs by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames/PROCESSKEY* output parameter.

> **Note:** Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

subNodeID **String** Optional. Node ID of a step within the process model that represents an inline process (or subprocess) within the process; that is a set of steps that have been collapsed into a single step. When you specify *subnodeID* , the service generates the image for the subprocess only, not for the entire process. You can retrieve the node IDs for the sub processes within a process by executing the pub.monitor.process.modelSteps:getModelSteps service and using the value returned in the *modelSteps/INLINESTEPID* field.

type **String** Whether to return the image in JPG or SVG format.

- `1` JPG
- `2` SVG

deployVersion **String** The version of the process model for which you want to generate a model image. The model version for externally executed process models is always 1. There is no image associated with integration processes.

Output Parameters

imageData **Document** The model image. The returned *imageData* document contains these fields:

- *imageStream* **byte[]** A byte array that contains the image data.
- *width* **String** Width of the image.
- *height* **String** Height of the image.
- *type* **String** Image type.
  - `1` Image is in JPG format.
  - `2` Image is in SVG format.

imageError **String** Errors that occurred during the retrieval of the image.

pub.monitor.process.model:getModelList

Retrieves a list of process models for which at least one process instance of the model has run.

Input Parameters

modelID **String** Optional. Complete model ID of a process model. Use the *modelID* parameter if you want to determine whether a specific model has had any running process instances. You can retrieve model IDs by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames/PROCESSKEY* output parameter. If you specify *modelID*, do not use the *modelNameInput* parameter.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

modelNameInput **String** Optional. Partial model name of a model. Use the *modelNameInput* parameter if you want to limit the retrieved models to those that match the partial model name. You can retrieve model names by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames/PROCESSKEYDECODE* output parameter. When you specify *modelNameInput*, do not use the *modelID* parameter.

Note: Whether Monitor treats *modelNameInput* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

enabled **String** Optional. For webMethods-executed process models only, whether you want to retrieve models that are enabled or disabled.

- `0` Disabled
- `1` Enabled
- `2` Default. Either enabled or disabled

sortColumn **String** Optional. How you want the service to sort the list of retrieved process models. This parameter works with the *sortOrder* parameter.

- `started` The value in the Started column, which is the number of process instances that use this process model that are started.
- `completed` The value in the Completed column, which is the number of process instances that are completed.
- `suspended` The value in the Suspended column, which is the number of process instances that are suspended.
- `failed` The value in the Failed column, which is the number of process instances that have failed.
- `canceled` The value in the Canceled column, which is the number of process instances that have been canceled.
- `resumed` The value in the Resumed column, which is the number of process instances that have been resumed.
- `resubmitted` The value in the Resubmitted column, which is the number of process instances that have been resubmitted.
- `modelName` Default. Name of the process model.
- `enabled` Whether the process models are enabled or disabled.
- `deployTime` The time the process models were updated in the Process Audit Log database component.
  - For a webMethods-executed process model, this is the last date and time the process model version was built and uploaded for execution.
  - For an externally executed process model, this is the last date and time the process model was uploaded for analysis.

- For an integration process, this is when the pub.monitor.integrationProcessLogging:createProcessMetadata service was used to log information about the process.

sortOrder **String** Optional. Whether to sort in ascending or descending order. This parameter works with the *sortColumn* parameter.

- `0` Default. Sort in ascending order.
- `1` Sort in descending order.

Output Parameters

modelTable **Document List** List of the retrieved webMethods-executed process models, externally executed process models, integration processes, and task models. For each model, the following fields are returned:

- *MODELID* **String** Unique model ID of the process model.
- *MODELNAME* **String** Name of the process model.

For the process model, the total number of process instances:

- *STARTED* **String** Number of process instances in Started status.
- *COMPLETED* **String** Number of process instances in Completed status.
- *FAILED* **String** Number of process instances in Failed status.
- *SUSPENDED* **String** Number of process instances in Suspended status.
- *CANCELED* **String** Number of process instances in Canceled status.
- *RESUMED* **String** Number of process instances in Resumed status.
- *RESUBMITTED* **String** Total number of process instances in Resubmitted status.
- *DEPLOYDATE* **String** Time the process model was last updated in the Process Audit Log database component. The time is in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *DEPLOYTIME* **Number** Time the process model was last updated in the Process Audit Log database component. The time is

specified in epoch time, which is the number of seconds since January 1, 1970.

- *ENABLED* **String** For webMethods-executed process models, whether the process model is enabled or disabled.
  - `0` Disabled
  - `1` Enabled
  - `-1` Disabled and generation failed
  - `-2` Enabled and generation failed
- *DeployVersion* **String** The version of the process model. Externally executed process models are always version 1.

wfTable **Document List** List of workflow models that were created using webMethods Workflow. If you still have workflow models in your logging database, they are returned in the following fields for each workflow model.

- *MODELID* **String** Unique model ID of the workflow.
- *MODELNAME* **String** Name of the workflow.
- *STARTED* **String** Number of processes in Started status.
- *COMPLETED* **String** Number of processes in Completed status.
- *FAILED* **String** Number of processes that are currently in the Failed status and are based on the workflow.
- *SUSPENDED* **String** Number of processes that are currently in the Suspended status and are based on the workflow.
- *CANCELED* **String** Number of processes in Canceled status.
- *RESUMED* **String** Number of processes in Resumed status.
- *RESUBMITTED* **String** Number of processes in Resubmitted status.
- *DEPLOYDATE* **String** Time the model was last updated in the Process Audit Log database tables, in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *DEPLOYTIME* **Number** Time the model was last updated in the Process Audit Log database component. The time is in epoch time, which is the number of seconds since January 1, 1970.

message **String** Error that occurred while executing this service.

Usage Notes

If there are multiple versions of a process model, the service returns all versions.

pub.monitor.process.model:getModelListWithFilter

Retrieves a list of process models that match a specified filter and for which at least one process instance of the model has run.

Use the pub.monitor.process.model:setProcessFilter service to set the filters that you want to reference in this getModelListWithFilter service.

Note: Whether the search is case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

criteriaField **String** Optional. Filter that you want to use to limit the list of returned values. Use the following format to create the filter:

field1=value1 , value2 , ... , valueN

For example, if you have set a filter field named *countryCode* and want to return only those process models for which *countryCode* is `es`, use the following filter, `countryCode=es`

To return process models for which the *countryCode* is `es` or `fr`, use the filter: `countryCode=es, fr`

To use multiple filter fields, specify `and` between the fields/values. For example, to return process models for which the *countryCode* is `es` or `fr` and the *department* is `AP`, use the filter, `countryCode=es, fr and department=AP`

modelID **String** Optional. Complete model ID of a process model. Use the *modelID* parameter if you want to determine whether a specific model has had any running process instances. To retrieve model IDs, invoke the pub.monitor.process.model:getModelNames service and use the value returned in the *modelNames/PROCESSKEY* output parameter. If you specify *modelID* , do not use the *modelNameInput* parameter.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2,

or SQL server) handles the queries that Monitor issues to obtain data.

modelNameInput **String** Optional. Partial model name of a model. Use the *modelNameInput* parameter if you want to limit the retrieved models to those that match the partial model name. You can retrieve model names by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames*/*PROCESSKEYDECODE* output parameter. When you specify *modelNameInput* , do not use the *modelID* parameter.

Note: Whether Monitor treats *modelNameInput* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

enabled **String** Optional. For webMethods-executed process models only, whether you want to retrieve models that are enabled or disabled.

- `0` Disabled.
- `1` Enabled.
- `2` Default. Either enabled or disabled.

sortColumn **String** Optional. How to sort retrieved process models. This parameter works with the *sortOrder* parameter. Sort by any of the following columns:

- `started` Number of process instances model that are started.
- `completed` Number of process instances that are completed.
- `suspended` Number of process instances that are suspended.
- `failed` Number of process instances that have failed.
- `canceled` Number of process instances that have been canceled.
- `resumed` Number of process instances that have been resumed.
- `resubmitted` Number of process instances that have been resubmitted.
- `modelName` Default. The name of the process model.
- `enabled` Whether the process models are enabled.
- `deployTime` The time the process models were updated in the Process Audit Log database tables. The value varies by model type:

- webMethods-executed process model, the last date and time the process model version was built and uploaded for execution.
- Externally executed process model, the last date and time the process model was uploaded for analysis.
- Integration process, when the pub.monitor.integrationProcessLogging: createProcessMetadata service was used to log information about the process.

sortOrder **String** Optional. Whether to sort in ascending or descending order. This parameter works with the *sortColumn* parameter.

- `0` Default. Sort in ascending order.
- `1` Sort in descending order.

Output Parameters

modelTable **Document List** List of the retrieved webMethods-executed process models, externally executed process models, integration processes, and task models. For each model, the following fields are returned:

- *MODELID* **String** Unique model ID of the process model.
- *MODELNAME* **String** Name of the process model.
- *STARTED* **String** Number of process instances in Started status.
- *COMPLETED* **String** Number of process instances in Completed status.
- *FAILED* **String** Number of process instances in Failed status.
- *SUSPENDED* **String** Number of process instances in Suspended status.
- *CANCELED* **String** Number of process instances in Canceled status.
- *RESUMED* **String** Number of process instances in Resumed status.
- *RESUBMITTED* **String** Number of process instances in Resubmitted status.
- *DEPLOYDATE* **String** Time the process model was last updated in the Process Audit Log database tables. The time is specified in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

- *DEPLOYTIME* **Number** Time the process model was last updated in the Process Audit Log tables. The time is in epoch time, which is the number of seconds since January 1, 1970.

ENABLED **String** For webMethods-executed process models, whether the process model is enabled or disabled.

- `0` Disabled.
- `1` Enabled.
- `-1` Disabled and generation failed.
- `-2` Enabled and generation failed.

wfTable **Document List** List of workflow models that were created using webMethods Workflow. If you still have workflow models in your logging database, they are returned in the following fields for each workflow model:

- *MODELID* **String** Unique model ID of the workflow.
- *STARTED* **String** Number of processes in Started status.
- *COMPLETED* **String** Number of processes in Completed status.
- *FAILED* **String** Number of processes in Failed status.
- *SUSPENDED* **String** Number of processes in Suspended status.
- *CANCELED* **String** Number of processes in Canceled status.
- *RESUMED* **String** Number of processes in Resumed status.
- *RESUBMITTED* **String** Number of processes in Resubmitted status.
- *DEPLOYDATE* **String** Time the model was last updated in the Process Audit Log database tables, in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *DEPLOYTIME* **Number** Time the model was last updated in the Process Audit Log database component. The time is in epoch time, which is the number of seconds since January 1, 1970.

message **String** Error that occurred during the execution of this service if this service encountered an error.

Usage Notes

The following are services are related to the *criteriaFilter* input field:

- To assign a filter field that you can reference in *criteriaFilter* , use the pub.monitor.process.model:setProcessFilter service.
- To retrieve a list of filter fields that are already set, use the pub.monitor.process.model:getProcessFilter service.
- To clear a filter field, use the pub.monitor.process.model:clearProcessFilter service.

pub.monitor.process.model:getModelNames

Retrieves the names and IDs of all process models for which information is stored in the logging database.

Input Parameters

None.

Output Parameters

modelNames **Document List** List of the retrieved model names and IDs. The following fields are returned for each process model.

- *PROCESSKEYDECODE* **String** The name of the process model.
- *PROCESSLABEL* **String** The name of the process model.
- *PROCESSKEY* **String** The model ID for the process.
- *PROCESSPATH* **String** The directory path to the folder where the process file in is stored.

message **String** Error that occurred during the execution of this service if this service encountered an error.

Usage Notes

If there are multiple versions of a process model, the service returns all versions.

pub.monitor.process.model:getModelNamesWithFilter

Retrieves names and IDs of process models that match a specified filter and for which information is stored in the logging database.

Use the pub.monitor.process.model:setProcessFilter service to set the filters that you want to reference in this getModelNamesWithFilter service.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

criteriaField **String** Optional. Filter to limit the list of returned values. Use the following format:

field1 =value1 , value2 , ... , valueN

For example, if you have set a filter field *countryCode* and want to return only the names and IDs of process models for which *countryCode* is `es`, use the filter, `countryCode=es`

To return the names and IDs of process models for which the *countryCode* is `es` or `fr`, use the filter, `countryCode=es, fr`

To use multiple filter fields, specify `and` between the fields/values. For example, to return names and IDs of process models for which the *countryCode* is `es` or `fr` and the *department* is `AP`, use the following filter, `countryCode=es, fr and department=AP`

Output Parameters

modelNames **Document List** List of the retrieved model names and IDs. The following fields are returned for each process model.

- *PROCESSKEYDECODE* **String** The name of the process model.
- *PROCESSLABEL* **String** The name of the process model.
- *PROCESSKEY* **String** The model ID for the process.
- *PROCESSPATH* **String** The directory path to the folder where the process file in is stored.

message **String** Error that occurred during the execution of this service if this service encountered an error.

Usage Notes

The following are services are related to the *criteriaFilter* input field:

- To assign a filter field that you can reference in *criteriaFilter* , use the pub.monitor.process.model:setProcessFilter service.

- To retrieve a list of filter fields that are already set, use the pub.monitor.process.model:getProcessFilter service.
- To clear a filter field, use the pub.monitor.process.model:clearProcessFilter service.

pub.monitor.process.model:getProcessFilter

Retrieves the list of all filter fields and values that are assigned to all process models.

Input Parameters

None.

Output Parameters

processFilter **Document List** List of the retrieved filter fields and values. The following fields are returned for each field.

- *processKey* **String** The internal ID (that is, process key) of a process model.
- *fieldName* **String** The name of the field assigned to the process model.
- *stringValue* **String** The value of the field.

Usage Notes

- The output from this service is sorted so that all the fields assigned to a process model are grouped together. Additionally, within the output for a process model, all the settings for the same field name are grouped together.
- To assign a filter field to a process model, use the pub.monitor.process.model:setProcessFilter service.
- To clear a filter field, use the pub.monitor.process.model:clearProcessFilter service.

pub.monitor.process.model:getUnusedModels

Retrieves process models that are available for execution, but for which no logging data exists in the Process Audit Log database component.

This includes process models for which no process instance has ever been executed and process models for which instances have run, but the logged data for the executed instances has been archived or deleted from the logging database.

Input Parameters

modelID

String Optional. Complete model ID of a process model. Use the *modelID* parameter to determine whether a specific model has no logged data for process instances. To retrieve model IDs, invoke the pub.monitor.process.model:getModelNames service and use the value returned in the *modelNames*/*PROCESSKEY* output parameter. If you specify *modelID*, do not use the *modelNameInput* parameter.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

modelNameInput

String Optional. Partial model name of a process model. Use the *modelNameInput* parameter if you want to limit the retrieved process models to those that match the partial model name. To retrieve model names, invoke the pub.monitor.process.model:getModelNames service and use the value returned in the *modelNames*/*PROCESSKEYDECODE* output parameter. When you specify *modelNameInput*, do not use the *modelID* parameter.

Note: Whether Monitor treats *modelNameInput* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

enabled

String For webMethods-executed process models only, whether you want to retrieve models that are enabled or disabled.

- `0` Disabled
- `1` Enabled
- `2` Default. Either enabled or disabled

sortBy

String Optional. How to sort the retrieved list of process models. This parameter works with the *sortOrder* parameter.

- `PROCESSLABEL` Default. Sorts by the name of the process model.
- `ENABLED` Sorts by whether the process models is enabled.
- `DESCRIPTION` Sorts by the description of the process model.
- `CREATEDBY` Sorts by the user that created the process model.

- `DEPLOYMENTTIMESTAMP` Sorts by the time the process model were updated in the Process Audit Log database. This time varies by the process model type:
  - webMethods-executed, the last date and time the process model version was built and uploaded for execution.
  - Externally executed, the last date and time the process model was uploaded for analysis.
  - Integration, when pub.monitor.integrationProcessLogging: createProcessMetadata service was used to log information about the process.

sortOrder **String** Optional. Whether to sort in ascending or descending order. This parameter works with the *sortBy* parameter.

- `0` Default. Sort in ascending order.
- `1` Sort in descending order.

Output Parameters

unusedModels **Document List** List of the retrieved webMethods-executed process models, externally executed process models, integration processes, and task models. For each model, the following fields are returned:

- *PROCESSKEYDECODE* **String** The name of the process model.
- *PROCESSLABEL* **String** The name of the process model.
- *PROCESSKEY* **String** The model ID for the process model.
- *PROCESSPATH* **String** The directory path to the folder where the process file is stored.
- *DESCRIPTION* **String** Description of the process model.
- *CREATEDBY* **String** User who created the process model.
- *DEPLOYMENTTIME* **Number** Time the process model was last updated in the Process Audit Log database. The time is in epoch time, which is the number of seconds since January 1, 1970.
- *ENABLED* **String** For webMethods-executed process models, whether the process model is enabled or disabled.
  - `0` Disabled process models.
  - `1` Enabled
  - `-1` Disabled and generation failed.
  - `-2` Enabled and generation failed.

- *TYPE* **String** A number that represents the type of model. This will always be 1 for models created using Designer.

- *DEPLOYMENTTIMESTAMP* **String** Time the process model was last updated in the Process Audit Log database. The time is in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

unusedWorkflows **Document List** List of workflow models that were created using webMethods Workflow. If you still have workflows in your logging database, they are returned in the following fields for each workflow model.

- *PROCESSKEYDECODE* **String** The name of the workflow.
- *PROCESSLABEL* **String** The name of the workflow.
- *PROCESSKEY* **String** The ID for the workflow.
- *PROCESSPATH* **String** This is null.
- *DESCRIPTION* **String** Description of the workflow.
- *CREATEDBY* **String** User who created the workflow.
- *DEPLOYMENTTIME* **Number** Deployment time, in epoch time; that is, the number of seconds since January 1, 1970.
- *ENABLED* **String** Always returns 0 for workflows.
- *TYPE* **String** A number that represents the type of model. This will always be 2 for the models created using Workflow.
- *DEPLOYMENTTIMESTAMP* **String** Deployment time in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

message **String** Error that occurred during the execution of this service if this service encountered an error.

Usage Notes

If there are multiple unused versions of a process model, the service returns all versions.

pub.monitor.process.model:setProcessFilter

Assigns a filter field and its value to a process model. You can then use the filter field and value in a filter that limits the items returned when you execute services to retrieve a list of process models, list of process model names and IDs, or a list of process instances.

You can specify filters with the following services:

- pub.monitor.process.model:getModelListWithFilter to limit the list of process models that are returned to those that match the filter you specify.
- pub.monitor.process.model:getModelNamesWithFilter to limit the list of process model names and IDs that are returned to those that match the filter you specify.
- pub.monitor.process.instance:getInstanceListWithFilter to limit the list of process instances that are returned to those that match filters you specify.

Input Parameters

processKey	**String** The internal identifier (that is, process key) of the process model to which you want to assign the filter field and value.
fieldName	**String** The name of the filter field you want to assign to the process model. For example, you might specify `countryCode` so that you can associate a country code with a process model.
stringValue	**String** The value to set for the filter field. For example, if you specify `countryCode` for *fieldName* , you might specify `es` for *stringValue* .

Output Parameters

result	**String** The outcome of the service. If the service completes successfully, the value of *result* will be `success`. Otherwise, *result* will contain the failure message.

Usage Notes

- To retrieve a list of filter fields that are already set, use the pub.monitor.process.model:getProcessFilter service.
- To clear a filter field, use the pub.monitor.process.model:clearProcessFilter service.

10 pub.monitor.process.modelControl Folder

- Summary of Elements in This Folder 146
- pub.monitor.process.modelControl:changeModelEnabled 146
- pub.monitor.process.modelControl:deleteUnusedModel 147
- pub.monitor.process.modelControl:getProcessLogical Servers 148
- pub.monitor.process.modelControl:isModelEnabled 149
- pub.monitor.process.modelControl:refreshModelNames 150

Summary of Elements in This Folder

Service and Description
pub.monitor.process.modelControl:changeModelEnabled Enables or disables the specified webMethods-executed process model.
pub.monitor.process.modelControl:deleteUnusedModel Deletes information about a specified process model from the logging database and, as a result, also the Monitor user interface.
pub.monitor.process.modelControl:getProcessLogical Servers Retrieves the names of the logical servers for a specified process model.
pub.monitor.process.modelControl:isModelEnabled Returns a list of process models and whether they are currently enabled or disabled.
pub.monitor.process.modelControl:refreshModelNames Clears the process model labels stored in the cache.

pub.monitor.process.modelControl:changeModelEnabled

Enables or disables the specified webMethods-executed process model.

Input Parameters

enableModelID **String** Complete model ID of a process model.
You can retrieve model IDs by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames/PROCESSKEY* output parameter.

Note: Whether Monitor treats *enableModelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

enabledStatus **String** Optional. Whether the process model is currently enabled or disabled. You can determine whether it is enabled or disabled by using the pub.monitor.process.modelControl:isModelEnabled service.

- `0` Default. Disabled model.
- `1` Enabled model.

Output Parameters

message **String** Success or error message, as appropriate.

Usage Notes

- If you disable a process model that is invoked by another process model, when a process instance based on the parent model is executed, it will fail at the step that attempts to start a process instance for the disabled process model.
- If there are multiple versions of a process model, this service acts on the latest version.

pub.monitor.process.modelControl:deleteUnusedModel

Deletes information about a specified process model from the logging database and, as a result, also the Monitor user interface.

Input Parameters

modelToDelete **String** Complete model ID of the process model you want to delete. You can retrieve model IDs by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames/PROCESSKEY* output parameter.

Note: Whether Monitor treats *modelToDelete* as case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

modelVersion **String** The version of the process model that you want to delete. The model version for externally executed process models is always 1.

Output Parameters

message **String** Confirmation or error message, as appropriate.

See Also

pub.monitor.process.modelControl:changeModelEnabled

pub.monitor.process.modelControl:isModelEnabled

Usage Notes

- You can delete information for process models that are disabled and unused.
- If there are multiple versions of a process model, this service acts on the latest version.

pub.monitor.process.modelControl:getProcessLogical Servers

Retrieves the names of the logical servers for a specified process model.

Input Parameters

modelID **String** Complete model ID of the process model whose logical server names you want to retrieve. You can retrieve model IDs by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames/PROCESSKEY* output parameter.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

logicalServers **Document List** List of the retrieved logical server names. For each logical server name, the following fields are returned:

SERVER **String** Name of the logical server.

message **String** Error that occurred during the execution of this service if this service encountered an error.

Usage Notes

If there are multiple versions of a process model, this service retrieves the logical server names for the latest version.

pub.monitor.process.modelControl:isModelEnabled

Returns a list of process models and whether they are currently enabled or disabled.

Input Parameters

None.

Output Parameters

modelStatuses **Document List** List of the retrieved process models and their enabled/disabled status. For each process model, the following fields are returned:

- *PROCESSLABEL* **String** Name of the process model.
- *MODELID* **String** Model ID of the process model.
- *ENABLED* **String** For webMethods-executed process models, whether the model is enabled or disabled. The following list indicates the value the service returns for each model status:
  - **0** Disabled models.
  - **1** Enabled models
  - **-1** Disabled and generation failed models.
  - **-2** Enabled and generation failed models.

message **String** Error that occurred during the execution of this service if this service encountered an error.

modelVersion **String** The version of the process model.

Usage Notes

If there are multiple versions of a process model, this service retrieves only returns information for the latest version.

pub.monitor.process.modelControl:refreshModelNames

Clears the process model labels stored in the cache.

After the process model labels are cleared from the cache, if a process model label is required, the process model label is fetched from the database and then stored in the cache.

Input Parameters

None.

Output Parameters

None.

11 pub.monitor.process.modelSteps Folder

- Summary of Elements in This Folder 152
- pub.monitor.process.modelSteps:getModelSteps 152
- pub.monitor.process.modelSteps:getModelTransitions 155
- pub.monitor.process.modelSteps:getStepIDNames 156

Summary of Elements in This Folder

Service and Description
pub.monitor.process.modelSteps:getModelSteps Retrieves information for each step in a specified webMethods-executed or externally executed process model.
pub.monitor.process.modelSteps:getModelTransitions Retrieves the step transitions for a specified webMethods-executed process model.
pub.monitor.process.modelSteps:getStepIDNames Retrieves step ID, step label, and step type for each step in a specified process model.

pub.monitor.process.modelSteps:getModelSteps

Retrieves information for each step in a specified webMethods-executed or externally executed process model.

Input Parameters

modelID **String** Complete model ID of the process model that contains the step information you want to obtain. You can retrieve available model IDs by running the pub.monitor.process.model:getModelNames service. Use the value returned in the *modelNames()/PROCESSKEY* output parameter. For example: MyProcessProject/MyProcess.

> **Note:** Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

deployVersion **String** The version of the process model. The model version for externally executed and integration processes is always 1. There is no image associated with integration processes.

Output Parameters

modelSteps **Document List** List of the steps in the specified process. The following fields are returned for each step:

- *PROCESSKEY* **String** Model ID of the process model for which the visual elements were retrieved. For example: *projectName/processName*
- *STEPID* **String** Step ID of the step.
- *STEPLABEL* **String** Step label of the step.
- *DESCRIPTION* **String** Description for the step.
- *TYPE* **String** The keyword that describes the type of step. Valid values are:
  - `Abstract Task` A BPMN abstract task activity.
  - `webMethods Subprocess` A webMethods subprocess.
  - `Referenced Process` A webMethods referenced process step (deprecated).
  - `Boundary Timer Event` A BPMN boundary timer event.
  - `BPMN Subprocess` A BPMN subprocess.
  - `Call Activity` A BPMN call activity.
  - `Service Task` A BPMN service task activity.
  - `User Task` A BPMN user task activity.
  - `Manual Task` A BPMN manual task activity.
  - `Rule Task` A BPMN rule task activity.
  - `Send Task` A BPMN send task activity.
  - `Receive Task` A BPMN receive task activity.
  - `Gateway` A BPMN gateway.
  - `Start Event` A BPMN start event.
  - `Intermediate Event` A BPMN intermediate event.
  - `End Event` A BPMN end event.
- *STEPINTTYPE* **String** Numerical representation of the type of visual element. The service returns one of the following numerical values as described below:
  - `1` Abstract task
  - `2` webMethods subprocess
  - `3` Referenced process (webMethods referenced process, deprecated)
  - `12` Boundary timer event
  - `13` BPMN subprocess
  - `14` Call activity
  - `30` Service task

- `31` User task
- `32` Manual task
- `33` Rule task
- `34` Send task
- `35` Receive task
- `40` Gateway
- `50` Start event
- `80` Intermediate event
- `110` End event

- *ICON* **String** URL to the image file that is used for the icon of the step. This will either be a URL to the file provided with Designer or if the model uses a non-standard image, the URL provided for that image when the model was created. Null for an abstract task.
- *ICONBYTE* **String** Byte array that contains the icon of the step.
- *ICON_X* **String** X coordinate of the top, left corner for where the icon is placed within the model image.
- *ICON_Y* **String** Y coordinate for of the top, left corner for where the icon is placed within the model image.
- *ICON_WIDTH* **String** Width of the icon used for the step.
- *ICON_HEIGHT* **String** Height of the icon used for the step.
- *SUBPROCESSKEY* **String** If the visual element represents a referenced process, *SUBPROCESSKEY* contains the model ID for the referenced process model. Otherwise, *SUBPROCESSKEY* will be null.
- *INLINESTEPID* **String** If visual element represents an inline subprocess, *INLINESTEPID* contains the model ID for the parent process model in which the subprocess resides. Otherwise, *INLINESTEPID* will be null.
- *REFSTEPID* **String** An internal identifier that the Integration Server uses.
- *SERVER* **String** Logical server name of the server that is to execute the step.
- *IS_START* **String** Whether the step is a start step. The service returns the following values for steps:
  - `1` A start step
  - `0` Not a start step

- *IS_STOP* **String** Whether the step is a stop step. The service returns the following values:
  - `1` A stop step
  - `0` Not a stop step

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.modelSteps:getModelTransitions

Retrieves the step transitions for a specified webMethods-executed process model.

Input Parameters

modelID **String** Complete model ID of the process model whose transitions you want to retrieve. You can retrieve model IDs by invoking the pub.monitor.process.model:getModelNames service and using the value returned in the *modelNames/PROCESSKEY* output parameter.

> **Note:** Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

deployVersion **String** The version of the process model. The model version for externally executed and integration processes is always 1. There is no image associated with integration processes.

Output Parameters

modelTransitions **Document List** List of the retrieved step transitions. For each step transition, the following fields are returned:

- *PROCESSKEY* **String** Model ID of the process model that contains the step transition.
- *SOURCESTEPID* **String** Step ID of the source step for this transition.
- *TARGETSTEPID* **String** Step ID of the target step for this transition.
- *SOURCEX* **String** X coordinate of the start of the line.

- *SOURCEY* **String** Y coordinate of the start of the line.
- *TARGETX* **String** X coordinate of the end of the line.
- *TARGETY* **String** Y coordinate of the end of the line.
- *VISUALTYPE* **String** Whether the line in the process model that represents the transition is straight or curved. The service codes that the service returns indicate the line shape:
  - `0` straight
  - `1` curved
- *TYPE* **String** Whether the transition is an internal (i.e., between steps in an internal pool or no pool) or external transition (i.e., between steps from an internal pool to an external pool). The service codes that the service returns indicate the line shape:
  - `0` internal transition
  - `1` external transition
- *LABEL* **String** Label of the transition, if any.
- *CONDITION* **String** Transition condition, if any.
- *LEVELCHANGE* **String** An internal value that Integration Server uses.

message **String** Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.process.modelSteps:getStepIDNames

Retrieves step ID, step label, and step type for each step in a specified process model.

Input Parameters

modelID **String** Complete model ID of the process model that contains the step information you want to obtain. You can retrieve available model IDs by running the pub.monitor.process.model:getModelNames service. Use the value returned in the *modelNames()/PROCESSKEY* output parameter. For example: MyProcessProject/MyProcess.

Note: Whether Monitor treats *modelID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

deployVersion **String** The version of the process model. The model version for externally executed and integration processes is always 1.

Output Parameters

stepNames **Document List** List of the retrieved step names and step IDs. For each step, the following fields are returned:

- *STEPID* Step ID of the step.
- *STEPLABEL* Step label for the step.
- *STEPINTTYPE* **String** Numerical representation of the type of step. The service returns one of the following values:
  - `1` Abstract task
  - `2` webMethods subprocess
  - `3` Referenced process (legacy webMethods referenced process, deprecated)
  - `12` Boundary timer event
  - `13` BPMN subprocess)
  - `14` Call activity
  - `30` Service task
  - `31` User task
  - `32` Manual task
  - `33` Rule task
  - `34` Send task
  - `35` Receive task
  - `40` Gateway
  - `50` Start event
  - `80` Intermediate event
  - `110` End event

message **String** Error that occurred during the execution of this service if this service encountered an error.

12 pub.monitor.service Folder

- Summary of Elements in This Folder 160
- pub.monitor.service:exists 161
- pub.monitor.service:getActions 162
- pub.monitor.service:getActivityLogs 163
- pub.monitor.service:getChildDetails 164
- pub.monitor.service:getCustomData 165
- pub.monitor.service:getDetails 167
- pub.monitor.service:getErrors 168
- pub.monitor.service:getHistory 169
- pub.monitor.service:getList 171
- pub.monitor.service:getListCustomData 175
- pub.monitor.service:getListCustomDataSet 179
- pub.monitor.service:getPipeline 183
- pub.monitor.service:isResubmittable 184

Summary of Elements in This Folder

Service and Description
pub.monitor.service:exists Checks whether a specified service has logged data to the logging database.
pub.monitor.service:getActions Retrieves all resubmit actions associated with a specified service.
pub.monitor.service:getActivityLogs Retrieves all user-defined messages that were logged for a specified service or all services that are descendants of the specified service.
pub.monitor.service:getChildDetails Retrieves detailed information for all services that have the same parent service.
pub.monitor.service:getCustomData Retrieves user-defined logged field values of a service and returns them as name/value pairs.
pub.monitor.service:getDetails Retrieves information about the most recently logged status for a specified service.
pub.monitor.service:getErrors Retrieves all errors that were logged for a specified service.
pub.monitor.service:getHistory Retrieves the status history for a specified service.
pub.monitor.service:getList Retrieves a list of services whose most recent log entry (that is, current state) matches specified criteria.
pub.monitor.service:getListCustomData

Service and Description
Retrieves a list of services that meet the specified criteria, including specifying the value of a single logged field, which instructs the service to return all instances where the value you specify was logged for a specified custom logged field.
pub.monitor.service:getListCustomDataSet Retrieves a list of services that meet the specified criteria, including specifying the value of multiple logged fields, which instructs the service to return all services where the value you specify was logged for multiple specified custom logged field.
pub.monitor.service:getPipeline Retrieves the input pipeline for a specified service.
pub.monitor.service:isResubmittable Checks whether a specified service is resubmittable.

pub.monitor.service:exists

Checks whether a specified service has logged data to the logging database.

Input Parameters

contextID **String** Context ID for the service for which to check the database. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

exists **String** Whether logging data for the service exists in logging database.

- `true` Logging data for the service exists in the database.
- `false` Logging data for the service does not exist in the database.

pub.monitor.service:getActions

Retrieves all resubmit actions associated with a specified service.

Input Parameters

contextID **String** Context ID for the service whose control actions you want to get. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

actions **Document List** List of the retrieved resubmit actions. For each resubmit action, the following fields are returned:

- *ROOTCONTEXTID* **String** Context ID of the service's root service. If the service is a top-level service, the root context ID is the same as the context ID.
- *PARENTCONTEXTID* **String** Context ID of the service's parent service if the service was nested or resubmitted.
- *CONTEXTID* **String** Context ID for the service.
- *ACTION* **String** Code for the resubmit action that was performed on the service; that is, the service returns the number, 1, for this parameter.
- *ACTIONDECODE* **String** The keyword for the resubmit action; that is, in English, the service returns “Service Resubmit” for this parameter.
- *USERNAME* **String**Integration Server user that performed the resubmit action.
- *SERVERID* **String** Host name and port for the Integration Server on which the service was resubmitted.
- *AUDITTIMESTAMP* **Number** Time the action was logged, in epoch time; that is, the number of seconds since January 1, 1970.

- *AUDITTIMESTRING* **String** Time the action was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

pub.monitor.service:getActivityLogs

Retrieves all user-defined messages that were logged for a specified service or all services that are descendants of the specified service.

Input Parameters

contextID **String** Context ID for the service whose user-defined messages you want to retrieve. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

activityLogs **Document List** List of the retrieved messages. For each user-defined message, the following fields are returned:

- *ROOTCONTEXTID* **String** Context ID of the service's root service. If the service is a top-level service, the root context ID is the same as the context ID.
- *PARENTCONTEXTID* **String** Context ID of the service's parent service if the service was nested or resubmitted.
- *CONTEXTID* **String** Context ID for the service.
- *ENTRYTYPE* **String** The type of user-defined message, that is error, warning, or message.
- *BRIEFMESSAGE* **String** A brief version of the text of the message that contains only up to 240 characters.
- *FULLMESSAGE* **String** The text of the user-defined message. It will contain up to 1024 characters.

- *B2BUSER* **String**Integration Server user name of the client that invoked the service.
- *SERVERID* **String** DNS name and port of the Integration Server that ran the service (for example, `titanium.east.webmethods.com:5555`).
- *AUDITTIMESTAMP* **Number** Time the message was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the message was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

pub.monitor.service:getChildDetails

Retrieves detailed information for all services that have the same parent service.

Input Parameters

contextID **String** Context ID for the parent service. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

details **Document List** List of the retrieved services that have the specified parent service. For each service, the following fields are returned:

- *ROOTCONTEXTID* **String** Context ID of the parent service's root service. If the parent service is a top-level service, the root context ID is the same as the context ID.
- *PARENTCONTEXTID* **String** Context ID of the specified parent service.
- *CONTEXTID* **String** Context ID for the child service; the information in *details* is for this service.

- *STATUS* **String** Status of the service. The service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value of the service. The service returns the keyword value that represents the status. For the list of keyword values, for example, “Started” or “Completed,” see "Status Reference" on page 189.
- *SERVICENAME* **String** Fully qualified name of the child service; the information in *details* is for this service.
- *USERID* **String**Integration Server user name of the client that invoked the service.
- *RESUBMITTABLE* **String** Whether the service exists in the logging database and is resubmittable.
  - `true` Service exists and is resubmittable.
  - `false` Service does not exist and is not resubmittable.
- *ERRORMESSAGE* **String** If the service's status is 4 (Failed), the error message for the service.
- *SERVERID* **String** DNS name and port of the Integration Server that ran the service (for example, `titan.east.webmethod.com:5555`).
- *DURATION* **String** If the service's status is 2 (Completed) or 4 (Failed), length of time the service ran (in milliseconds).
- *AUDITTIMESTAMP* **Number** Time the service's status was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the service's status was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

pub.monitor.service:getCustomData

Retrieves user-defined logged field values of a service and returns them as name/value pairs.

Input Parameters

contextID **String** Context ID for the service whose user-defined logged fields you want to retrieve. Specify the complete, exact ID.

fieldName **String** Optional. Complete name of the logged field whose value you want to retrieve.

sortBy **String** Optional. Value to use to sort the returned list of logged field values. This parameter works with the *sortOrder* parameter.

- `SERVERID` DNS name and port for the Integration Server that ran the service.
- `MSGID` ID of user-defined messages that contain the logged fields.
- `FIELDNAME` Name of the field for which a value was logged.
- `STRINGVALUE` Values of string type document fields.
- `NUMBERVALUE` Values of Number type document fields.
- `DATEVALUE` Values of Date type document fields.

sortOrder **String** Optional. Whether to sort the returned list of logged field values in ascending or descending order. The documents are sorted by the field identified by the *sortBy* parameter.

- `0` Default. Sort in ascending order.
- `1` Sort in descending order.

Output Parameters

customData **Document List** The list of returned logged field values. For each logged field value, the following fields are returned:

- *CONTEXTID* **String** Context ID for the service.
- *SERVERID* **String** ID of server where service that logged the fields ran.
- *MSGID* **String** ID of user-defined messages that contain the logged fields.
- *FIELDNAME* **String** Name of the field for which a value was logged.
- *STRINGVALUE* **String** The value of the field if the logged field value is a string.
- *NUMBERVALUE* **String** The value of the field if the logged field value is a number; otherwise the service returns 0.0 in this field.
- *DATEVALUE* **String** The value of the field if the logged field value is date.

message **String** Optional. Error that occurred during the execution of this service if this service encountered an error.

pub.monitor.service:getDetails

Retrieves information about the most recently logged status for a specified service.

Input Parameters

contextID **String** Context ID for the service for which to get information. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

details **Document** The retrieved information for the specified service. The following fields are returned for the service.

- *ROOTCONTEXTID* **String** Context ID of the service's root service. If the service is a top-level service, the root context ID is the same as the context ID.
- *PARENTCONTEXTID* **String** Context ID of the service's parent service, if the service was nested or resubmitted.
- *CONTEXTID* **String** Context ID for the service.
- *CUSTOMCONTEXTID* **String** The full, user-defined ID of the retrieved service that was assigned by executing the pub.flow:setCustomContextID service.
- *AUDITEDPARENTID* **String** Complete ID of the parent service for which you have logged user-defined fields.
- *STATUS* **String** Status of the service. The getDetails service returns the numerical value that represents the status. For a description of the numerical values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value of the service. The getDetails service returns the keyword value that represents the status. For the list of keyword values, for example, "Started" or "Completed", see "Status Reference" on page 189.
- *SERVICENAME* **String** Name of the service.

- *USERID* **String**Integration Server user name of the client that invoked the service.
- *RESUBMITTABLE* **String** Whether the service exists in the logging database and is resubmittable.
  - `true` Service exists and is resubmittable.
  - `false` Service does not exist and is not resubmittable.
- *ERRORMESSAGE* **String** If the service's status is 4 (Failed), the error message for the service.
- *SERVERID* **String** DNS name and port of the Integration Server that ran the service (for example, `titan.east.webmethod.com:5555`).
- *DURATION* **String** If the service's status is 2 (Completed) or 4 (Failed), length of time the service ran (in milliseconds)

Usage Notes

If there are multiple log entries with the same timestamp and the timestamp is the most recent timestamp, the service returns all the entries.

pub.monitor.service:getErrors

Retrieves all errors that were logged for a specified service.

Input Parameters

contextID **String** Context ID for the service whose errors you want to retrieve. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

errors **Document List** The retrieved errors for the specified service. For each error, the following fields are returned:

- *ROOTCONTEXTID* **String** Context ID for the root service of the service that logged the error message. If the service that logged the

error message is a top-level service, the root context ID is the same as the context ID.

- *PARENTCONTEXTID* **String** If the service was nested or resubmitted, context ID for the parent service of the service that logged the error message.
- *CONTEXTID* **String** Context ID for the service that logged the error message.
- *ERRORMSG* **String** The text of the error message.
- *ERRSTACKTRACE* **String** Stack trace data associated with the error.
- *SERVICENAME* **String** Name of the service that logged the error message.
- *SERVERID* **String** DNS name and port of the Integration Server that ran the service (for example, `titan.east.webmethod.com:5555`).
- *AUDITTIMESTAMP* **Number** Time the error was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the error was logged in string format, YYYY-MM-DDhh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

pub.monitor.service:getHistory

Retrieves the status history for a specified service.

Input Parameters

contextID **String** Context ID for the service whose status history you want to retrieve. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

history **Document List** The retrieved logged statuses for the specified service. For each logged status, the following fields are returned:

- *ROOTCONTEXTID* **String** Context ID of the service's root service. If the service is a top-level service, the root context ID is the same as the context ID.
- *PARENTCONTEXTID* **String** Context ID of the service's parent service, if the service was nested or resubmitted.
- *CONTEXTID* **String** Context ID for the service.
- *STATUS* **String** Status of the service when the message was logged. The getHistory service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value of the service when the message was logged. The getHistory service returns the keyword value that represents the status. For the list of keyword values, for example, “Started” or “Completed,” see "Status Reference" on page 189.
- *SERVICENAME* **String** Name of the service.
- *USERID* **String**Integration Server user name of the client that invoked the service.
- *RESUBMITTABLE* **String** Whether the service is currently resubmittable.
  - `true` Service is resubmittable.
  - `false` Service is not resubmittable.
- *ERRORMESSAGE* **String** If the service's status was 4 (Failed) when the message was logged, the text of the error message.
- *SERVERID* **String** DNS name and port for the Integration Server that ran the service (for example, `titan.east.webmethod.com:5555`).
- *DURATION* **String** If the service's status was 2 (Completed) or 4 (Failed) when the message was logged, length of time the service ran (in milliseconds).
- *AUDITTIMESTAMP* **Number** Time the service's status was logged, in epoch time; that is, the number of seconds since January 1, 1970.

- *AUDITTIMESTRING* **String** Time the service's status was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone

pub.monitor.service:getList

Retrieves a list of services whose most recent log entry (that is, current state) matches specified criteria.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

serviceName **String** Optional. Fully qualified or partial service name on services that you want to retrieve. The *serviceNameExact* parameter is used with this parameter.

serviceNameExact **String** Optional. How to match on the value specified in *serviceName* .

- `true` Exactly match the *serviceName* parameter.
- `false` Default. Contains a substring that matches the *serviceName* parameter.

serverID **String** Optional. Complete or partial DNS name and port for the Integration Server that ran the service (for example, `titan.east.webmethod.com:5555`). The *serverIDExact* parameter is used with this parameter.

serverIDExact **String** Optional. How to match on the value specified in *serverID* .

- `true` Exactly matches the *serverID* parameter.
- `false` Default. Contains a substring that matches the *serverID* parameter.

contextID **String** Optional. Complete context ID of the services you want to retrieve.

parentContextID **String** Optional. Complete context ID of the parent service of the services you want to retrieve.

rootContextID **String** Optional. Complete context ID for the root service of the services you want to retrieve.

customContextID **String** Optional. The full, user-defined ID for the services that you want to retrieve. User-defined IDs are assigned by executing the pub.flow:setCustomContextID service. Use *customContextID* to retrieve services based on exact match of the given custom *ContextID* .

username **String** Optional. Integration Server user name of the client that invoked the services that you want to retrieve.

Note: Whether Monitor treats *username* as case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

status **String** Optional. Status of the services that you want to retrieve. Use *status* if you want to retrieves services that all have a single specified status. You can specify either the keyword value for the status or the numerical value that represents the status. For a list of numerical and keyword values, see "Status Reference" on page 189.

statusSet **String List** Optional. A set of statuses for the of the services you want to retrieve. Use *statusSet* when you want retrieve services of more than one specified status. For each status that you specify in *statusSet* , specify the keyword value for the status or the numerical value that represents the status, as described above for the *status* parameter.

range **String** Optional. The date range for when the most recent log entries for the services you want to retrieved were logged. If you use this parameter, do not use the *fromDate* or *toDate* parameter. A week is Sunday through Saturday.

- `Today` Today
- `Yesterday` Yesterday
- `In the last 7 days` Within the last 7 days, including today.
- `Last week` Any day in the last calendar week.
- `This week` Any day in this calendar week.
- `Last month` Any day in the last calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

fromDate **Object** Optional. The start date of when the most recent log entries of the services you want retrieved were logged. The

getList service will get services whose log entries were logged on or after this date. If you use this parameter, use *toDate* to specify the end date; do not use the *range* parameter when you use the *fromDate* and *toDate* parameters.

toDate **Object** Optional. The end date of when the most recent log entries of the services you want retrieved were logged. The getList service will get services whose log entries were logged on or before this date. If you use this parameter, use *fromDate* to specify the start date; do not use the *range* parameter when you use the *fromDate* and *toDate* parameters.

maxRows **String** Optional. Maximum number of services to find, starting with those most recently logged. By default, the service gets all services.

isAnd **String** Optional. Whether the service is to use an AND condition or an OR condition for the criteria specified in the input parameters.

- `true` Use an AND condition. Services that match all the criteria you specify are returned. This is the default.
- `false` Use an OR condition. Services that match any of the criteria you specify are returned.

sortColumn **String** How you want the retrieved list of services sorted. This parameter works with the *sortAscending* parameter.

- `ROOTCONTEXTID` Root context ID
- `PARENTCONTEXTID` Parent context ID
- `CONTEXTID` Context ID
- `SERVICENAME` Service name
- `STATUS` Status
- `USERID`Integration Server client that invoked the service.
- `SERVERID`Integration Server that ran the service.
- `AUDITTIMESTAMP` Default. Time the most recent log entry was logged.

sortAscending **String** Optional.Whether to sort the retrieved lists of services in ascending or descending order. This parameter works with the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

Output Parameters

services **Document List** The retrieved list of services. For each service, the following fields are returned:

- *ROOTCONTEXTID* **String** Root context ID of the service.
- *PARENTCONTEXTID* **String** Context ID of the service's parent service.
- *CONTEXTID* **String** Context ID of the service.
- *SERVICENAME* **String** Name of the service.
- *SERVERID* **String**Integration Server on which the service ran.
- *USERID* **String**Integration Server user name of the client that invoked the service.
- *FIRSTSTATUS* **Number** First status of the process instance.
- *FIRSTSTATUSDECODE* **String** Status value of the first status of the service. The getList service returns the keyword value that represents the status. For the list of keyword values, for example, “Started” or “Completed,” see "Status Reference" on page 189.
- *FIRSTTIME* **Number** Time that data was last logged for the process instance, in epoch time; that is, the number of seconds since January 1, 1970.
- *FIRSTTIMESTRING* **String** Time that data was first logged for the process instance, in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where: YYYY-MM-DD is the date, hh:mm:ss:SSS is the time and zzz is the time in milliseconds.
- *LASTSTATUS* **Number** Last status of the process instance.
- *LASTSTATUSDECODE* **String** Status value of the last status of the service. The getList service returns the keyword value that represents the status. For the list of keyword values, for example, “Started” or “Completed,” see "Status Reference" on page 189.
- *LASTTIME* **Number** Time that data was last logged for the process instance, in epoch time; that is, the number of seconds since January 1, 1970.
- *LASTTIMESTRING* **String** Time that data was last logged for the process instance, in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where: YYYY-MM-DD is the date, hh:mm:ss:SSS is the time and zzz is the time in milliseconds.

- *DURATION* **String** If the service's status is 2 (Completed) or 4 (Failed), the length of time the service ran (in milliseconds).
- *RESUBMITTABLE* **String** Whether the service is resubmittable.
  - `true` Service is resubmittable.
  - `false` Service is not resubmittable.

pub.monitor.service:getListCustomData

Retrieves a list of services that meet the specified criteria, including specifying the value of a single logged field, which instructs the service to return all instances where the value you specify was logged for a specified custom logged field.

Note: Whether the search is case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

serviceName **String** Optional. Fully qualified or partial service name of services that you want to retrieve. The *serviceNameExact* parameter is used with this parameter.

serviceNameExact **String** Optional. How to match the value specified in *serviceName* .

- `true` Exactly match the *serviceName* parameter.
- `false` Default. Contains a substring that matches the *serviceName* parameter.

serverID **String** Optional. Complete or partial DNS name and port for the Integration Server that ran the service (for example, `titan.east.webmethod.com:5555`). The *serverIDExact* parameter is used with this parameter.

serverIDExact **String** Optional. How to match the value specified in *serverID* . To retrieve services executed on Integration Servers whose server IDs...

- `true` Exactly matches the *serverID* parameter.
- `false` Default. Contains a substring that matches the *serverID* parameter.

contextID **String** Optional. Complete context ID of the services to retrieve.

parentContextID **String** Optional. Complete context ID of the parent service of the services to retrieve.

rootContextID **String** Optional. Complete context ID for the root service of the services to retrieve.

customContextID **String** Optional. The full, user-defined ID for the services to retrieve. User-defined IDs are assigned by executing the pub.flow:setCustomContextID service.

username **String** Optional. Integration Server user name of the client that invoked the services to retrieve.

Note: Whether Monitor treats *username* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

status **String** Optional. Status of the services that you want to retrieve. Use *status* if you want to retrieves services that all have a single specified status. You can specify either the keyword value for the status or the numerical value that represents the status. For a list of numerical and keyword values, see "Status Reference" on page 189.

statusSet **String List** Optional. A set of statuses of the services you want to retrieve. Use *statusSet* when you want retrieve services of more than one specified status. For each status that you specify in *statusSet* , specify the keyword value for the status or the numerical value that represents the status, as described above for the *status* parameter.

range **String** Optional. The date range for when the most recent log entries for the services you want to retrieve were logged. If you use this parameter, do not use the *fromDate* or *toDate* parameter. A week is Sunday through Saturday.

- `Today` Today
- `Yesterday` Yesterday
- `In the last 7 days` Within the last 7 days, including today.
- `Last week` Any day in the last calendar week.
- `This week` Any day in this calendar week.
- `Last month` Any day in the last calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

fromDate **String** Optional. The start date of when the most recent log entries of the services you want retrieved were logged. The getList service will get services whose log entries were logged on or after this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *toDate* to specify the end date; do not use the *range* parameter when you use the *fromDate* and *toDate* parameters.

toDate **String** Optional. The end date of when the most recent log entries of the services you want retrieved were logged. The getList service will get services whose log entries were logged on or before this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *fromDate* to specify the start date; do not use the *range* parameter when you use the *fromDate* and *toDate* parameters.

maxRows **String** Optional. Maximum number of services to find, starting with those most recently logged. By default, the service gets all services.

fieldName **String** Complete name of the logged field that you want to use to search for services.

fieldType **String** The data type of *fieldName* . Specify either `String`, `Number`, or `Date`.

comparator **String** A comparator indicates how the service should compare the value you specify in *value* to the actual value logged for the custom field specified by *fieldName* . Specify one of the following: =, Contains, Not Contains, != , < , >, <= , >=

value **String** The value that you want the service to use to compare with the actual value stored for the custom field specified by *fieldName* .

isAnd **String** Optional. Whether the service is to use an AND condition or an OR condition for the specified criteria specified in the input parameters.

- `true` Use an AND condition. Services that match all the criteria you specify are returned. This is the default.
- `false` Use an OR condition. Services that match any of the criteria you specify are returned.

sortColumn **String** The column to use to sort the list of retrieved services. This parameter works with the *sortAscending* parameter. Sort by:

- `ROOTCONTEXTID` Root context ID

- `PARENTCONTEXTID` Parent context ID
- `CONTEXTID` Context ID
- `SERVICENAME` Service name
- `STATUS` Status
- `USERID`Integration Server client that invoked the service.
- `SERVERID`Integration Server that ran the service.
- `AUDITTIMESTAMP` Default. Time the most recent log entry was logged.

sortAscending **String** Optional. Whether to sort the retrieved lists of services in ascending or descending order. This parameter works with the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

Output Parameters

services **Document List** The retrieved list of services. For each service, the following fields are returned:

- *ROOTCONTEXTID* **String** Root context ID of the service.
- *PARENTCONTEXTID* **String** Context ID of the service's parent service.
- *CONTEXTID* **String** Context ID of the service.
- *AUDITTIMESTAMP* **Number** Time the service's status was logged, in epoch time; that is, the number of seconds since January 1, 1970.
- *AUDITTIMESTRING* **String** Time the service's status was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *USERID* **String**Integration Server user name of the client that invoked the service.
- *RESUBMITTABLE* **String** Whether the service is resubmittable.
  - `true` Service is resubmittable.
  - `false` Service is not resubmittable.
- *DURATION* **String** If the service's status is 2 (Completed) or 4 (Failed), the length of time the service ran (in milliseconds).
- *SERVICENAME* **String** Name of the service.

- *STATUS* **String** Status of the service. The getList service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value of the service. The getList service returns the keyword value that represents the status. For the list of keyword values, for example, "Started" or "Completed", see "Status Reference" on page 189.
- *ERRORMESSAGE* **String** If the service's status is 4 (Failed), the text of the error message.
- *SERVERID* **String**Integration Server on which the service ran.
- *CUSTOMCONTEXTID* **String** The full, user-defined ID of the retrieved service that was assigned by executing the pub.flow:setCustomContextID service.
- *AUDITEDPARENTID* **String** Complete ID of the parent service for which you have logged user-defined fields.

pub.monitor.service:getListCustomDataSet

Retrieves a list of services that meet the specified criteria, including specifying the value of multiple logged fields, which instructs the service to return all services where the value you specify was logged for multiple specified custom logged field.

Note: Whether the search is case-sensitive or case-insensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Input Parameters

serviceName **String** Optional. Fully qualified or partial service name of services that you want to retrieve. The *serviceNameExact* parameter is used with this parameter.

serviceNameExact **String** Optional. How to match the value specified in *serviceName* .

- `true` Exactly match the *serviceName* parameter.
- `false` Default. Contains a substring that matches the *serviceName* parameter.

serverID **String** Optional. Complete or partial DNS name and port for the Integration Server that ran the service (for example,

`titan.east.webmethod.com:5555`). The *serverIDExact* parameter is used with this parameter.

serverIDExact **String** Optional. How to match the value specified in *serverID* . To retrieve services executed on Integration Servers whose server IDs...

- `true` Exactly matches the *serverID* parameter.
- `false` Default. Contains a substring that matches the *serverID* parameter.

contextID **String** Optional. Complete context ID of the services to retrieve.

parentContextID **String** Optional. Complete context ID of the parent service of the services to retrieve.

rootContextID **String** Optional. Complete context ID for the root service of the services to retrieve.

customContextID **String** Optional. The full, user-defined ID for the service to retrieve. User-defined IDs are assigned by executing the pub.flow:setCustomContextID service.

username **String** Optional. Integration Server user name of the client that invoked the services that you want to retrieve.

Note: Whether Monitor treats *username* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

status **String** Optional. Status of the services that you want to retrieve. Use *status* if you want to retrieves services that all have a single specified status. You can specify either the keyword value for the status or the numerical value that represents the status. For a list of numerical and keyword values, see "Status Reference" on page 189.

statusSet **String List** Optional. A set of statuses of the services you want to retrieve. Use *statusSet* when you want retrieve services of more than one specified status. For each status that you specify in *statusSet* , specify the keyword value for the status or the numerical value that represents the status, as described above for the *status* parameter.

range **String** Optional. The date range for when the most recent log entries for the services you want to retrieve were logged. If you

use this parameter, do not use the *fromDate* or *toDate* parameter. A week is Sunday through Saturday.

- `Today` Today
- `Yesterday` Yesterday
- `In the last 7 days` Within the last 7 days, including today.
- `Last week` Any day in the last calendar week.
- `This week` Any day in this calendar week.
- `Last month` Any day in the last calendar month.
- `This month` Any day in the current calendar month.
- `Year to date` Any day in the current calendar year.

fromDate **String** Optional. The start date of when the most recent log entries of the services you want retrieved were logged. The getList service will get services whose log entries were logged on or after this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *toDate* to specify the end date; do not use the *range* parameter when you use the *fromDate* and *toDate* parameters.

toDate **String** Optional. The end date of when the most recent log entries of the services you want retrieved were logged. The getList service will get services whose log entries were logged on or before this date. Use the format *YYYY-MM-DD HH:MM:SS* . If you use this parameter, use *fromDate* to specify the start date; do not use the *range* parameter when you use the *fromDate* and *toDate* parameters.

maxRows **String** Optional. Maximum number of services to find, starting with those most recently logged. By default, the service gets all services.

customTable **String Table** The custom fields and their values that you want to use to search for services. The service returns services that match all the custom field data you specify.

> **Note:** When searching for logged service data using user-logged fields as search criteria, Monitor always uses the OR operation.

For each custom field, specify the following fields.

- *fieldName* **String** The complete name of the logged field that you want to use to search for services.
- *fieldType* **String** The data type of *fieldName* . Specify either `String`, `Number`, or `Date`.
- *comparator* **String** A comparator indicates how the service should compare the value you specify in *value* to the actual

value logged for the custom field specified by *fieldName* .Specify one of the following: = , Contains, Not Contains, != , < , >, <= , >=

- *value* **String** The value that you want the service to use to compare with the actual value stored for the custom field specified by *fieldName* .

isAnd **String** Optional. Whether the service is to use an AND condition or an OR condition for the criteria specified in the input parameters.

- `true` Use an AND condition. Services that match all the criteria you specify are returned. This is the default.
- `false` Use an OR condition. Services that match any of the criteria you specify are returned.

sortColumn **String** How you want the retrieved list of services sorted. This parameter works with the *sortAscending* parameter.

- `ROOTCONTEXTID` Root context ID
- `PARENTCONTEXTID` Parent context ID
- `CONTEXTID` Context ID
- `SERVICENAME` Service name
- `STATUS` Status
- `USERID`Integration Server client that invoked the service.
- `SERVERID`Integration Server that ran the service.
- `AUDITTIMESTAMP` Default. Time the most recent log entry was logged.

sortAscending **String** Optional.Whether to sort the retrieved lists of services in ascending or descending order. This parameter works with the *sortColumn* parameter.

- `true` Default. Sort in ascending order.
- `false` Sort in descending order.

Output Parameters

services **Document List** The retrieved list of services. For each service, the following fields are returned:

- *ROOTCONTEXTID* **String** Root context ID of the service.
- *PARENTCONTEXTID* **String** Context ID of the service's parent service.
- *CONTEXTID* **String** Context ID of the service.
- *AUDITTIMESTAMP* **Number** Time the service's status was logged, in epoch time; that is, the number of seconds since January 1, 1970.

- *AUDITTIMESTRING* **String** Time the service's status was logged in string format, YYYY-MM-DD hh:mm:ss.SSS zzz, where:
  - YYYY-MM-DD is the date
  - hh:mm:ss:SSS is the time, including milliseconds
  - zzz is the time zone
- *USERID* **String**Integration Server user name of the client that invoked the service.
- *RESUBMITTABLE* **String** Whether the service is resubmittable.
  - `true` Service is resubmittable.
  - `false` Service is not resubmittable.
- *DURATION* **String** If the service's status is 2 (Completed) or 4 (Failed), the length of time the service ran (in milliseconds).
- *SERVICENAME* **String** Name of the service.
- *STATUS* **String** Status of the service. The getList service returns the numerical value that represents the status. For a description of the status values, see "Status Reference" on page 189.
- *STATUSDECODE* **String** Status value of the service. The getList service returns the keyword value that represents the status. For the list of keyword values, for example, “Started” or “Completed,” see "Status Reference" on page 189.
- *ERRORMESSAGE* **String** If the service's status is 4 (Failed), the text of the error message.
- *SERVERID* **String**Integration Server on which the service ran.
- *CUSTOMCONTEXTID* **String** The full, user-defined ID of the retrieved service that was assigned by executing the service.
- *AUDITEDPARENTID* **String** Complete ID of the parent service for which you have logged user-defined fields.

pub.monitor.service:getPipeline

Retrieves the input pipeline for a specified service.

Input Parameters

contextID **String** Context ID of the service with the input pipeline you want to retrieve. Specify the complete, exact ID.

Note: Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

inputPipeline **Document** The retrieved input pipeline.

pub.monitor.service:isResubmittable

Checks whether a specified service is resubmittable.

Input Parameters

contextID **String** Context ID for the service to check. Specify the complete, exact ID.

Note: Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

Output Parameters

resubmittable **String** Whether the service exists in the logging database and is resubmittable.

- `true` Service exists and is resubmittable.
- `false` Service does not exist and is not resubmittable.

Usage Notes

To qualify as resubmittable, a service must be a top-level service whose input pipeline was logged in the logging database.

13 pub.monitor.serviceControl Folder

- Summary of Elements in This Folder 186
- pub.monitor.serviceControl:resubmit 186

Summary of Elements in This Folder

Service and Description
pub.monitor.serviceControl:resubmit Resubmits a specified service to a specified Integration Server.

pub.monitor.serviceControl:resubmit

Resubmits a specified service to a specified Integration Server.

Input Parameters

contextID **String** Context ID for the service to resubmit. Specify the complete, exact ID.

> **Note:** Whether Monitor treats *contextID* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

altServer **String** Optional. Integration Server to which to resubmit the service, in the format *Integration Server_host* :*Integration Server_port* . By default, the service is resubmitted to the Integration Server on which the service originally ran.

> **Note:** Whether Monitor treats *altServer* as case-sensitive depends on how the underlying database (for example, Oracle, DB2, or SQL server) handles the queries that Monitor issues to obtain data.

inputPipeline **Document** Optional. Input pipeline to resubmit. You can retrieve the pipeline using the pub.monitor.service:getPipeline service. If you do not provide *inputPipeline* , or if you do not have permission to modify the pipeline, the service retrieves the pipeline that was logged with the service in the logging database.

Output Parameters

None.

Usage Notes

You can resubmit services to any Integration Server that is defined as a remote server in the Integration Server Administrator for the local Integration Server (that is, the Integration Server on which Monitor is installed). If you want to resubmit services on the local Integration Server, that Integration Server must be defined to itself as a remote server. The remote server alias you enter in the Integration Server Administrator must be the complete DNS name for the Integration Server (for example, `titanium.east.webmethods.com`). For instructions on defining Integration Server as remote servers, see *webMethods Integration Server Administrator's Guide*.

A Status Reference

■ Statuses 190

Statuses

Many services use a numerical value to represent a status.

- When the status variable is an input parameter, use the table below to determine the numerical value to use for an input parameter based on the status you want.
- When the status variable is an output variable, use the table below to determine a status based on the numerical value in the output variable.

Value	Status
1	Started
2	Completed
4	Failed
8	Suspended
16	Waiting
32	Cached
64	Expired
128	Message
256	User Data
512	Retries Exceeded (documents and services only)
768	Unsatisfied Join
776	Interrupted
1024	Stopped
2048	Resumed
4096	Activated

Value	Status
8192	Requeued
16384	Reassigned
32768	Resubmitted
32769	Rejected (documents only)
32770	In Doubt (documents only)
32772	Retried (services only)
32777	Resubmitted - ErrorOnly (services only)
33024	Queued (tasks only)
33025	Accepted (tasks only)
33026	Completed (tasks only)
33030	Failed (Escalated)