

ORACLE®

PEOPLESOFT

PeopleTools 8.53: Reporting Web Services

October 2014

ORACLE®

PeopleTools 8.53: Reporting Web Services

CDSKU pt853pbr1_r03

Copyright © 1988, 2014, Oracle and/or its affiliates. All rights reserved.

License Restrictions Warranty/Consequential Damages Disclaimer

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

Warranty Disclaimer

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

Restricted Rights Notice

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Hazardous Applications Notice

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Trademark Notice

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

Third Party Content, Products, and Services Disclaimer

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Contents

Preface	ix
Understanding the PeopleSoft Online Help and PeopleBooks.....	ix
PeopleSoft Hosted Documentation.....	ix
Locally Installed Help.....	ix
Downloadable PeopleBook PDF Files.....	ix
Common Help Documentation.....	ix
Field and Control Definitions.....	x
Typographical Conventions.....	x
ISO Country and Currency Codes.....	xi
Region and Industry Identifiers.....	xi
Access to Oracle Support.....	xi
Documentation Accessibility.....	xii
Using and Managing the PeopleSoft Online Help.....	xii
PeopleTools Related Links.....	xii
Contact Us.....	xii
Follow Us.....	xii
Chapter 1: Getting Started with Reporting Web Services	13
Web Services Overview.....	13
Reporting Web Services Implementation.....	15
Chapter 2: Understanding Query Access Service	17
Query Access Service Operations.....	17
Query Creation Services.....	17
Query Security Services.....	18
Query Metadata Services.....	18
Query Execution Services.....	19
QAS Web Service.....	19
Understanding QAS_QRY_SERVICE.....	19
WSDL Document.....	20
Message Schema.....	21
SOAP Template.....	21
Use Cases.....	21
Creating a Query.....	21
Executing a Query.....	23
Chapter 3: Creating a Query	25
Understanding the Query Creation Process.....	25
Common Elements in Service Operations Requests.....	25
Building a Sample Query Using QAS.....	26
Selecting Records.....	26
QAS_RECORDS_OPER.....	26
QAS_RECORD_DEFN_OPER.....	28
QAS_HIERARCHY_RECORDS_OPER.....	30
QAS_RELATED_RECORDS_OPER.....	31
Adding Fields to Query Content.....	32
QAS_FIELDS_OPER.....	32
QAS_FIELD_PROPS_OPER.....	33
Defining Selection Criteria.....	36
QAS_TREES_OPER.....	36

QAS_TREE_DETAILS_OPER.....	37
Deleting a Query.....	39
QAS_QUERY_DELETE_OPER.....	39
Saving a Query.....	40
QAS_SAVE_QUERY_OPER.....	40
Simple Query Example.....	50
Query Using Prompts Example.....	51
Related Join Query Example.....	53
Related Left Outer Join Query Example.....	55
Hierarchy Join Query Example.....	56
Query with an Aggregate Value Example.....	57
Query with Expression Example.....	58
Subquery Example.....	60
Chapter 4: Executing a Query.....	63
Understanding Query Execution.....	63
Query Execution Models.....	63
Output Format and Output Type.....	64
Report Repository.....	65
WebRowSet Format.....	65
XMLP Format.....	67
Excel Format.....	67
HTML Format.....	68
Date and Time Handling.....	68
Execution Logging.....	69
Selecting a Query.....	69
QAS_LISTQUERY_OPER.....	69
QAS_QUERY_DETAILS_OPER.....	71
Listing Fields.....	77
QAS_LISTQUERYFIELDS.....	77
Entering Prompts.....	79
QAS_LISTQUERYPROMPTS_OPER.....	79
QAS_GETPROMPTTABLEVALUES_OPER.....	82
QAS_GETXLAT_OPER.....	83
Executing the Query.....	84
XML String.....	84
QAS_EXECUTEQRYSYNC_OPER.....	88
QAS_EXECUTEQRYSYNCPOLL_OPER.....	94
QAS_EXECUTEQRYASYNC_OPER.....	97
Retrieving Query Results.....	100
QAS_GETQUERYRESULTS_OPER.....	100
Canceling a Query.....	102
QAS_CANCELQUERY_OPER.....	102
Retrieving Query Status.....	103
QAS_QUERYSTATUS_OPER.....	103
Chapter 5: Accessing PeopleSoft Application Tables.....	105
Understanding QAS Security.....	105
Query Security.....	105
Service Operation Security.....	105
WS-Security.....	105
Process Profile.....	106
QAS Security Flow.....	106

QAS Security Service Operations.....	107
QAS_AUTHTOKEN_OPER.....	107
QAS_LISTROLE_OPER.....	108
QAS_LISTUSERROLES_OPER.....	109
QAS_LISTUSER_OPER.....	110
QAS_LISTROLEUSERS_OPER.....	111
QAS_LOGIN_OPER.....	112
Using QAS Administration.....	113
Chapter 6: Understanding Process Scheduler Service.....	115
Process Scheduler Service Operations.....	115
Process Scheduler Service.....	116
Understanding PROCESSREQUEST Service.....	117
WSDL Document.....	118
Message Schema.....	118
SOAP Template.....	118
Process Scheduler Service Operation Security.....	118
Process Scheduler Web Service Security.....	118
Chapter 7: Using Process Scheduler Service Operations.....	121
Understanding Process Scheduler Service Operations.....	121
Guidelines for Completing Service Operation Requests.....	121
Scheduling a Request.....	121
PRCS_GETPROMPT.....	122
PRCS_GETPROCESSNAMES.....	123
PRCS_GETPARAMS.....	124
PRCS_SCHEDULE.....	125
PRCS_SCHEDULE_PSQUERY.....	132
PRCS_SCHEDULE_CONQUERY.....	134
PRCS_SCHEDULE_BIPUBLISHER.....	135
PRCS_RUNJOBSETNOW.....	136
Monitoring a Request.....	137
PRCS_FINDREQUESTS.....	138
PRCS_GETREQUEST.....	142
PRCS_UPDATEREQUEST.....	143
PRCS_GETREPORT.....	144
PRCS_GETSERVERCONSTRAINTS.....	146
PRCS_GETOUTPUTTYPE.....	148
PRCS_GETOUTPUTFORMATS.....	150
PRCS_PRINT_JOB_HTML_REPORT.....	152
PRCS_PRINT_JOB_REQUEST_REPORT.....	153
PRCS_PRINT_SCHEDULE_HTML_REPORT.....	155

Preface

Understanding the PeopleSoft Online Help and PeopleBooks

The PeopleSoft Online Help is a website that enables you to view all help content for PeopleSoft Applications and PeopleTools. The help provides standard navigation and full-text searching, as well as context-sensitive online help for PeopleSoft users.

PeopleSoft Hosted Documentation

You access the PeopleSoft Online Help on Oracle's PeopleSoft Hosted Documentation website, which enables you to access the full help website and context-sensitive help directly from an Oracle hosted server. The hosted documentation is updated on a regular schedule, ensuring that you have access to the most current documentation. This reduces the need to view separate documentation posts for application maintenance on My Oracle Support, because that documentation is now incorporated into the hosted website content. The Hosted Documentation website is available in English only.

Locally Installed Help

If your organization has firewall restrictions that prevent you from using the Hosted Documentation website, you can install the PeopleSoft Online Help locally. If you install the help locally, you have more control over which documents users can access and you can include links to your organization's custom documentation on help pages.

In addition, if you locally install the PeopleSoft Online Help, you can use any search engine for full-text searching. Your installation documentation includes instructions about how to set up Oracle Secure Enterprise Search for full-text searching.

See *PeopleTools 8.53 Installation* for your database platform, "Installing PeopleSoft Online Help." If you do not use Secure Enterprise Search, see the documentation for your chosen search engine.

Note: Before users can access the search engine on a locally installed help website, you must enable the Search portlet and link. Click the Help link on any page in the PeopleSoft Online Help for instructions.

Downloadable PeopleBook PDF Files

You can access downloadable PDF versions of the help content in the traditional PeopleBook format. The content in the PeopleBook PDFs is the same as the content in the PeopleSoft Online Help, but it has a different structure and it does not include the interactive navigation features that are available in the online help.

Common Help Documentation

Common help documentation contains information that applies to multiple applications. The two main types of common help are:

- Application Fundamentals

- Using PeopleSoft Applications

Most product lines provide a set of application fundamentals help topics that discuss essential information about the setup and design of your system. This information applies to many or all applications in the PeopleSoft product line. Whether you are implementing a single application, some combination of applications within the product line, or the entire product line, you should be familiar with the contents of the appropriate application fundamentals help. They provide the starting points for fundamental implementation tasks.

In addition, the *PeopleTools: PeopleSoft Applications User's Guide* introduces you to the various elements of the PeopleSoft Pure Internet Architecture. It also explains how to use the navigational hierarchy, components, and pages to perform basic functions as you navigate through the system. While your application or implementation may differ, the topics in this user's guide provide general information about using PeopleSoft Applications.

Field and Control Definitions

PeopleSoft documentation includes definitions for most fields and controls that appear on application pages. These definitions describe how to use a field or control, where populated values come from, the effects of selecting certain values, and so on. If a field or control is not defined, then it either requires no additional explanation or is documented in a common elements section earlier in the documentation. For example, the Date field rarely requires additional explanation and may not be defined in the documentation for some pages.

Typographical Conventions

The following table describes the typographical conventions that are used in the online help.

<i>Typographical Convention</i>	<i>Description</i>
Key+Key	Indicates a key combination action. For example, a plus sign (+) between keys means that you must hold down the first key while you press the second key. For Alt+W, hold down the Alt key while you press the W key.
... (ellipses)	Indicate that the preceding item or series can be repeated any number of times in PeopleCode syntax.
{ } (curly braces)	Indicate a choice between two options in PeopleCode syntax. Options are separated by a pipe ().
[] (square brackets)	Indicate optional items in PeopleCode syntax.
& (ampersand)	When placed before a parameter in PeopleCode syntax, an ampersand indicates that the parameter is an already instantiated object. Ampersands also precede all PeopleCode variables.
⇒	This continuation character has been inserted at the end of a line of code that has been wrapped at the page margin. The code should be viewed or entered as a single, continuous line of code without the continuation character.

ISO Country and Currency Codes

PeopleSoft Online Help topics use International Organization for Standardization (ISO) country and currency codes to identify country-specific information and monetary amounts.

ISO country codes may appear as country identifiers, and ISO currency codes may appear as currency identifiers in your PeopleSoft documentation. Reference to an ISO country code in your documentation does not imply that your application includes every ISO country code. The following example is a country-specific heading: "(FRA) Hiring an Employee."

The PeopleSoft Currency Code table (CURRENCY_CD_TBL) contains sample currency code data. The Currency Code table is based on ISO Standard 4217, "Codes for the representation of currencies," and also relies on ISO country codes in the Country table (COUNTRY_TBL). The navigation to the pages where you maintain currency code and country information depends on which PeopleSoft applications you are using. To access the pages for maintaining the Currency Code and Country tables, consult the online help for your applications for more information.

Region and Industry Identifiers

Information that applies only to a specific region or industry is preceded by a standard identifier in parentheses. This identifier typically appears at the beginning of a section heading, but it may also appear at the beginning of a note or other text.

Example of a region-specific heading: "(Latin America) Setting Up Depreciation"

Region Identifiers

Regions are identified by the region name. The following region identifiers may appear in the PeopleSoft Online Help:

- Asia Pacific
- Europe
- Latin America
- North America

Industry Identifiers

Industries are identified by the industry name or by an abbreviation for that industry. The following industry identifiers may appear in the PeopleSoft Online Help:

- USF (U.S. Federal)
- E&G (Education and Government)

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Using and Managing the PeopleSoft Online Help

Click the Help link in the universal navigation header of any page in the PeopleSoft Online Help to see information on the following topics:

- What's new in the PeopleSoft Online Help.
 - PeopleSoft Online Help accessibility.
 - Accessing, navigating, and searching the PeopleSoft Online Help.
 - Managing a locally installed PeopleSoft Online Help website.
-

PeopleTools Related Links

[Oracle's PeopleSoft PeopleTools 8.53 Documentation Home Page \[ID 1494462.1\]](#)

[PeopleSoft Information Portal on Oracle.com](#)

[My Oracle Support](#)

[PeopleSoft Training from Oracle University](#)

[PeopleSoft Video Feature Overviews on YouTube](#)

Contact Us

[Send us your suggestions](#) Please include release numbers for the PeopleTools and applications that you are using.

Follow Us



Get the latest PeopleSoft updates on [Facebook](#).



Follow PeopleSoft on [Twitter@PeopleSoft_Info](#).

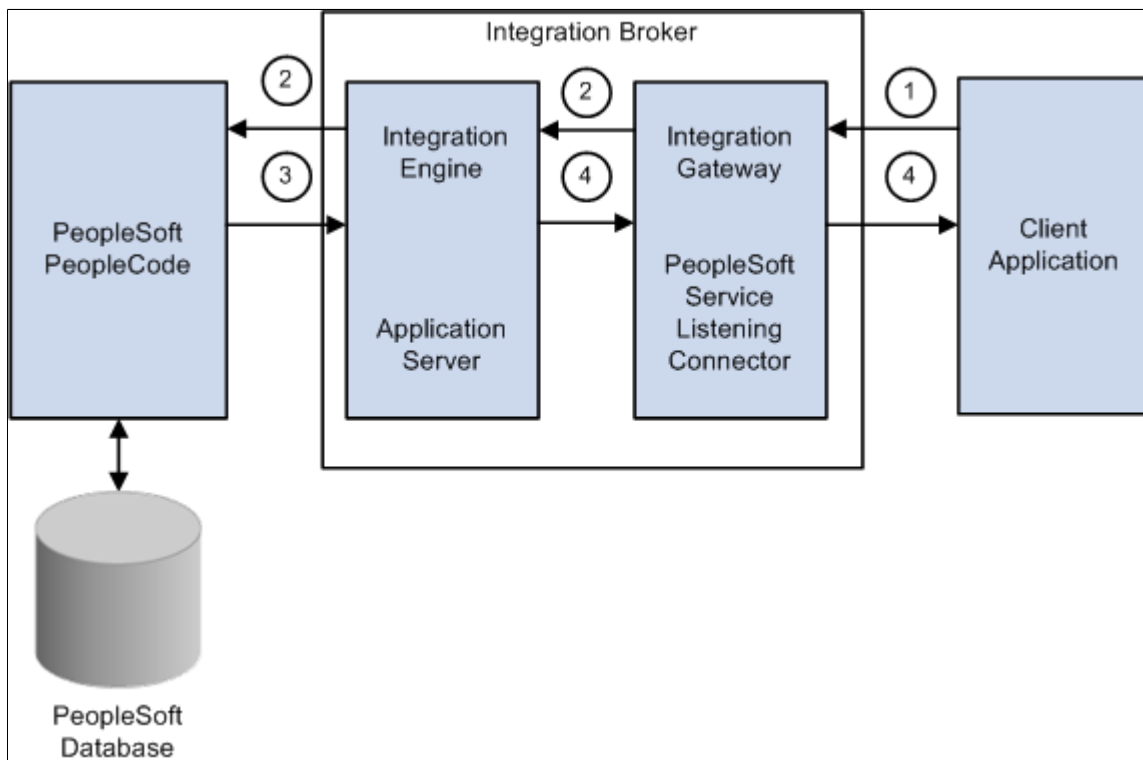
Getting Started with Reporting Web Services

Web Services Overview

A web service provides external applications a web-services-based means of accessing PeopleSoft data. Web services are implemented through the PeopleTools Integration Broker (IB) framework. The Integration Gateway web application receives all the web service requests and forwards them to the Integration Engine (application server) for processing.

Image: Using Reporting Web Services with client application

This diagram illustrates how the web service is used with external applications.



This process occurs when you use a reporting web service with a client application:

1. The client application invokes one of the service operations from the API (application programming interface) published by PeopleSoft application.
2. PeopleSoft Integration Broker receives the service operation request and validates the WS security credentials.

3. The request is passed to Application Server for processing. The Application Server authenticates the service operation and routes it to the respective handler. The handler executes the PeopleCode and sends the response to Integration Gateway.
4. Integration Broker sends the response to the client application.

WS-Security

Web services security (WS-Security) is implemented on the integration gateway for inbound and outbound integrations with third-party systems. WS-Security adds a layer of security to sending and receiving service operations by adding a UsernameToken that identifies the sender and authenticates its identity to the web service provider. On inbound processing, PeopleSoft Integration Broker can process requests received from integration partners that contain WS-Security UsernameToken and passwords in the SOAP (Simple Object Access Protocol) header of the inbound SOAP request. The user name and password should be encrypted via PKI (public key infrastructure).

Reporting Web Services are delivered as restricted services, which means that sensitive fields of the service definition and of associated service operations appear in read-only mode. Each service operation is delivered with WS Security Req (required) Verification set to *Encrypt and Digitally Sign or HTTPS*.

See "Implementing Web Services Security" (PeopleTools 8.53: PeopleSoft Integration Broker Administration).

As delivered, reporting web service operations must be either encrypted and digitally signed or sent over HTTPS. Customers can change the security settings if desired.

Integration Broker (IB) authenticates the request in the following way:

Message Sent Over	Header Attributes	Result
HTTPS	PS-Token is present in the HTTP header.	IB authenticates message.
HTTPS	User ID and password are present in the SOAP Header.	IB authenticates message.
HTTPS	Message is encrypted and digitally signed.	IB authenticates message.
HTTP	PS-Token is present in the HTTP header.	IB rejects message.
HTTP	User ID and password are present in the SOAP header.	IB rejects message.
HTTP	Message is encrypted and digitally signed.	IB authenticates message.

Reporting web service operations require a user ID and password. If a user ID and password are not supplied in the SOAP header, Integration Broker rejects the request.

If the request is received from another PeopleSoft system, the user ID associated with the requesting PeopleSoft node is used when the SOAP header does not specify a user name.

When a request is received, PeopleSoft Integration Broker validates the user ID and password in the SOAP header to determine whether the user has the proper security to invoke the service operation requested.

Reporting Web Services Implementation

Before you can deploy a Reporting Web Service, you must configure Integration Broker:

Step	Reference
1. Configure Integration Gateway.	See "Understanding PeopleSoft Integration Broker" (PeopleTools 8.53: PeopleSoft Integration Broker).
2. Set Service Configuration Target URL.	See "Using the Service Configuration Page to Set Service Configuration Properties" (PeopleTools 8.53: PeopleSoft Integration Broker Administration).
3. Export the WSDL to the client application.	QAS_QRY_SERVICE is the service for Query Access Manager. PROCESSREQUEST is the service for Process Scheduler.

Chapter 2

Understanding Query Access Service

Query Access Service Operations

To enable external applications to access PeopleSoft Query, a number of service operations are available. Depending on the application, the external application will need to use several service operations in combination. The service operations for QAS are part of the service QAS_QRY_SERVICE.

QAS service operations are grouped into the following categories to help describe the service operations:

- Query Creation Services (QCS)
- Query Security Services (QSS)
- Query Metadata Services (QMS)
- Query Execution Services (QES)

Note: No processing or security is dependent on the categories; they are simply groupings for describing the service operations.

Query Creation Services

Query creation services are used by client applications to create and save a new query. This table lists the Query Creation Services:

Service Operation Name	Description
QAS_RECORDS_OPER	Returns the list of records, along with descriptions, that are accessible to the user.
QAS_HIERARCHY_RECORDS_OPER	Returns the list of hierarchy records, if any, along with their descriptions.
QAS_RELATED_RECORDS_OPER	Returns the list of related records of all the fields in the record.
QAS_RECORD_DEFN_OPER	Returns the definition of a given record, including field descriptions and key information.
QAS_TREES_OPER	Returns the list of PeopleSoft trees that are accessible to the user.
QAS_TREE_DETAILS_OPER	Returns the tree details, including setID, effective date, and the list of all the nodes.
QAS_FIELDS_OPER	Returns the fields from all records that are accessible to the user.

Service Operation Name	Description
QAS_FIELD_PROPS_OPER	Returns the properties of a given field.
QAS_QUERY_DETAILS_OPER	Returns the complete details of an existing query in XML format.
QAS_SAVE_QUERY_OPER	Validates and verifies XML-format query and saves the query in the PeopleSoft database.
QAS_QUERY_DELETE_OPER	Deletes the query from the PeopleSoft database.

Query Security Services

Query security services are used to identify users and roles. This table lists the Query Security Services:

Service Operation Name	Description
QAS_AUTHTOKEN_OPER	Returns the user ID based on the PS Token passed in.
QAS_LISTROLE_OPER	Returns the list of roles, along with descriptions.
QAS_LISTUSER_OPER	Returns the list of users, along with descriptions.
QAS_LISTROLEUSERS_OPER	Returns the list of users for a given role, along with descriptions.
QAS_LISTUSERROLES_OPER	Returns the list of roles for a given user, along with descriptions.
QAS_LOGIN_OPER	Enables the client application to sign on to the PeopleSoft database.

Query Metadata Services

Query metadata services are used to extract application data from the PeopleSoft database. This table lists the Query Metadata Services:

Service Operation Name	Description
QAS_LISTQUERY_OPER	Returns the list of queries, along with the query descriptions, and owner type.
QAS_LISTQUERYPROMPTS_OPER	Returns a list of prompts used in a specific query.
QAS_LISTQUERYFIELDS_OPER	Returns a list of fields for a given query.
QAS_GETXLAT_OPER	Returns the translate value for a given field.
QAS_GETPROMPTTABLEVALUES_OPER	Returns a list of field values for a given prompt.

Query Execution Services

Query Execution Services are used to run the query from a third-party application. This table lists service operations used to execute a query:

Service Operation Name	Description
QAS_CANCELQUERY_OPER	Returns the status of query cancellation.
QAS_QUERYSTATUS_OPER	Returns query execution status.
QAS_EXECUTEQRYSYNC_OPER	Returns query result.
QAS_EXECUTEQRYASYNC_OPER	Returns query result.
QAS_EXECUTEQRYSYNCPOLL_OPER	Runs the query through process scheduler and returns the query instance ID. The service operation QAS_GETQUERYRESULTS_OPER is used to retrieve the results.
QAS_GETQUERYRESULTS_OPER	Used to return the query results when the query is executed on the PeopleSoft database using QAS_EXECUTEQRYSYNCPOLL_OPER.

QAS Web Service

This section provides an overview of the QAS web service named QAS_QRY_SERVICE and discusses:

- WSDL (Web Services Description Language) document
- Message schema
- SOAP template

Understanding QAS_QRY_SERVICE

Services are used to logically group a set of service operations. The service QAS_QRY_SERVICE contains all of the service operations for QAS. When you publish a web service, a WSDL document is created containing all of the selected service operations as well as the endpoints based on the service configuration.

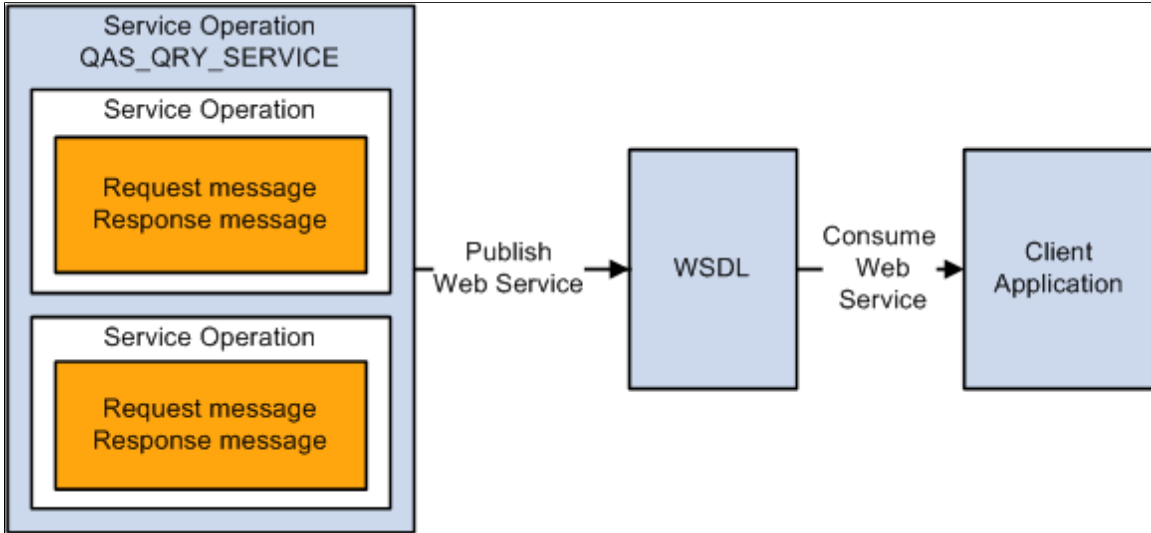
QAS_QRY_SERVICE is delivered as a restricted service. Only a user with the Peoplesoft Administrator role can make changes to the restricted state.

See "Using the Service Configuration Page to Set Service Configuration Properties" (PeopleTools 8.53: PeopleSoft Integration Broker Administration).

See "Restricting and Enabling Write Access to Service Definitions" (PeopleTools 8.53: PeopleSoft Integration Broker).

Image: Providing a web service

This diagram illustrates how a client application consumes the web service



The published web service is stored in the WSDL Repository. Client applications consume the WSDL by accessing the WSDL URL from the repository. The WSDL for QAS_QRY_SERVICE is delivered and updated with the appropriate WSDL Repository location whenever the target location is updated on the Service Configuration page (Select PeopleTools, Integration Broker, Configuration, Service Configuration).

See "Using the Service Configuration Page to Set Service Configuration Properties" (PeopleTools 8.53: PeopleSoft Integration Broker Administration).

WSDL Document

To view the generated WSDL Repository for the QAS_QRY_SERVICE, click the View WSDL link on the service page (select PeopleTools, Integration Broker, Integration Setup, Services).

Image: WSDL Repository page

This example illustrates the fields and controls on the WSDL Repository page.

WSDL Repository				
Service: QAS_QRY_SERVICE				
WSDL				
WSDL: QAS_QRY_SERVICE.1 Default Last Upd DTTm: 04/07/2009 8:02:17PM				
Exported Service Operations				
Operation	Routing External Alias	Request Message	Response Message	Fault Message
QAS_AUTHTOKEN_OPER	QAS_AUTHTOKEN_OPER.VERSION_1	QAS_AUTHTOKEN_REQ_MSG.VERSION_1	QAS_AUTHTOKEN_RESP_MSG.VERSION_1	
QAS_CANCELQUERY_OPER	QAS_CANCELQUERY.VERSION_1	QAS_CANCELQUERY_REQ_MSG.VERSION_1	QAS_CANCELQUERY_RESP_MSG.VERSION_1	
QAS_EXECUTEQRYSYNC_OPER	QAS_EXECUTEQRYSYNC_OPER.VERSION_1	QAS_EXEQRY_ASYNC_REQ_MSG.VERSION_1		
QAS_EXECUTEQRYSYNCPOLL_OPER	QAS_EXECUTEQRYSYNCPOLL_OPER.VERSION_1	QAS_EXEQRY_SYNCPOLL_REQ_MSG.VERSION_1	QAS_EXEQRYSYNCPOLL_RESP_MSG.VERSION_1	
QAS_EXECUTEQRYSYNC_OPER	QAS_EXECUTEQRYSYNC.VERSION_1	QAS_EXEQRY_SYNC_REQ_MSG.VERSION_1	QAS_GETQUERYRESULTS_RESP_MSG.VERSION_1	

All of the service operations contained in the WSDL are listed in the Exported Service Operations section. Use the View WSDL link to view the WSDL document.

Message Schema

The WSDL document is created using the request and response message schemas for each service operation. To view the message schema, select PeopleTools, Integration Broker, Integration Setup, Messages and select the message to view. Select the Schema tab to view the schema.

SOAP Template

For each service operation, you can use the Generate SOAP Template utility to generate SOAP message templates for request messages, response messages, and fault messages found in the WSDL document. The generated SOAP template will include WS-Security SOAP Header tags. You can then use the templates to test SOAP messages in your third-party application.

Note: WS-Security should be used with all QAS service operations.

See "Implementing Web Services Security" (PeopleTools 8.53: PeopleSoft Integration Broker Administration).

Use Cases

This section discusses the two most common use cases for QAS::

- Create a Query
- Execute a Query

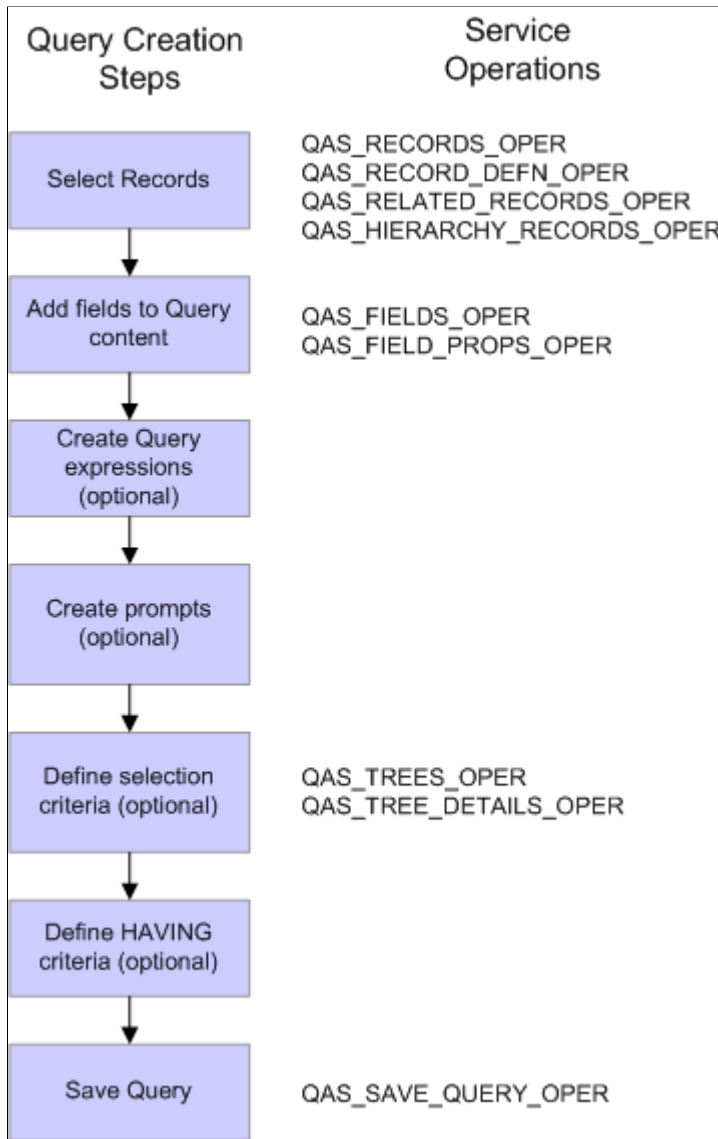
Creating a Query

QAS provides several service operations that can be used by an external application to access the PeopleSoft application database to select records and fields to create a valid PeopleSoft Query. Creating

a query involves several steps, some of which do not require access to the PeopleSoft records, fields, and trees and therefore do not have associated service operations.:

Image: Steps to create a query

This diagram illustrates the steps and corresponding service operation that can be used to access the PeopleSoft database information.



You can use QAS_RECORDS_OPER, QAS_RECORD_DEFN_OPER, QAS_RELATED_RECORDS_OPER, and QAS_HIERARCHY_RECORDS_OPER to select records.

QAS_FIELDS_OPER and QAS_FIELD_PROPS_OPER can be used to help select the fields and prompts to add to the query content.

QAS_TREE_OPER and QAS_TREE_DETAILS_OPER can be used when your selection criteria uses trees.

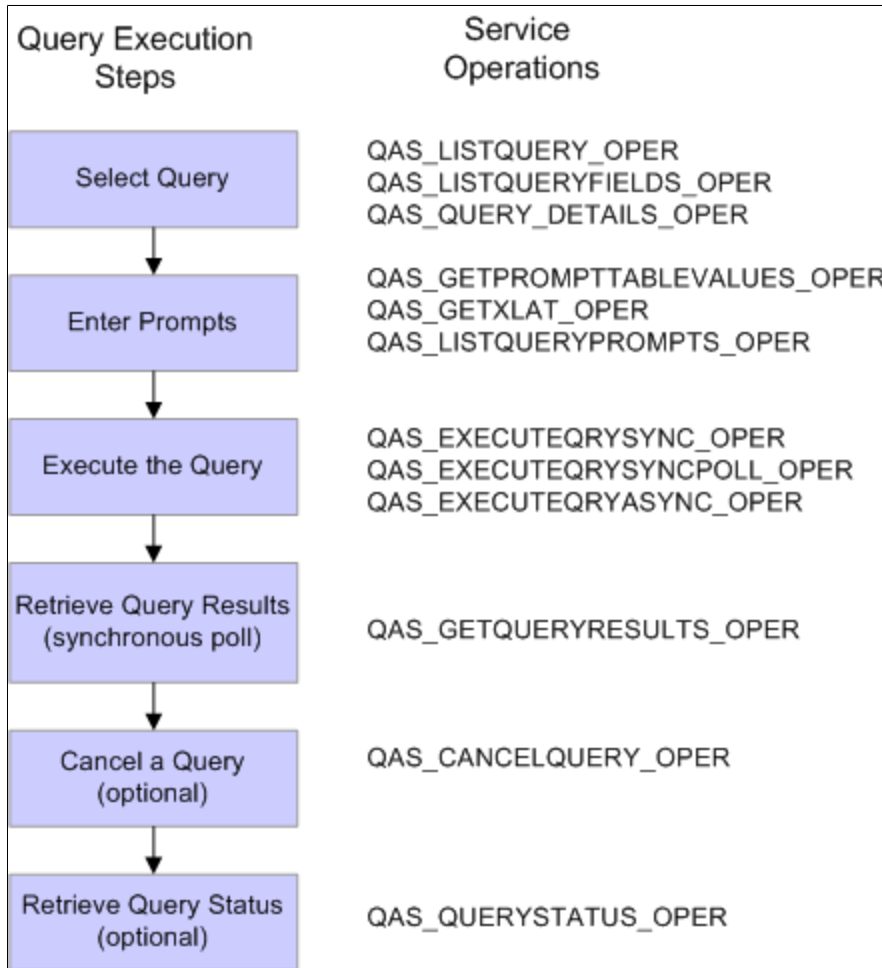
The QAS_SAVE_QUERY_OPER is used to save a query definition to the PeopleSoft database.

Executing a Query

To execute an existing query in the PeopleSoft database, the external application will use Query Metadata Services, Query Execution Services, and possibly the Query Result Service. This diagram illustrates the steps and corresponding service operations that can be used to execute a PeopleSoft query and retrieve the results:

Image: Steps to execute a query

This diagram illustrates the steps and corresponding service operations that can be used to execute a PeopleSoft query and retrieve the results



To execute a query, you will need to know the query name. QAS_LISTQUERY_OPER, QAS_LISTQUERYFIELDS_OPER, and QAS_QUERY_DETAILS_OPER can be used to help select the query name.

For queries containing prompts, use QAS_GETPROMPTTABLEVALUES_OPER, QAS_GETXLAT_OPER, and QAS_LISTQUERYPROMPTS_OPER to identify the prompts and obtain a list of values.

You can also filter fields to be returned in the query results using QAS_LISTQUERYFIELDS_OPER.

You can select how you want to execute the query and use the appropriate service operation.

For synchronous poll execution, you will use QAS_GETQUERYRESULTS_OPER to retrieve the results.

You can invoke `QAS_CANCELQUERY_OPER` after obtaining the query result to clean up the `PSQASRUN` and `IB` tables.

You can check the query status using `QAS_QUERYSTATUS_OPER`.

Chapter 3

Creating a Query

Understanding the Query Creation Process

A third-party application will create a PeopleSoft query using the same basic steps as an online PeopleSoft user. In the process of creating the query, you will need to integrate with the PeopleSoft system to access records and fields, as well as save the query in the PeopleSoft database. This table lists the steps to create and save the query.

Steps	Description
1. Select records.	QAS service operations are available to select the records from the PeopleSoft database. Query security is enforced and only records available to the user ID provided in the request will be returned.
2. Add fields to the query.	QAS service operations are available to select fields and field properties from the PeopleSoft database.
3. Create query expressions (optional).	The third-party application will create any expressions necessary for the query.
4. Create Prompts (optional).	The third-party application will create any prompts necessary for the query.
5. Define selection criteria (optional).	The third-party application will create the selection criteria necessary for the query. QAS provides service operations for accessing tree information that can be used in the selection criteria.
6. Define HAVING criteria (optional).	The third-party application will create any HAVING criteria necessary for the query.
7. Save the query.	Once the third-party has defined the query, the QAS service operation for saving the query is used to save the query in the PeopleSoft database.

Note: Creating connected queries is not supported in QAS.

Note: Updating queries is not supported in QAS.

Common Elements in Service Operations Requests

The following guidelines apply across all QAS requests:

- All requests should contain a SOAP header containing user ID and password.

- All search strings use the Begins with criterion.
- Parameters are case-insensitive, unless specifically noted or specific enumeration values are listed.
- All elements are required, unless specifically noted. Even if no value is necessary, the element must be included in the request.
- Values are required in the element, unless specifically noted.
- Leading and trailing blanks are not allowed in XML values.

Note: All examples are shown without the SOAP header. For information about the SOAP header, refer to “Accessing PeopleSoft Application Tables.”

Related Links

[WS-Security](#)

Building a Sample Query Using QAS

These topics includes a sample request and response message for each of the service operations you will use to build a query. Most of the examples will use message catalog entries. The tables are:

- PSMSGSETDEFN – Message Sets
- PSMSGCATDEFN – Message Catalog

Selecting Records

This section describes the service operations that are available for selecting records. These include:

- QAS_RECORDS_OPER
- QAS_RECORD_DEFN_OPER
- QAS_HIERARCHY_RECORDS_OPER
- QAS_RELATED_RECORDS_OPER

QAS_RECORDS_OPER

Use this service operation to select a list of records, along with the descriptions that you can access.

Request Message: QAS_RECORDS_REQ_MSG

<i>Element Name</i>	<i>Description</i>
SEARCH_STRING Required element	<p>Search string used for specifying the name or the first few characters of the search criterion definition name.</p> <p>If no value is entered, a list of all accessible records will be returned.</p> <hr/> <p>Note: For field names, QAS does not interpret the underscore as a wildcard. For example, if you enter <i>NODE_andFieldName</i> for the SEARCH_CRITERION, QAS will retrieve records that contain fields such as <i>NODE_TYPE</i> and not <i>NODECOUNT</i>.</p>
SEARCH_CRITERION Required element	<p>Valid search criteria are:</p> <ul style="list-style-type: none"> • RecordName • FieldName • Description <p>In the SEARCH_STRING element, enter the record name, field name, or description. Record name is the default criterion.</p>

Example Request:

This request will return a list of records that start with *PSMSGC*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_RECORDS_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_RECORDS_REQ_MSG>
      <SEARCH_STRING>PSMSGC</SEARCH_STRING>
      <SEARCH_CRITERION>RecordName</SEARCH_CRITERION>
    </qas:QAS_RECORDS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_RECORDS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
RECORD_NAME	Record name.
RECORD_DESCRIPTION	Record description.
INFORMATION_MESSAGE	Returns information about the request. For example if the criterion in the request is incorrect or if no records meet the criteria, a message indicating the error is returned.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_RECORDS_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterprise/Tools=
/schemas/QAS_RECORDS_RESP_MSG.VERSION_1">
      <RECORD>
        <RECORD_NAME>PSMSGCATDEFN</RECORD_NAME>
        <RECORD_DESCRIPTION>Message Catalog</RECORD_DESCRIPTION>
      </RECORD>
    </qcs:QAS_RECORDS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

QAS_RECORD_DEFN_OPER

Use this service operation to return the definition of a given record, including field descriptions and key information.

Request Message: QAS_RECORD_DEFN_REQ_MSG

<i>Element Name</i>	<i>Description</i>
RECORD_NAME	Complete record name. Required.
Required element	

Example Request:

This request will return the record definition for *PSMSGCATDEFN*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q=
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_RECORD_DEFN_REQ_MSG.VERISO=
N_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_RECORD_DEFN_REQ_MSG>
      <RECORD_NAME>psmsgcatdefn</RECORD_NAME>
    </qas:QAS_RECORD_DEFN_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_RECORD_DEFN_RESP_MSG

<i>Element Name</i>	<i>Description</i>
RECORD_NAME	Record name.
FIELD_NAME	Field name.
FIELD_DESCRIPTION	Field description.
KEY	Indicates whether this field is a key; value will be <i>Yes</i> or <i>No</i> .

Element Name	Description
FIELD_TYPE	Field type.
FIELD_LENGTH	Field length.
FIELD_LOOKUP_TABLE	Prompt table name, if applicable.
FIELD_EDIT_TYPE	<p>Field edit type is a 32-bit binary number. Counting from 1 from the right, a 1 in bit 10 indicates translate values, a 1 in bit 14 indicates a yes/no table, and a 1 in bit 15 indicates a prompt table. Client applications need to do bit-wise AND on this value.</p> <p>For example, the value 2337 represents:</p> <ul style="list-style-type: none"> • Key = 1 • Search item = 2048 • List item = 32 • Required = 256 <p>See "Flag" (PeopleTools 8.53: PeopleCode API Reference).</p>
FIELD_SHORT_NAME	Short name.
FIELD_LONG_NAME	Long name.
INFORMATION_MESSAGE	Returns information about the request. For example if the criterion in the request is incorrect or if no records meet the criteria, a message indicating the error is returned.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_RECORD_DEFN_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterprise/T
ools/schemas/QAS_RECORD_DEFN_RESP_MSG.VERSION_1">
      <RECORD_NAME>PSMSGCATDEFN</RECORD_NAME>
      <FIELD>
        <FIELD_NAME>MESSAGE_SET_NBR</FIELD_NAME>
        <FIELD_DESCRIPTION>Message Set Number</FIELD_DESCRIPTION>
        <KEY>Yes</KEY>
        <FIELD_TYPE>Number</FIELD_TYPE>
        <FIELD_LENGTH>5</FIELD_LENGTH>
        <FIELD_LOOKUP_TABLE/>
        <FIELD_EDIT_TYPE>2337</FIELD_EDIT_TYPE>
        <FIELD_SHORT_NAME>Set</FIELD_SHORT_NAME>
        <FIELD_LONG_NAME>Message Set Number</FIELD_LONG_NAME>
      </FIELD>
      <FIELD>
        <FIELD_NAME>MESSAGE_NBR</FIELD_NAME>
        <FIELD_DESCRIPTION>Message Number</FIELD_DESCRIPTION>
        <KEY>Yes</KEY>
      </FIELD>
    </qcs:QAS_RECORD_DEFN_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        <FIELD_TYPE>Number</FIELD_TYPE>
        <FIELD_LENGTH>5</FIELD_LENGTH>
        <FIELD_LOOKUP_TABLE/>
        <FIELD_EDIT_TYPE>2337</FIELD_EDIT_TYPE>
        <FIELD_SHORT_NAME>Msg</FIELD_SHORT_NAME>
        <FIELD_LONG_NAME>Message Number</FIELD_LONG_NAME>
    </FIELD>
    <FIELD>
        <FIELD_NAME>MESSAGE_TEXT</FIELD_NAME>
        <FIELD_DESCRIPTION>Message Text</FIELD_DESCRIPTION>
        <KEY>No</KEY>
        <FIELD_TYPE>Character</FIELD_TYPE>
        <FIELD_LENGTH>100</FIELD_LENGTH>
        <FIELD_LOOKUP_TABLE/>
        <FIELD_EDIT_TYPE>33554720</FIELD_EDIT_TYPE>
        <FIELD_SHORT_NAME>Message</FIELD_SHORT_NAME>
        <FIELD_LONG_NAME>Message Text</FIELD_LONG_NAME>
    </FIELD>
    . . . . .
    </qcs:QAS_RECORD_DEFN_RESP_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

QAS_HIERARCHY_RECORDS_OPER

Use this service operation to return the list of hierarchy records, if any, along with their descriptions for the requested record.

Request Message: QAS_HIERARCHY_RECORDS_REQ_MSG

<i>Element Name</i>	<i>Description</i>
RECORD_NAME Required element	Complete record name. Required.

Example Request:

This request will return a list of all hierarchy records for *PSMSGCATDEFN*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_HIERARCHY_RECORDS_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_HIERARCHY_RECORDS_REQ_MSG>
      <RECORD_NAME>PSMSGCATDEFN</RECORD_NAME>
    </qas:QAS_HIERARCHY_RECORDS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_HIERARCHY_RECORDS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
RECORD_NAME	Record name.
RECORD_DESCRIPTION	Record description.

Element Name	Description
INFORMATION_MESSAGE	Returns information about the request. For example if the criterion in the request is incorrect or if no records meet the criteria, a message indicating the error is returned.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_HIERARCHY_RECORDS_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterp
rise/Tools/schemas/QAS_HIERARCHY_RECORDS_RESP_MSG.VERSION_1">
      <RECORD>
        <RECORD_NAME>PSMSGSETDEFN</RECORD_NAME>
        <RECORD_DESCRIPTION>Message Sets</RECORD_DESCRIPTION>
      </RECORD>
      <RECORD>
        <RECORD_NAME>PSMSGCATDEFN</RECORD_NAME>
        <RECORD_DESCRIPTION>Message Catalog</RECORD_DESCRIPTION>
      </RECORD>
      <RECORD>
        <RECORD_NAME>MSG_CAT_VW</RECORD_NAME>
        <RECORD_DESCRIPTION>Message Catalog Trans. View</RECORD_DESCRIPTION>
      </RECORD>
    </qcs:QAS_HIERARCHY_RECORDS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

QAS_RELATED_RECORDS_OPER

Use this service operation to return a list of related records for all the fields in the requested record.

Request Message: QAS_RELATED_RECORDS_REQ_MSG

Element Name	Description
RECORD_NAME	Complete record name. Required.
Required element	

Example Request :

This request will return a list of all the related records for *QE_EMPLOYEE*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_RELATED_RECORDS_REQ_MSG.VE
RSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_RELATED_RECORDS_REQ_MSG>
      <RECORD_NAME>QE_EMPLOYEE</RECORD_NAME>
    </qas:QAS_RELATED_RECORDS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_RELATED_RECORDS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
FIELD_NAME	Field for which a related record exists.
RELATED_RECORD_NAME	Record name.
RELATED_RECORD_DESCRIPTION	Record description.
INFORMATION_MESSAGE	Returns information about the request. For example if the criterion in the request is incorrect or if no records meet the criteria, a message indicating the error is returned.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_RELATED_RECORDS_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterpri
se/Tools/schemas/QAS_RELATED_RECORDS_RESP_MSG.VERSION_1">
      <FIELD>
        <FIELD_NAME>QE_JOBCODE</FIELD_NAME>
        <RELATED_RECORD_NAME>QE_JOBCODE_TBL</RELATED_RECORD_NAME>
        <RELATED_RECORD_DESCRIPTION>JobCode Table</RELATED_RECORD_DESCRIPTION>
      </FIELD>
      <FIELD>
        <FIELD_NAME>DEPTID</FIELD_NAME>
        <RELATED_RECORD_NAME>QE_DEPT_TBL</RELATED_RECORD_NAME>
        <RELATED_RECORD_DESCRIPTION>QE Data Department Table</RELATED_RECORD_DE
SCRIPTION>
      </FIELD>
    </qcs:QAS_RELATED_RECORDS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Adding Fields to Query Content

This section describes the service operations that are available for adding fields to a query. These include:

- QAS_FIELDS_OPER
- QAS_FIELDS_PROPS_OPER

QAS_FIELDS_OPER

Use this service operation to return the fields from all records that the user can access.

Request Message: QAS_FIELDS_REQ_MSG

<i>Element Name</i>	<i>Description</i>
SEARCH_STRING Required element	Search string used for specifying the field name or the first few characters of the field name. If a string is not entered, fields from all records that are the user can access will be returned.

Example Request:

This request will return all the fields that start with *MESSAGE_N*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_FIELDS_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_FIELDS_REQ_MSG>
      <SEARCH_STRING>MESSAGE_N</SEARCH_STRING>
    </qas:QAS_FIELDS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_FIELDS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
FIELD_NAME	Field name.
INFO_MESSAGE	Returns information about the request. For example if the criterion in the request is incorrect or if no fields meet the criteria, a message indicating the error is returned.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_FIELDS_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_FIELDS_RESP_MSG.VERSION_1">
      <FIELD_NAME>MESSAGE_NBR</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR01</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR02</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR03</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR04</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR05</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR06</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR07</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR08</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR09</FIELD_NAME>
      <FIELD_NAME>MESSAGE_NBR10</FIELD_NAME>
    </qcs:QAS_FIELDS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

QAS_FIELD_PROPS_OPER

Use this service operation to return the properties of a given field.

Request Message: QAS_FIELD_PROPS_REQ_MSG

<i>Element Name</i>	<i>Description</i>
FIELD_NAME Required element	Complete field name. Required.

Example Request:

This request message will return the details for field *MESSAGE_NBR*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_FIELD_PROPS_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_FIELD_PROPS_REQ_MSG>
      <FIELD_NAME>MESSAGE_NBR</FIELD_NAME>
    </qas:QAS_FIELD_PROPS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_FIELD_PROPS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
FIELD_NAME	Field name.
FIELD_DESCRIPTION	Field description.
FIELD_TYPE	Field type. Valid values are: <ul style="list-style-type: none"> • Date • Time • Datetime • Character • Long Character • Number • Signed Number • Image • Image Reference • File

Element Name	Description
FIELD_FORMAT	Field format. Valid values are: <ul style="list-style-type: none"> • None • Name • Phone • Zip Code • Social Security Number • Upper • Mixed Case • Century • Numbers Only • Social Insurance Number • International Phone Number • International Postal Code • Seconds • Microseconds • Century/Seconds • Century/Microseconds
FIELD_LENGTH	Field length.
FIELD_DECIMALS	Number of decimal positions.
FIELD_SHORT_NAME	Field's short name.
FIELD_LONG_NAME	Field's long name.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_FIELD_PROPS_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_FIELD_PROPS_RESP_MSG.VERSION_1">
      <PROPERTY>
        <FIELD_NAME>MESSAGE_NBR</FIELD_NAME>
        <FIELD_DESCRIPTION>Message Number</FIELD_DESCRIPTION>
        <FIELD_TYPE>Number</FIELD_TYPE>
        <FIELD_FORMAT>None</FIELD_FORMAT>
        <FIELD_LENGTH>5</FIELD_LENGTH>
      </PROPERTY>
    </qcs:QAS_FIELD_PROPS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        <FIELD_DECIMALS>0</FIELD_DECIMALS>
        <FIELD_SHORT_NAME>Msg</FIELD_SHORT_NAME>
        <FIELD_LONG_NAME>Message Number</FIELD_LONG_NAME>
    </PROPERTY>
  </qcs:QAS_FIELD_PROPS_RESP_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Defining Selection Criteria

This section describes the service operations that are available for working with trees in the selection criteria. These include:

- QAS_TREES_OPER
- QAS_TREE_DETAILS_OPER

QAS_TREES_OPER

This service operation returns the list of PeopleSoft trees that are accessible to the user.

Request Message: QAS_TREES_REQ_MSG

<i>Element Name</i>	<i>Description</i>
SEARCH_STRING	Complete tree name or the first few letters of the name.
Required element	If no value is entered, a list of all accessible trees will be returned.

Example Request:

This request returns all trees that start with *QE_J*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_TREES_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_TREES_REQ_MSG>
      <SEARCH_STRING>QE_J</SEARCH_STRING>
    </qas:QAS_TREES_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_TREES_RESP_MSG

<i>Element Name</i>	<i>Description</i>
TREE_NAME	Tree name.
TREE_DESCRIPTION	Tree description.
TREE_SET_ID	Set ID.

Element Name	Description
TREE_SET_CONTROL_VALUE	Set control value.
TREE_EFFECTIVE_DATE	Effective date.
INFO_MESSAGE	Returns information about the request. For example if the criterion in the request is incorrect or if no trees meet the criteria, a message indicating the error is returned.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_TREES_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterprise/Tools/s
chemas/QAS_TREES_RESP_MSG.VERSION_1">
      <TREE>
        <TREE_NAME>QE_JOBCODES</TREE_NAME>
        <TREE_DESCRIPTION>JobCodes</TREE_DESCRIPTION>
        <TREE_SET_ID/>
        <TREE_SET_CONTROL_VALUE/>
        <TREE_EFFECTIVE_DATE>1999-01-01</TREE_EFFECTIVE_DATE>
      </TREE>
    </qcs:QAS_TREES_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

QAS_TREE_DETAILS_OPER

This service operation returns the tree details for a specific tree.

Request Message: QAS_TREE_DETAILS_REQ_MSG

Element Name	Description
TREE_NAME Required element	Complete tree name. Required.
TREE_SET_ID Required element	SetID. Required.
TREE_SET_CONTROL Required element	Set control value. Required.
TREE_EFFECTIVE_DATE Required element	Effective date in YYYY-MM-DD format. Required.

Element Name	Description
INFO_MESSAGE	Returns information about the request. For example if the criterion in the request is incorrect or if no trees meet the criteria, a message indicating the error is returned.

Example Request:

This is an example of a request to retrieve tree details for the tree *QE_JOBCODES*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_TREE_DETAILS_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_TREE_DETAILS_REQ_MSG>
      <TREE_NAME>QE_JOBCODES</TREE_NAME>
      <TREE_SET_ID/>
      <TREE_SET_CONTROL/>
      <TREE_EFFECTIVE_DATE>1999-01-01</TREE_EFFECTIVE_DATE>
    </qas:QAS_TREE_DETAILS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_TREE_DETAILS_RESP_MSG

Element Name	Description
TREE_NAME	Tree name.
NODE_NAME	Tree node name.
NODE_DESCRIPTION	Tree node description.
NODE_LEVEL	Tree node level.
PARENT_NODE	Parent node.

Example response:

This response is very long, so only a portion is shown.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_TREE_DETAILS_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_TREE_DETAILS_RESP_MSG.VERSION_1">
      <TREE_NAME>QE_JOBCODES</TREE_NAME>
      <NODE>
        <NODE_LEVEL>1</NODE_LEVEL>
        <NODE_NAME>ALL JOBS</NODE_NAME>
        <NODE_DESCRIPTION>All Job Codes</NODE_DESCRIPTION>
      </NODE>
      <NODE>
        <NODE_LEVEL>2</NODE_LEVEL>
        <NODE_NAME>EXECUTIVE</NODE_NAME>
        <NODE_DESCRIPTION>Executive</NODE_DESCRIPTION>
        <PARENT_NODE>ALL JOBS</PARENT_NODE>
      </NODE>
    </qcs:QAS_TREE_DETAILS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    </NODE>
    <NODE>
      <NODE_LEVEL>3</NODE_LEVEL>
      <NODE_NAME>VP</NODE_NAME>
      <NODE_DESCRIPTION>Vice President</NODE_DESCRIPTION>
      <PARENT_NODE>EXECUTIVE</PARENT_NODE>
    </NODE>
    ...
  </qcs:QAS_TREE_DETAILS_RESP_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Deleting a Query

QAS provides a service operation to delete a query in the PeopleSoft database. Query security will allow only authorized users to delete a query. Only the query owner can delete a private query.

Note: Connected queries cannot be deleted using QAS.

QAS_QUERY_DELETE_OPER

This service operation is used to delete an existing query.

Request Message: QAS_QUERY_DELETE_REQ_MSG

<i>Element Name</i>	<i>Description</i>
QUERY_NAME	Complete query name. Required.
Required element	

Example Request:

This is an example of a request to delete the query *MGR_TEST*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_DELETE_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERY_DELETE_REQ_MSG>
      <QUERY_NAME>MGR_TEST</QUERY_NAME>
    </qas:QAS_QUERY_DELETE_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_QUERY_DELETE_RESP_MSG

<i>Element Name</i>	<i>Description</i>
STATUS	Status of the delete operation, either <i>Success</i> or <i>Failure</i> .
INFO_MESSAGE	If the status is <i>Failure</i> , the reason for the failure is indicated.

Saving a Query

QAS provides a service operation to save the query in the PeopleSoft application database. The save query request must contain a name and at least one select, one record, and one field.

QAS_SAVE_QUERY_OPER

This service operation validates and verifies XML-format query and saves the query in the PeopleSoft database.

Request Message: QAS_QUERY_SAVE_REQ_MSG

This message has several groupings, and depending on the specific query, some groupings may appear multiple times, while other groupings do not apply. The elements are listed here in groups.

<Records> Every request must contain at least one record. For each record, indicate the elements.

Records Elements	Description
<i>All element tags are required if, unless indicated otherwise</i>	<i>All element values are required except where noted</i>
RCDNUM	Record number.
RCDNAME	Record name.
CORRNAME	Alias name such as <i>A,B,C</i> , and so on.
JOINTYPE	Type of join. Valid values are: <ul style="list-style-type: none"> • None • Hierarchy • Related • Tree • LeftOuter • RelatedLeftOuter Value is required only when the corresponding record is part of a JOIN in the query
JOINRCDALIAS	Record with which join is done. Value is required only when the corresponding record is part of a JOIN in the query.

Records Elements	Description
All element tags are required if, unless indicated otherwise	All element values are required except where noted
JOINFIELD	Field with which join is done. Value is required only when the corresponding record is part of a JOIN in the query.
RCDSELECTNUM	Select number.

<Fields> The request must contain at least one field. For each field, indicate the elements:

Fields Elements	Description
All element tags are required if, unless indicated otherwise	All element values are required except where noted
FIELDNUM	Field number.
FIELDNAME	Field name. When an expression is used as a field, the expression name is used for the field name and the value entered for this element is ignored.
FIELDRCDDNUM	Record number. This element is ignored when an expression is used as a field.
DESCR	Description.
Not Required.	Value is not required.
AGGREGATE_TYPE	Aggregation type. Valid values are: <ul style="list-style-type: none"> • Sum • Count • Min • Max • Average • None

Fields Elements <i>All element tags are required if, unless indicated otherwise</i>	Description <i>All element values are required except where noted</i>
HEADING_TYPE	<p>Heading type. Valid values are:</p> <ul style="list-style-type: none"> • None • Text • RFT Short • RFT Long <hr/> <p>Note: If an expression is used as a field, the only valid value is <i>Heading</i>.</p>
HEADING	<p>If the HEADING_TYPE is <i>Text</i>, this element is used to enter the heading text.</p> <hr/> <p>Note: Value is required when the HEADING_TYPE value is <i>Text</i>. The default heading for an expression used as a field is the expression text.</p>
COLUMNNUM	Column number.
ORDERBYNUM	Order by number.
ORDERBYDIR	<p>Direction of field ordering. Valid values are:</p> <ul style="list-style-type: none"> • Ascending • Descending <hr/> <p>Note: Value is not required, the default is Ascending.</p>
FIELDSELECTNUM	Identifier of SELECT in which this field is included.
EXPRESSION_AS_FIELD	Specify a valid expression number in order to use an expression as a field. For regular fields, leave this element blank.

<Criteria> If the query contains criteria, all criteria elements are required. For each criteria, indicate the elements.

Criteria Elements <i>All element tags are required to define criteria</i>	Description <i>All element values are required except where noted</i>
CRTNUM	Criterion number.

Criteria Elements	Description
All element tags are required to define criteria	All element values are required except where noted
CRTNAME	Criterion name.
CRTHAVINGFLAG	<p>True if specifying a HAVING clause. Valid values are:</p> <ul style="list-style-type: none"> • True • False <hr/> <p>Note: Value is not required; the default is <i>False</i>.</p>
CRTSELECTNUM	Identifier for the SELECT for this criterion.
CRTNEGATION	<p>Negation in criterion. Valid values are:</p> <ul style="list-style-type: none"> • True • False <hr/> <p>Note: This is the equivalent of NOT in a SQL statement.</p>

<p>Criteria Elements</p> <p><i>All element tags are required to define criteria</i></p>	<p>Description</p> <p><i>All element values are required except where noted</i></p>
<p>CONDITION_TYPE</p>	<p>Condition type in criterion. Valid values are:</p> <ul style="list-style-type: none"> • none • between • not between • equal to • not equal to • exists • not exists • greater than • not greater than • in list • not in list • in tree • not in tree • is null • not is null • less than • not less than • like • not like • EffDate less than or equal to • EffDate greater than or equal to • EffDate less than • EffDate greater than • First EffDate • Last EffDate • in tree join

Criteria Elements	Description
All element tags are required to define criteria	All element values are required except where noted
LEFT_PARENTHESIS_LEVEL	Left parenthesis level specified from 0 onwards. Used for GROUP BY.
RIGHT_PARENTHESIS_LEVEL	Right parenthesis level specified from 0 onwards. Used for GROUP BY.
CRTEXP1TYPE	Expression 1 type in criterion. Valid values are: <ul style="list-style-type: none"> • Field • Expression
CRTEXP1TEXT	Expression text for criterion 1.
CRTEXP1NUM	Expression number for the criteria.
CRTEXP1RCDALIAS	Alias for record used in expression 1 criterion. Used when CRTEXP1TYPE is <i>Field</i> . Alias name, such as <i>A,B,C</i> , and so on.
CRTEXP1FIELD	Field used in expression 1 criterion. Used when CRTEXP1TYPE is <i>Field</i> .
CRTEXP2RCDALIAS	Alias for record used in expression 2 criterion. Used when CRTEXP2TYPE is <i>Field</i> .
CRTEXP2FIELD	Field used in expression 2.

<p>Criteria Elements</p> <p><i>All element tags are required to define criteria</i></p>	<p>Description</p> <p><i>All element values are required except where noted</i></p>
<p>CRTEXP2TYPE</p>	<p>Expression 2 type in criterion. Valid values are:</p> <ul style="list-style-type: none"> • Field • Expression • Constant • Prompt • Subquery • List • Const-Const • Const-Field • Const-Expr • Field-Const • Field-Field • Field-Expr • Expr-Const • Expr-Field • Expr-Expr
<p>CRTEXP2TEXT</p>	<p>Expression 2 text is used to specify lists in A,B,C,D format, trees in string format, and prompt numbers.</p>
<p>CRTLOGICALOPER</p>	<p>Logical operator that links the criteria. Valid values are:</p> <ul style="list-style-type: none"> • AND • OR • not used
<p>CRT_BELONGSTO</p>	<p>If the criteria is for an outer join, you must indicate the correlation name (alias) that this criteria belongs to.</p> <hr/> <p>Note: This element is used for a query that contains an Outer Join. For others, value for this element is null.</p> <hr/>

<Expressions> If a query contains expressions, all expression elements are required. For each expression, indicate the elements.

Expressions Elements	Description
All element tags are required to define expression	All element values are required except where noted
EXPNUM	Expression number.
EXPSELECTNUM	Identifier of SELECT number for this expression.
EXPNAME	Expression name.
EXPTYPE	Expression type. Valid values are: <ul style="list-style-type: none"> • Character • Date • Datetime • Drilling URL • Long Character • Number • Signed Number • Time • Image
EXPLENGTH	Expression length.
EXPDECIMALPOS	Number of decimal places in the expression.
EXPTEXT	Expression text.
EXPAGGREGATE	Specifies whether the expression is an aggregate function.

<Prompts> If the query contains prompts, all prompt elements are required.

Prompts Elements	Description
All element tags are required to define prompt	All element values are required except where noted
PROMPT_NUM	Prompt number.
PROMPT_NAME	Prompt name.

Prompts Elements	Description
All element tags are required to define prompt	All element values are required except where noted
PROMPT_UNIQUE_NAME	Unique prompt name. <hr/> Note: This value is not required; however, the client application is responsible for ensuring that all default prompt names are unique. <hr/>
PROMPT_FLDNAME	Name of the field used as prompt.
PROMPT_TABLE	Name of the prompt table.
PROMPT_EDITTYPE	Edit type. Valid values are: <ul style="list-style-type: none"> • No table edit • Prompt table • Translate table • Yes/No table
PROMPT_HEADING	Heading used for the prompt.
PROMPT_HEADINGTYPE	Type of prompt heading. Valid values are: <ul style="list-style-type: none"> • RFT Long • RFT Short • Text
PROMPT_FLDLENGTH	Field length for prompt.
PROMPT_FLDDECIMALPOS	Number of decimal positions in prompt.

<Select> The request must contain at least one select. For each select, indicate the elements.

Select Elements	Description
All elements are required.	All element values are required except where noted
SELECTNUM	SELECT number.
PARENTSELECTNUM	Number of parent SELECT. For main SELECT, this must be set to 0.

Select Elements	Description
All elements are required.	All element values are required except where noted
SELECTTYPE	Type of selection. Valid values are: <ul style="list-style-type: none"> • Main • Union • Subquery
QRYDISTINCT	Indicates whether this query is distinct. Valid values are: <ul style="list-style-type: none"> • True • False <hr/> Note: Value is not required; the default is <i>False</i> . <hr/>

<Properties> The request must contain the properties that define the query. Indicate the property elements for the query.

Properties Elements	Description
All element tags are required	All element values are required except where noted
QUERY_NAME	Query name.
DESCRIPTION	Short description. Optional.
DESCRLONG	Long description. Optional.
QUERY_OWNER	Indicate the query owner. Valid values are: <ul style="list-style-type: none"> • Public • Private <hr/> Note: If the query owner is not specified, the save will fail. <hr/>

Depending on the complexity of the query, many different variations are available. Refer to the following sample queries.

Response Message: QAS_QUERY_SAVE_RESP_MSG

Element Name	Description
STATUS	Status of the save operation, either <i>Success</i> or <i>Failure</i> .
INFO_MESSAGE	Returns information about the save operation.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_QUERY_SAVE_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterprise/To
ols/schemas/QAS_QUERY_SAVE_RESP_MSG.VERSION_1">
      <INFO_MESSAGE>Query TEST1 saved successfully.</INFO_MESSAGE>
    </qcs:QAS_QUERY_SAVE_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Simple Query Example

This is an example of a simple query TEST1 that has two columns. In this example, no criteria, expressions, or prompts exist. The SQL for this query is:

```
SELECT A.MESSAGE SET_NBR, A.DESCR
FROM PMSGSETDEFN A
```

SOAP document:

```
soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qa=
s="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_SAVE_REQ_MSG.VERSION_
1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERY_SAVE_REQ_MSG>
      <!--1 or more repetitions:-->
      <RECORD>
        <RCDNUM>1</RCDNUM>
        <RCDNAME>PMSGSETDEFN</RCDNAME>
        <CORRNAME>A</CORRNAME>
        <JOINTYPE></JOINTYPE>
        <JOINRCDALIAS></JOINRCDALIAS>
        <JOINFIELD></JOINFIELD>
        <RCDSELECTNUM>1</RCDSELECTNUM>
      </RECORD>
      <!--1 or more repetitions:-->
      <FIELD>
        <FIELDNUM>1</FIELDNUM>
        <FIELDNAME>MESSAGE_set_nbr</FIELDNAME>
        <FIELDRCDDNUM>1</FIELDRCDDNUM>
        <DESCR></DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>RFT Short</HEADING_TYPE>
        <HEADING></HEADING>
        <COLUMNNUM>1</COLUMNNUM>
        <ORDERBYNUM>1</ORDERBYNUM>
        <ORDERBYDIR></ORDERBYDIR>
        <FIELDSELECTNUM>1</FIELDSELECTNUM>
        <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
      </FIELD>
      <FIELD>
        <FIELDNUM>2</FIELDNUM>
        <FIELDNAME>descr</FIELDNAME>
        <FIELDRCDDNUM>1</FIELDRCDDNUM>
        <DESCR></DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>RFT Short</HEADING_TYPE>
        <HEADING></HEADING>
        <COLUMNNUM>2</COLUMNNUM>
        <ORDERBYNUM>0</ORDERBYNUM>
        <ORDERBYDIR></ORDERBYDIR>
        <FIELDSELECTNUM>1</FIELDSELECTNUM>
        <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
```

```

    </FIELD>
  <SELECT>
    <SELECTNUM>1</SELECTNUM>
    <PARENTSELECTNUM>0</PARENTSELECTNUM>
    <SELECTTYPE>Main</SELECTTYPE>
    <QRYDISTINCT></QRYDISTINCT>
  </SELECT>
  <QUERY_NAME>SimpleQuery</QUERY_NAME>
  <DESCRIPTION>test a simple query</DESCRIPTION>
  <DESCRLONG>Simple Query with one table and 2 fields</DESCRLONG>
  <QUERY_OWNER>Public</QUERY_OWNER>
</qas:QAS_QUERY_SAVE_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Query Using Prompts Example

This is an example of a query using two prompts, `NODE_TYPE` uses the translate table and `PORTAL_NAME` uses a prompt table. The SQL for this query is:

```

SELECT A.MSGNODENAME, A.VERSION, A.NODE_TYPE, A.PORTAL_NAME
FROM PMSGNODEDEFN A
WHERE A.NODE_TYPE = :1
      AND A.PORTAL_NAME = :2

```

SOAP document:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_SAVE_REQ_MSG.VERSION=>
_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERY_SAVE_REQ_MSG>
      <!--1 or more repetitions:-->
      <RECORD>
        <RCDNUM>1</RCDNUM>
        <RCDNAME>PMSGNODEDEFN</RCDNAME>
        <CORRNAME>A</CORRNAME>
        <JOINTYPE></JOINTYPE>
        <JOINRCDALIAS></JOINRCDALIAS>
        <JOINFIELD></JOINFIELD>
        <RCDSELECTNUM>1</RCDSELECTNUM>
      </RECORD>
      <!--1 or more repetitions:-->
      <FIELD>
        <FIELDNUM>1</FIELDNUM>
        <FIELDNAME>MSGNODENAME</FIELDNAME>
        <FIELDRCDDNUM>1</FIELDRCDDNUM>
        <DESCR>Message Node Name</DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>RFT Long</HEADING_TYPE>
        <HEADING></HEADING>
        <COLUMNNUM>1</COLUMNNUM>
        <ORDERBYNUM>0</ORDERBYNUM>
        <ORDERBYDIR></ORDERBYDIR>
        <FIELDSELECTNUM>1</FIELDSELECTNUM>
        <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
      </FIELD>
      <FIELD>
        <FIELDNUM>2</FIELDNUM>
        <FIELDNAME>VERSION</FIELDNAME>
        <FIELDRCDDNUM>1</FIELDRCDDNUM>
        <DESCR>Version</DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>RFT Long</HEADING_TYPE>
        <HEADING></HEADING>
        <COLUMNNUM>2</COLUMNNUM>
        <ORDERBYNUM>0</ORDERBYNUM>

```

```

    <ORDERBYDIR></ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>
<FIELD>
  <FIELDNUM>3</FIELDNUM>
  <FIELDNAME>node_type</FIELDNAME>
  <FIELDRCDDNUM>1</FIELDRCDDNUM>
  <DESCR></DESCR>
  <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
  <HEADING_TYPE>RFT Long</HEADING_TYPE>
  <HEADING></HEADING>
  <COLUMNNUM>3</COLUMNNUM>
  <ORDERBYNUM>0</ORDERBYNUM>
  <ORDERBYDIR></ORDERBYDIR>
  <FIELDSELECTNUM>1</FIELDSELECTNUM>
  <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>
<FIELD>
  <FIELDNUM>4</FIELDNUM>
  <FIELDNAME>PORTAL_NAME</FIELDNAME>
  <FIELDRCDDNUM>1</FIELDRCDDNUM>
  <DESCR>Portal</DESCR>
  <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
  <HEADING_TYPE>RFT Long</HEADING_TYPE>
  <HEADING></HEADING>
  <COLUMNNUM>4</COLUMNNUM>
  <ORDERBYNUM>0</ORDERBYNUM>
  <ORDERBYDIR></ORDERBYDIR>
  <FIELDSELECTNUM>1</FIELDSELECTNUM>
  <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>
<!--Zero or more repetitions:-->
<CRITERION>
  <CRTNUM>1</CRTNUM>
  <CRTNAME>1</CRTNAME>
  <CRTHAVINGFLAG>False</CRTHAVINGFLAG>
  <CRTSELECTNUM>1</CRTSELECTNUM>
  <CRTNEGATION>False</CRTNEGATION>
  <CONDITION_TYPE>equal to</CONDITION_TYPE>
  <LEFT_PARENTHESIS_LEVEL>0</LEFT_PARENTHESIS_LEVEL>
  <RIGHT_PARENTHESIS_LEVEL>0</RIGHT_PARENTHESIS_LEVEL>
  <CRTEXP1TYPE>Field</CRTEXP1TYPE>
  <CRTEXP1TEXT></CRTEXP1TEXT>
  <CRTEXP1NUM></CRTEXP1NUM>
  <CRTEXP1RCDALIAS>A</CRTEXP1RCDALIAS>
  <CRTEXP1FIELD>NODE_TYPE</CRTEXP1FIELD>
  <CRTEXP2RCDALIAS></CRTEXP2RCDALIAS>
  <CRTEXP2FIELD></CRTEXP2FIELD>
  <CRTEXP2TYPE>prompt</CRTEXP2TYPE>
  <CRTEXP2TEXT>:1</CRTEXP2TEXT>
  <CRTLOGICALOPER>not used</CRTLOGICALOPER>
</CRITERION>
<CRITERION>
  <CRTNUM>2</CRTNUM>
  <CRTNAME>2</CRTNAME>
  <CRTHAVINGFLAG>False</CRTHAVINGFLAG>
  <CRTSELECTNUM>1</CRTSELECTNUM>
  <CRTNEGATION>False</CRTNEGATION>
  <CONDITION_TYPE>equal to</CONDITION_TYPE>
  <LEFT_PARENTHESIS_LEVEL>0</LEFT_PARENTHESIS_LEVEL>
  <RIGHT_PARENTHESIS_LEVEL>0</RIGHT_PARENTHESIS_LEVEL>
  <CRTEXP1TYPE>Field</CRTEXP1TYPE>
  <CRTEXP1TEXT></CRTEXP1TEXT>
  <CRTEXP1NUM></CRTEXP1NUM>
  <CRTEXP1RCDALIAS>A</CRTEXP1RCDALIAS>
  <CRTEXP1FIELD>portal_name</CRTEXP1FIELD>
  <CRTEXP2RCDALIAS></CRTEXP2RCDALIAS>
  <CRTEXP2FIELD></CRTEXP2FIELD>
  <CRTEXP2TYPE>prompt</CRTEXP2TYPE>
  <CRTEXP2TEXT>:2</CRTEXP2TEXT>

```

```

        <CRTLOGICALOPER>not used</CRTLOGICALOPER>
    </CRITERION>
    <!--Zero or more repetitions:-->
    <PROMPT>
        <PROMPT_NUM>1</PROMPT_NUM>
        <PROMPT_NAME>NodeType</PROMPT_NAME>
        <PROMPT_UNIQUE_NAME>BIND1</PROMPT_UNIQUE_NAME>
        <PROMPT_FLDNAME>NODE_TYPE</PROMPT_FLDNAME>
        <PROMPT_TABLE></PROMPT_TABLE>
        <PROMPT_EDITTYPE>Translate table</PROMPT_EDITTYPE>
        <PROMPT_HEADING>Node Type</PROMPT_HEADING>
        <PROMPT_HEADINGTYPE>Text</PROMPT_HEADINGTYPE>
        <PROMPT_FLDLENGTH>30</PROMPT_FLDLENGTH>
        <PROMPT_FLDDECIMALPOS>0</PROMPT_FLDDECIMALPOS>
    </PROMPT>
    <PROMPT>
        <PROMPT_NUM>2</PROMPT_NUM>
        <PROMPT_NAME>PortalName</PROMPT_NAME>
        <PROMPT_UNIQUE_NAME>BIND2</PROMPT_UNIQUE_NAME>
        <PROMPT_FLDNAME>PORTAL_TYPE</PROMPT_FLDNAME>
        <PROMPT_TABLE>PSPRDMDEFN</PROMPT_TABLE>
        <PROMPT_EDITTYPE>Prompt table</PROMPT_EDITTYPE>
        <PROMPT_HEADING>Portal</PROMPT_HEADING>
        <PROMPT_HEADINGTYPE>Text</PROMPT_HEADINGTYPE>
        <PROMPT_FLDLENGTH>30</PROMPT_FLDLENGTH>
        <PROMPT_FLDDECIMALPOS>0</PROMPT_FLDDECIMALPOS>
    </PROMPT>
    <!--1 or more repetitions:-->
    <SELECT>
        <SELECTNUM>1</SELECTNUM>
        <PARENTSELECTNUM>0</PARENTSELECTNUM>
        <SELECTTYPE>Main</SELECTTYPE>
        <QRYDISTINCT>False</QRYDISTINCT>
    </SELECT>
    <QUERY_NAME>Query_with_prompt</QUERY_NAME>
    <DESCRIPTION>Prompt example</DESCRIPTION>
    <DESCRLONG>Query prompts for node type and portal name</DESCRLONG>
    <QUERY_OWNER>public</QUERY_OWNER>
    </qas:QAS_QUERY_SAVE_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Related Join Query Example

In a related record join, you can automatically join two records based on a relationship that has been predefined in the record designer. For example, if a field has a prompt table defined for it, PeopleSoft Query displays a join link to the right of the shared field.

This SQL creates a query with a related join:

```

SELECT A.PRCJOBNAME, A.PRCSTYPE, B.SERVERNAME
FROM PS_PRCJOBDEFN A, PS_SERVERDEFN B
WHERE B.SERVERNAME = A.SERVERNAME

```

SOAP document:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_SAVE_REQ_MSG.VERSION_1">
    <soapenv:Header/>
    <soapenv:Body>
        <qas:QAS_QUERY_SAVE_REQ_MSG>
            <!--1 or more repetitions:-->
            <RECORD>
                <RCDNUM>1</RCDNUM>
                <RCDNAME>PRCSJOBDEFN</RCDNAME>
                <CORRNAME>A</CORRNAME>
            </RECORD>
        </qas:QAS_QUERY_SAVE_REQ_MSG>
    </soapenv:Body>
</soapenv:Envelope>

```

```

    <JOINTYPE></JOINTYPE>
    <JOINRCDALIAS></JOINRCDALIAS>
    <JOINFIELD></JOINFIELD>
    <RCDSELECTNUM>1</RCDSELECTNUM>
</RECORD>
<RECORD>
    <RCDNUM>2</RCDNUM>
    <RCDNAME>SERVERDEFN</RCDNAME>
    <CORRNAME>B</CORRNAME>
    <JOINTYPE>related</JOINTYPE>
    <JOINRCDALIAS>A</JOINRCDALIAS>
    <JOINFIELD>SERVERNAME</JOINFIELD>
    <RCDSELECTNUM>1</RCDSELECTNUM>
</RECORD>
<!--1 or more repetitions:-->
<FIELD>
    <FIELDNUM>1</FIELDNUM>
    <FIELDNAME>PRCSJOBNAME</FIELDNAME>
    <FIELDRCDDNUM>1</FIELDRCDDNUM>
    <DESCR>Process Job Name</DESCR>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Long</HEADING_TYPE>
    <HEADING></HEADING>
    <COLUMNNUM>1</COLUMNNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>
<FIELD>
    <FIELDNUM>2</FIELDNUM>
    <FIELDNAME>PRCSTYPE</FIELDNAME>
    <FIELDRCDDNUM>1</FIELDRCDDNUM>
    <DESCR>Process Type</DESCR>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Long</HEADING_TYPE>
    <HEADING></HEADING>
    <COLUMNNUM>2</COLUMNNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>
<FIELD>
    <FIELDNUM>3</FIELDNUM>
    <FIELDNAME>SERVERNAME</FIELDNAME>
    <FIELDRCDDNUM>2</FIELDRCDDNUM>
    <DESCR>Server Name</DESCR>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Long</HEADING_TYPE>
    <HEADING></HEADING>
    <COLUMNNUM>3</COLUMNNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>
<!--1 or more repetitions:-->
<SELECT>
    <SELECTNUM>1</SELECTNUM>
    <PARENTSELECTNUM>0</PARENTSELECTNUM>
    <SELECTTYPE>Main</SELECTTYPE>
    <QRYDISTINCT>False</QRYDISTINCT>
</SELECT>
<QUERY_NAME>RelatedJoin</QUERY_NAME>
<DESCRIPTION>Related Join example</DESCRIPTION>
<DESCRLONG>Query with related join</DESCRLONG>
<QUERY_OWNER>Public</QUERY_OWNER>
</gas:QAS_QUERY_SAVE_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Related Left Outer Join Query Example

This SQL creates a query with a related left outer join:

```
SELECT A.PORTAL_NAME
FROM (PSMSGNODEDEFN A LEFT OUTER JOIN PSPRDMDEFN B ON B.PORTAL_NAME = A.PORTAL_
NAME )
```

SOAP document:

```
<QAS_QUERY_SAVE_REQ_MSG>
  <RECORD>
    <RCDNUM>1</RCDNUM>
    <RCDNAME>PSMSGNODEDEFN</RCDNAME>
    <CORRNAME>A</CORRNAME>
    <JOINTYPE/>
    <JOINRCDALIAS/>
    <JOINFIELD/>
    <RCDSELECTNUM>1</RCDSELECTNUM>
  </RECORD>
  <RECORD>
    <RCDNUM>2</RCDNUM>
    <RCDNAME>PSPRDMDEFN</RCDNAME>
    <CORRNAME>B</CORRNAME>
    <JOINRCDALIAS>A</JOINRCDALIAS>
    <JOINFIELD>PORTAL_NAME</JOINFIELD>
    <JOINTYPE>RelatedLeftOuter</JOINTYPE>
    <RCDSELECTNUM>1</RCDSELECTNUM>
  </RECORD>
  <FIELD>
    <FIELDNUM>1</FIELDNUM>
    <FIELDNAME>PORTAL_NAME</FIELDNAME>
    <FIELDRCDNUM>1</FIELDRCDNUM>
    <DESCR>Portal Name</DESCR>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Short</HEADING_TYPE>
    <HEADING>Portal Name</HEADING>
    <COLUMNNUM>1</COLUMNNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD>0</EXPRESSION_AS_FIELD>
  </FIELD>
  <CRITERION>
    <CRTNUM>1</CRTNUM>
    <CRTNAME>1</CRTNAME>
    <CRTHAVINGFLAG>False</CRTHAVINGFLAG>
    <CRTSELECTNUM>1</CRTSELECTNUM>
    <CRTNEGATION>False</CRTNEGATION>
    <CONDITION_TYPE>equal to</CONDITION_TYPE>
    <LEFT_PARENTHESIS_LEVEL>0</LEFT_PARENTHESIS_LEVEL>
    <RIGHT_PARENTHESIS_LEVEL>0</RIGHT_PARENTHESIS_LEVEL>
    <CRTEXP1TYPE>Field</CRTEXP1TYPE>
    <CRTEXP1TEXT>B.PORTAL_NAME</CRTEXP1TEXT>
    <CRTEXP1NUM>0</CRTEXP1NUM>
    <CRTEXP1RCDALIAS>B</CRTEXP1RCDALIAS>
    <CRTEXP1FIELD>PORTAL_NAME</CRTEXP1FIELD>
    <CRTEXP2RCDALIAS>A</CRTEXP2RCDALIAS>
    <CRTEXP2FIELD>PORTAL_NAME</CRTEXP2FIELD>
    <CRTEXP2TYPE>Field</CRTEXP2TYPE>
    <CRTEXP2TEXT>A.PORTAL_NAME</CRTEXP2TEXT>
    <CRTLOGICALOPER>not used</CRTLOGICALOPER>
    <CRT_BELONGSTO>B</CRT_BELONGSTO>
  </CRITERION>
  <SELECT>
    <SELECTNUM>1</SELECTNUM>
    <PARENTSELECTNUM>0</PARENTSELECTNUM>
    <SELECTTYPE>Main</SELECTTYPE>
    <QRYDISTINCT>False</QRYDISTINCT>
  </SELECT>
```

```

<QUERY_NAME>OJ_TRIAL</QUERY_NAME>
<DESCRIPTION>This is a query with a left outer join</DESCRIPTION>
<DESCRLONG>outer join example used for testing with QAS</DESCRLONG>
<QUERY_OWNER>Public</QUERY_OWNER>
</QAS_QUERY_SAVE_REQ_MSG>

```

Hierarchy Join Query Example

A record hierarchy join joins a parent table to a child table. A child table is a table that uses all the same key fields as its parent, plus one or more additional keys. The SQL for this query is:

```

SELECT A.PRCNAME, A.PRCSTYPE, B.PRCSEGRP
FROM PS_PRCSEDEFN A, PS_PRCSEDEFNGRP B
WHERE B.PRCSTYPE = A.PRCSTYPE AND B.PRCNAME = A.PRCNAME

```

Soap document:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q=
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_SAVE_REQ_MSG.VERSION=
_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERY_SAVE_REQ_MSG>
      <!--1 or more repetitions:-->
      <RECORD>
        <RCDNUM>1</RCDNUM>
        <RCDNAME>PRCSEDEFN</RCDNAME>
        <CORRNAME>A</CORRNAME>
        <JOINTYPE></JOINTYPE>
        <JOINRCDALIAS></JOINRCDALIAS>
        <JOINFIELD></JOINFIELD>
        <RCDSELECTNUM>1</RCDSELECTNUM>
      </RECORD>
      <RECORD>
        <RCDNUM>2</RCDNUM>
        <RCDNAME>PRCSEDEFNGRP</RCDNAME>
        <CORRNAME>B</CORRNAME>
        <JOINTYPE>hierarchy</JOINTYPE>
        <JOINRCDALIAS>A</JOINRCDALIAS>
        <JOINFIELD></JOINFIELD>
        <RCDSELECTNUM>1</RCDSELECTNUM>
      </RECORD>
      <!--1 or more repetitions:-->
      <FIELD>
        <FIELDNUM>1</FIELDNUM>
        <FIELDNAME>PRCNAME</FIELDNAME>
        <FIELDRCDDNUM>1</FIELDRCDDNUM>
        <DESCR></DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>RFT Long</HEADING_TYPE>
        <HEADING></HEADING>
        <COLUMNNUM>1</COLUMNNUM>
        <ORDERBYNUM>0</ORDERBYNUM>
        <ORDERBYDIR></ORDERBYDIR>
        <FIELDSELECTNUM>1</FIELDSELECTNUM>
        <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
      </FIELD>
      <FIELD>
        <FIELDNUM>2</FIELDNUM>
        <FIELDNAME>PRCSTYPE</FIELDNAME>
        <FIELDRCDDNUM>1</FIELDRCDDNUM>
        <DESCR></DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>RFT Long</HEADING_TYPE>
        <HEADING></HEADING>
        <COLUMNNUM>2</COLUMNNUM>
        <ORDERBYNUM>0</ORDERBYNUM>

```



```

        <ORDERBYDIR></ORDERBYDIR>
        <FIELDSELECTNUM>1</FIELDSELECTNUM>
        <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
    </FIELD>
    <FIELD>
        <FIELDNUM>3</FIELDNUM>
        <FIELDNAME>PRCSGRP</FIELDNAME>
        <FIELDRCDDNUM>2</FIELDRCDDNUM>
        <DESCR></DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>RFT Short</HEADING_TYPE>
        <HEADING></HEADING>
        <COLUMNNUM>3</COLUMNNUM>
        <ORDERBYNUM>0</ORDERBYNUM>
        <ORDERBYDIR></ORDERBYDIR>
        <FIELDSELECTNUM>1</FIELDSELECTNUM>
        <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
    </FIELD>

    <!--1 or more repetitions:-->
    <SELECT>
        <SELECTNUM>1</SELECTNUM>
        <PARENTSELECTNUM>0</PARENTSELECTNUM>
        <SELECTTYPE>Main</SELECTTYPE>
        <QRYDISTINCT>False</QRYDISTINCT>
    </SELECT>
    <QUERY_NAME>HierarchyJoin</QUERY_NAME>
    <DESCRIPTION>Hierarchy Join example</DESCRIPTION>
    <DESCRLONG>Thios is a query with a hierarchy join</DESCRLONG>
    <QUERY_OWNER>public</QUERY_OWNER>
</qas:QAS_QUERY_SAVE_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Query with an Aggregate Value Example

This is an example of a query that contains the aggregate count for *EMPLID*. The SQL for this query is:

```

SELECT A.MESSAGE_SET_NBR, COUNT(*)
FROM PMSGCATDEFN A
GROUP BY A.MESSAGE_SET_NBR
ORDER BY 1

```

SOAP document:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_SAVE_REQ_MSG.VERSION=1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERY_SAVE_REQ_MSG>
      <!--1 or more repetitions:-->
      <RECORD>
        <RCDNUM>1</RCDNUM>
        <RCDNAME>PMSGCATDEFN</RCDNAME>
        <CORRNAME>A</CORRNAME>
        <JOINTYPE></JOINTYPE>
        <JOINRCDALIAS></JOINRCDALIAS>
        <JOINFIELD></JOINFIELD>
        <RCDSELECTNUM>1</RCDSELECTNUM>
      </RECORD>
      <!--1 or more repetitions:-->
    </FIELD>
    <FIELDNUM>1</FIELDNUM>
    <FIELDNAME>MESSAGE_SET_NBR</FIELDNAME>
    <FIELDRCDDNUM>1</FIELDRCDDNUM>
    <DESCR></DESCR>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Long</HEADING_TYPE>
  </qas:QAS_QUERY_SAVE_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

```

    <HEADING></HEADING>
    <COLUMNNUM>1</COLUMNNUM>
    <ORDERBYNUM>1</ORDERBYNUM>
    <ORDERBYDIR></ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>
<FIELD>
  <FIELDNUM>2</FIELDNUM>
  <FIELDNAME>MESSAGE_NBR</FIELDNAME>
  <FIELDRCDDNUM>1</FIELDRCDDNUM>
  <DESCR></DESCR>
  <AGGREGATE_TYPE>Count</AGGREGATE_TYPE>
  <HEADING_TYPE>Text</HEADING_TYPE>
  <HEADING>Count</HEADING>
  <COLUMNNUM>2</COLUMNNUM>
  <ORDERBYNUM>0</ORDERBYNUM>
  <ORDERBYDIR></ORDERBYDIR>
  <FIELDSELECTNUM>1</FIELDSELECTNUM>
  <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>

<!--1 or more repetitions:-->
<SELECT>
  <SELECTNUM>1</SELECTNUM>
  <PARENTSELECTNUM>0</PARENTSELECTNUM>
  <SELECTTYPE>Main</SELECTTYPE>
  <QRYDISTINCT></QRYDISTINCT>
</SELECT>
<QUERY_NAME>CountExample</QUERY_NAME>
<DESCRIPTION>Example using count</DESCRIPTION>
<DESCRLONG>This query counts number of messages in each message set<
/DESCRLONG>
  <QUERY_OWNER>public</QUERY_OWNER>
</qas:QAS_QUERY_SAVE_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Query with Expression Example

This is an example of a query that contains an expression for total price. The SQL for this query is:

```

SELECT A.QE_ORDER_NBR, A.QE_ORDER_LINE_NBR, A.QE_QTY * A.QE_PRICE
FROM PS_QEORDER_DTL A
ORDER BY 1

```

SOAP document:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_SAVE_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERY_SAVE_REQ_MSG>
      <!--1 or more repetitions:-->
      <RECORD>
        <RCDNUM>1</RCDNUM>
        <RCDNAME>qeorder_dtl</RCDNAME>
        <CORRNAME>A</CORRNAME>
        <JOINTYPE></JOINTYPE>
        <JOINRCDALIAS></JOINRCDALIAS>
        <JOINFIELD></JOINFIELD>
        <RCDSELECTNUM>1</RCDSELECTNUM>
      </RECORD>
      <!--1 or more repetitions:-->
    </FIELD>
    <FIELD>
      <FIELDNUM>1</FIELDNUM>
      <FIELDNAME>QE_ORDER_NBR</FIELDNAME>

```

```

    <FIELDRCDDNUM>1</FIELDRCDDNUM>
    <DESCR></DESCR>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Short</HEADING_TYPE>
    <HEADING></HEADING>
    <COLUMNNUM>1</COLUMNNUM>
    <ORDERBYNUM>1</ORDERBYNUM>
    <ORDERBYDIR></ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
  </FIELD>
  <FIELD>
    <FIELDNUM>2</FIELDNUM>
    <FIELDNAME>QE_ORDER_LINE_NBR</FIELDNAME>
    <FIELDRCDDNUM>1</FIELDRCDDNUM>
    <DESCR></DESCR>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Short</HEADING_TYPE>
    <HEADING></HEADING>
    <COLUMNNUM>2</COLUMNNUM>
    <ORDERBYNUM></ORDERBYNUM>
    <ORDERBYDIR></ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
  </FIELD>
  <FIELD>
    <FIELDNUM>3</FIELDNUM>
    <FIELDNAME></FIELDNAME>
    <FIELDRCDDNUM>1</FIELDRCDDNUM>
    <DESCR></DESCR>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>Text</HEADING_TYPE>
    <HEADING>Total Line Amount</HEADING>
    <COLUMNNUM>3</COLUMNNUM>
    <ORDERBYNUM>1</ORDERBYNUM>
    <ORDERBYDIR></ORDERBYDIR>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD>1</EXPRESSION_AS_FIELD>
  </FIELD>

  <!--Zero or more repetitions:-->
  <EXPRESSION>
    <EXPNUM>1</EXPNUM>
    <EXPSELECTNUM>1</EXPSELECTNUM>
    <EXPNAME>1</EXPNAME>
    <EXPTYPE>number</EXPTYPE>
    <EXPLENGTH>11</EXPLENGTH>
    <EXPDECIMALPOS>2</EXPDECIMALPOS>
    <EXPTTEXT>A.QE_QTY * A.QE_PRICE</EXPTTEXT>
    <EXPAGGREGATE>False</EXPAGGREGATE>
  </EXPRESSION>

  <!--1 or more repetitions:-->
  <SELECT>
    <SELECTNUM>1</SELECTNUM>
    <PARENTSELECTNUM>0</PARENTSELECTNUM>
    <SELECTTYPE>Main</SELECTTYPE>
    <QRYDISTINCT>False</QRYDISTINCT>
  </SELECT>
  <QUERY_NAME>test_expression</QUERY_NAME>
  <DESCRIPTION>Test using expression as field</DESCRIPTION>
  <DESCRLONG></DESCRLONG>
  <QUERY_OWNER>Public</QUERY_OWNER>
</qas:QAS_QUERY_SAVE_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Subquery Example

This is an example of a query that contains a subquery. The SQL for this query is:

```
SELECT DISTINCT A.EMPLID, A.QE_EMPLOYEE_NAME FROM PS_QE_EMPLOYEE A WHERE A.EMPLID⇒
  IN (SELECT B.QE_EMPLID FROM PS_QE_PERS_DATA B WHERE ( B.QE_HIGHLY_COMP_EMP = 'E'⇒
)) ORDER BY 2
```

SOAP document:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q⇒
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_SAVE_REQ_MSG.VERSION⇒
_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERY_SAVE_REQ_MSG>
      <!--1 or more repetitions:-->
      <RECORD>
        <RCDNUM>1</RCDNUM>
        <RCDNAME>QE_EMPLOYEE</RCDNAME>
        <CORRNAME>A</CORRNAME>
        <JOINTYPE/>
        <JOINRCDALIAS/>
        <JOINFIELD/>
        <RCDSELECTNUM>1</RCDSELECTNUM>
      </RECORD>
      <RECORD>
        <RCDNUM>2</RCDNUM>
        <RCDNAME>QE_PERS_DATA</RCDNAME>
        <CORRNAME>B</CORRNAME>
        <JOINTYPE/>
        <JOINRCDALIAS/>
        <JOINFIELD/>
        <RCDSELECTNUM>2</RCDSELECTNUM>
      </RECORD>
      <!--1 or more repetitions:-->
      <FIELD>
        <FIELDNUM>1</FIELDNUM>
        <FIELDNAME>EMPLID</FIELDNAME>
        <FIELDRCDNUM>1</FIELDRCDNUM>
        <DESCR>Emplid</DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>text</HEADING_TYPE>
        <HEADING>Employee ID</HEADING>
        <COLUMNNUM>1</COLUMNNUM>
        <ORDERBYNUM>0</ORDERBYNUM>
        <ORDERBYDIR>Ascending</ORDERBYDIR>
        <FIELDSELECTNUM>1</FIELDSELECTNUM>
        <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
      </FIELD>
      <FIELD>
        <FIELDNUM>2</FIELDNUM>
        <FIELDNAME>QE_EMPLOYEE_NAME</FIELDNAME>
        <FIELDRCDNUM>1</FIELDRCDNUM>
        <DESCR>Name</DESCR>
        <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
        <HEADING_TYPE>text</HEADING_TYPE>
        <HEADING>Name</HEADING>
        <COLUMNNUM>2</COLUMNNUM>
        <ORDERBYNUM>1</ORDERBYNUM>
        <ORDERBYDIR>Ascending</ORDERBYDIR>
        <FIELDSELECTNUM>1</FIELDSELECTNUM>
        <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
      </FIELD>
      <FIELD>
        <FIELDNUM>3</FIELDNUM>
        <FIELDNAME>QE_EMPLID</FIELDNAME>
        <FIELDRCDNUM>2</FIELDRCDNUM>
        <DESCR>qeEmplid</DESCR>
```

```

    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>text</HEADING_TYPE>
    <HEADING>Employee ID</HEADING>
    <COLUMNNUM>1</COLUMNNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <FIELDSELECTNUM>2</FIELDSELECTNUM>
    <EXPRESSION_AS_FIELD></EXPRESSION_AS_FIELD>
</FIELD>
<!--Zero or more repetitions:-->
<CRITERION>
    <CRTNUM>1</CRTNUM>
    <CRTNAME>myCrit</CRTNAME>
    <CRTHAVINGFLAG>False</CRTHAVINGFLAG>
    <CRTSELECTNUM>1</CRTSELECTNUM>
    <CRTNEGATION>False</CRTNEGATION>
    <CONDITION_TYPE>in list</CONDITION_TYPE>
    <LEFT_PARENTHESIS_LEVEL>0</LEFT_PARENTHESIS_LEVEL>
    <RIGHT_PARENTHESIS_LEVEL>0</RIGHT_PARENTHESIS_LEVEL>
    <CRTEXP1TYPE>Field</CRTEXP1TYPE>
    <CRTEXP1TEXT>1</CRTEXP1TEXT>
    <CRTEXP1NUM>1</CRTEXP1NUM>
    <CRTEXP1RCDALIAS>A</CRTEXP1RCDALIAS>
    <CRTEXP1FIELD>EMPLID</CRTEXP1FIELD>
    <CRTEXP2RCDALIAS/>
    <CRTEXP2FIELD/>
    <CRTEXP2TYPE>subquery</CRTEXP2TYPE>
    <CRTEXP2TEXT>2</CRTEXP2TEXT>
    <CRTLOGICALOPER>not used</CRTLOGICALOPER>
</CRITERION>
<CRITERION>
    <CRTNUM>2</CRTNUM>
    <CRTNAME>subCrit</CRTNAME>
    <CRTHAVINGFLAG>False</CRTHAVINGFLAG>
    <CRTSELECTNUM>2</CRTSELECTNUM>
    <CRTNEGATION>False</CRTNEGATION>
    <CONDITION_TYPE>equal to</CONDITION_TYPE>
    <LEFT_PARENTHESIS_LEVEL>1</LEFT_PARENTHESIS_LEVEL>
    <RIGHT_PARENTHESIS_LEVEL>1</RIGHT_PARENTHESIS_LEVEL>
    <CRTEXP1TYPE>Field</CRTEXP1TYPE>
    <CRTEXP1TEXT>1</CRTEXP1TEXT>
    <CRTEXP1NUM>1</CRTEXP1NUM>
    <CRTEXP1RCDALIAS>B</CRTEXP1RCDALIAS>
    <CRTEXP1FIELD>QE_HIGHLY_COMP_EMP</CRTEXP1FIELD>
    <CRTEXP2RCDALIAS/>
    <CRTEXP2FIELD/>
    <CRTEXP2TYPE>Constant</CRTEXP2TYPE>
    <CRTEXP2TEXT>E</CRTEXP2TEXT>
    <CRTLOGICALOPER>not used</CRTLOGICALOPER>
</CRITERION>
<!--1 or more repetitions:-->
<SELECT>
    <SELECTNUM>1</SELECTNUM>
    <PARENTSELECTNUM>0</PARENTSELECTNUM>
    <SELECTTYPE>Main</SELECTTYPE>
    <QRYDISTINCT>True</QRYDISTINCT>
</SELECT><SELECT>
    <SELECTNUM>2</SELECTNUM>
    <PARENTSELECTNUM>1</PARENTSELECTNUM>
    <SELECTTYPE>Subquery</SELECTTYPE>
    <QRYDISTINCT>False</QRYDISTINCT>
</SELECT>
<QUERY_NAME>QuerywithSubQuery</QUERY_NAME>
<DESCRIPTION>This is a query with a subquery</DESCRIPTION>
<DESCRLONG>Subquery example used for testing with QAS</DESCRLONG>
<QUERY_OWNER>Public</QUERY_OWNER>
</qas:QAS_QUERY_SAVE_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```


Chapter 4

Executing a Query

Understanding Query Execution

A third-party application will execute a PeopleSoft query using the same basic steps as a PeopleSoft online user. This table lists the steps to execute a query and retrieve query results.

Steps	Description
1. Select query.	QAS service operations are available to find and list existing queries.
2. Enter prompts.	QAS web services are available to retrieve query prompts and get translate values and prompt table values.
3. Filter fields.	A service operation is available to list all of the fields in a query.
4. Execute the query.	QAS service operations are available to execute a query synchronously or asynchronously.
5. Retrieve query results.	A service operation is available to retrieve the query results.
6. Cancel a query.	QAS provides a service operation to cancel query execution.
7. Retrieve query status.	QAS provides a service operation to retrieve the current query execution status.

Query Execution Models

Query Execution Services supports the following models:

Model	Description
Synchronous Request and Response	Request is sent and processed returning the response.
Synchronous Request and Synchronous Poll with Chunked Response	Request is sent synchronously and run in the Process Scheduler. Requestor will need to retrieve the results when the process has finished.
Asynchronous Request and Response	Request is sent asynchronously and queued in Integration Broker and processed. The response is returned to the http url specified in the ws addressing soap header. Requestor has the option of requesting a response when the process has finished.

Note: In order to use Integration Broker for asynchronous and synchronous poll message processing Pub/Sub must be active.

See "Monitoring Asynchronous Service Operation Transactions" (PeopleTools 8.53: Integration Broker Service Operations Monitor), "Understanding Messaging Server Administration" (PeopleTools 8.53: PeopleSoft Integration Broker Administration).

Performance Considerations

Which execution model to run depends on the number of rows and columns returned in the response. Note the following guidelines:

- Synchronous execution is not recommended for large result sets.
- Response data that consists of a large number of columns with fewer rows of data has a slower response than response data that consists of fewer columns and more rows, even though the response size may be smaller.
- Executing synchronous poll query with block size = 0 or Max will result in all data in one block.
- In synchronous poll execution, the larger the response block, the fewer blocks are returned and total response time for all blocks is shorter, but the user has to wait longer for the response.
- In synchronous poll execution, if the heap size in the web server is not sufficient to retrieve the large response block, a JAVA out of memory exception will occur.
- In synchronous poll execution, if a smaller block size (less than 100,000) is used, the result will be too many blocks containing very few rows. Optimal request block size in KB is in the range from 100,000 to 1,000,000.

Output Format and Output Type

When you execute a query using QAS service operations, the output type can be WEBROWSET, XMLP, EXCEL or HTML. The output format will be either FILE or NONFILE.

This table shows the valid combinations of output format and output type.

<i>Type</i>	<i>FILE</i>	<i>NONFILE</i>
WEBROWSET	Creates a file in webrowset format.	Creates a response in webrowset format.
XMLP	Creates an XML file. This file is suitable to use with BI Publisher reports.	Creates a response in XMLP format. <hr/> Note: Connected Query is not supported for nonfile output. <hr/>
EXCEL	Creates an XLS file.	Not valid
HTML	Creates an HTML file.	Not valid

Report Repository

Only HTTP and HTTPS are supported as the transport mechanism to publish QAS file output to the PeopleSoft Report Repository. This means that for QAS to work with file output in SYNC and ASYNC execution, a URI (universal resource identifier) scheme of HTTP or HTTPS must be defined for a server.

See "Defining HTTP Distribution Nodes" (PeopleTools 8.53: PeopleSoft Process Scheduler).

WebRowSet Format

WebRowSet represents a set of fetched rows that can be passed between tiers and components and for which the data (obtained in XML format) is managed either by storing it in a string or by writing it to a file in the local file system. PeopleSoft uses the standard WebRowSet format based on the WebRowSet schema. When you use WebRowSet, the rows (tabular data) referred to as rowsets can be read and written in XML format, thus enabling rowsets to be sent over the Internet using the HTTP/XML protocol.

WebRowSet Schema definition describes the internal data of a RowSet object in three distinct areas:

Properties	These properties describe the standard synchronization provider properties in addition to the more general rowset properties.
Metadata	The metadata describes the tabular structure governed by a WebRowSet object.
Data	The data is the current data values.

This example shows the four rows of data returned for a simple query MSGSET that has two columns. The SQL for this query is:

```
SELECT A.MESSAGE_SET_NBR, A.DESCR
FROM PSMMSGSETDEFN A
```

Example WebRowset File:

```
<?xml version='1.0'?>
<webRowSet xmlns='http://java.sun.com/xml/ns/jdbc'>
  <properties>
    <command></command>
    <concurrency>1007</concurrency>
    <datasource/>
    <escape-processing>true</escape-processing>
    <fetch-direction>1000</fetch-direction>
    <fetch-size>0</fetch-size>
    <isolation-level>1</isolation-level>
    <key-columns/>
    <map/>
    <max-field-size>0</max-field-size>
    <max-rows>0</max-rows>
    <query-timeout>0</query-timeout>
    <read-only>true</read-only>
    <rowset-type>1003</rowset-type>
    <show-deleted>false</show-deleted>
    <table-name/>
    <url></url>
    <sync-provider>
      <sync-provider-name></sync-provider-name>
      <sync-provider-vendor></sync-provider-vendor>
      <sync-provider-version></sync-provider-version>
      <sync-provider-grade></sync-provider-grade>
      <data-source-lock></data-source-lock>
    </sync-provider>
  </properties>
```

```

<metadata>
  <column-count>2</column-count>
  <column-definition>
    <column-index>1</column-index>
    <auto-increment>false</auto-increment>
    <case-sensitive>false</case-sensitive>
    <currency>false</currency>
    <nullable>false</nullable>
    <signed>false</signed>
    <searchable>false</searchable>
    <column-display-size>0</column-display-size>
    <column-label>Message Set</column-label>
    <column-name>A.MESSAGE_SET_N</column-name>
    <schema-name/>
    <column-precision>0</column-precision>
    <column-scale>0</column-scale>
    <table-name/>
    <catalog-name/>
    <column-type>2</column-type>
    <column-type-name>NUMBER</column-type-name>
  </column-definition>
  <column-definition>
    <column-index>2</column-index>
    <auto-increment>false</auto-increment>
    <case-sensitive>false</case-sensitive>
    <currency>false</currency>
    <nullable>false</nullable>
    <signed>false</signed>
    <searchable>false</searchable>
    <column-display-size>0</column-display-size>
    <column-label>Message Set Description</column-label>
    <column-name>A.DESCR</column-name>
    <schema-name/>
    <column-precision>0</column-precision>
    <column-scale>0</column-scale>
    <table-name/>
    <catalog-name/>
    <column-type>1</column-type>
    <column-type-name>CHAR</column-type-name>
  </column-definition>
</metadata>
<data>
  <currentRow>
    <columnValue>1</columnValue>
    <columnValue>
      <![CDATA[PeopleTools Message Bar Items]]>
    </columnValue>
  </currentRow>
  <currentRow>
    <columnValue>2</columnValue>
    <columnValue>
      <![CDATA[PeopleCode]]>
    </columnValue>
  </currentRow>
  <currentRow>
    <columnValue>3</columnValue>
    <columnValue>
      <![CDATA[General Tools Messages]]>
    </columnValue>
  </currentRow>
  <currentRow>
    <columnValue>4</columnValue>
    <columnValue>
      <![CDATA[Help Processor]]>
    </columnValue>
  </currentRow>
</data>
</webRowSet>

```

XMLP Format

XMLP format creates an XML file suitable for use with Oracle BI Publisher. Oracle BI Publisher expects the XML data file to consist of a root node with a set of related elements below.

This example shows the four rows of data returned for a simple query MSGSET that has two columns. The SQL for this query is:

```
SELECT A.MESSAGE_SET_NBR, A.DESCR
       FROM PMSGSETDEFN A
```

Example XMLP File:

```
<?xml version='1.0'?>
<query numRows="4" queryname="MSGSET" xmlns:xsi="http://www.w3.org/2001/XMLSchema-i⇒
nstance"
xsi:noNamespaceSchemaLocation="">
  <row rownumber="1">
    <MESSAGE_SET_NBR>1</MESSAGE_SET_NBR>
    <DESCR>PeopleTools Message Bar Items</DESCR>
  </row>
  <row rownumber="2">
    <MESSAGE_SET_NBR>2</MESSAGE_SET_NBR>
    <DESCR>PeopleCode</DESCR>
  </row>
  <row rownumber="3">
    <MESSAGE_SET_NBR>3</MESSAGE_SET_NBR>
    <DESCR>General Tools Messages</DESCR>
  </row>
  <row rownumber="4">
    <MESSAGE_SET_NBR>4</MESSAGE_SET_NBR>
    <DESCR>Help Processor</DESCR>
  </row>
</query>
```

Excel Format

Excel format creates an xls file that can be opened in Microsoft Excel.

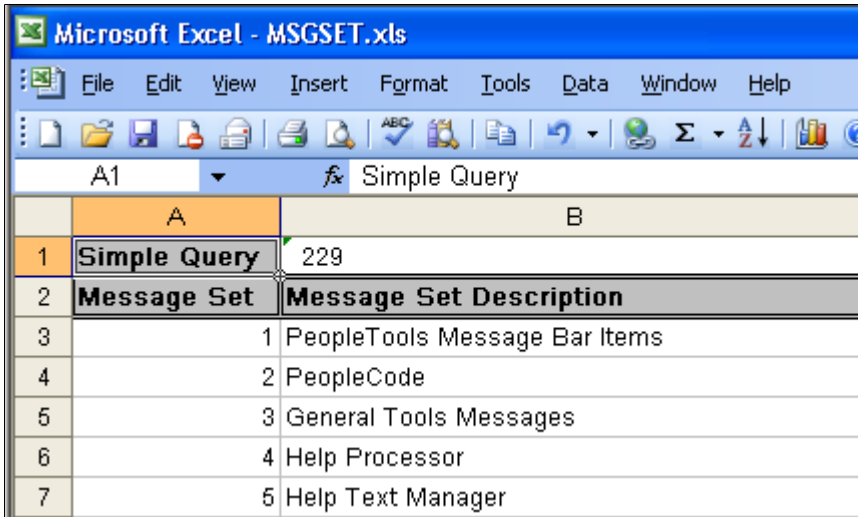
This example shows the five rows of data returned for a simple query MSGSET that has two columns. The SQL for this query is:

```
SELECT A.MESSAGE_SET_NBR, A.DESCR
       FROM PMSGSETDEFN A
```

Example XLS File:

Image: Excel format

This example illustrates the fields and controls on the Excel format.



HTML Format

HTML format creates an HTML file.

This example shows the four rows of data returned for a simple query MSGSET that has two columns. The SQL for this query is:

```
SELECT A.MESSAGE_SET_NBR, A.DESCR
FROM PMSGSETDEFN A
```

Example HTML File:

Image: HTML format

This example illustrates the fields and controls on the HTML format.



Date and Time Handling

When you create a query containing date fields, the default display in PeopleSoft Pure Internet Architecture is based on the regional setting. For example, if the personalization setting uses date format MMDDYY and date separator /, the date is displayed as MM/DD/YYYY.

See "Defining User-Specific, Locale-Based Formatting" (PeopleTools 8.53: Global Technology).

When you execute the query using one of the QAS service operations, date and time are returned in the following way:

- Date is returned to the client applications as YYYY-MM-DD. For example, April 6, 2009 is 2009-04-06.
- Time is returned as hh:mm:ss.milliseconds. For example, 9:30 a.m. is returned as 9:30:00.000000.
- Datetime is returned with the UTC (coordinated universal time) offset for the base time zone. For example, if the base time zone is PDT (Pacific Daylight Time), then April 6, 2009 9:30 a.m. is returned as 2009-04-06T09:30:00-0700.

If a date field is used as a prompt, the FieldValue must be entered as YYYY-MM-DD in the QAS execution request.

Execution Logging

HTTP transfer code is used to post file output to the PSFT Report Repository. For synchronous execution run to file, QAS HTTP transfer code logs messages to the application server log. Refer to the PeopleSoft Process Scheduler PeopleBook for information on setting logging levels and viewing log reports.

See "Logging System for PeopleSoft Process Scheduler Server" (PeopleTools 8.53: PeopleSoft Process Scheduler).

For all other debugging purposes, standard Integration Broker logs can be viewed.

See "Monitoring Asynchronous Service Operation Instances" (PeopleTools 8.53: Integration Broker Service Operations Monitor), "Understanding Synchronous Service Operation Statuses" (PeopleTools 8.53: Integration Broker Service Operations Monitor).

Selecting a Query

Once a query has been saved in the PeopleSoft application database, third-party applications can use service operations to find and list existing queries. This section describes the following service operations, which are available to select a query:

- QAS_LISTQUERY_OPER
- QAS_QUERY_DETAILS_OPER

QAS_LISTQUERY_OPER

Use this service operation to return a list of queries, along with the query descriptions, and owner type. If no queries match the search string, the response message will be empty.

Request Message: QAS_LISTQUERY_REQ_MSG

<i>Element Name</i>	<i>Description</i>
<i>All elements are optional</i>	
SearchString	Search string used for specifying the query name or the first few characters of the query name. If no value is entered, all queries for the user will be returned.
OwnerType	Optionally enter either <i>public</i> or <i>private</i> for the owner type. If no value is entered, all queries both public and private for the user will be returned.
MaxRows	Optionally enter the maximum number of rows to return.
isConnectedQry	Enter <i>Y</i> to return list of connected queries; enter <i>N</i> to return list of queries. If the no value is entered, a list of all queries, including connected queries, is returned.

Example Request:

This request will retrieve a list of all public queries that start with *XRFW*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTQUERY_REQ_MSG.VERSION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTQUERY_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_LISTQUERY_REQ_MSG>
      <!--Zero or more repetitions:-->
      <qas:QAS_LISTQUERY_REQ>
        <qas1:PTQASWRK class="R">
          <!--Optional:-->
          <qas1:SearchString>XRFW</qas1:SearchString>
          <!--Optional:-->
          <qas1:OwnerType>public</qas1:OwnerType>
          <!--Optional:-->
          <qas1:MaxRows></qas1:MaxRows>
          <!--Optional:-->
          <qas:isConnectedQry></qas:isConnectedQry>
        </qas1:PTQASWRK>
      </qas:QAS_LISTQUERY_REQ>
    </qas:QAS_LISTQUERY_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_LISTQUERY_RESP_MSG

<i>Element Name</i>	<i>Description</i>
QueryName	Query name.
Description	Query description.

Element Name	Description
OwnerType	Query owner type.
isConnectedQry	Connected query.

Example Response:

```
<?xml version="1.0"?>
<QAS_LISTQUERY_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS⇒
_LISTQUERY_RESP_MSG.VERSION_1">
  <QAS_LISTQUERY_RESP>
    <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas⇒
/QAS_LISTQUERY_RESP.VERSION_1">
      <QueryName>XRFWIN</QueryName>
      <OwnerType>public</OwnerType>
      <Description>XRFWIN</Description>
      <isConnectedQry>N</isConnectedQry>
    </PTQASWRK>
  </QAS_LISTQUERY_RESP>
  <QAS_LISTQUERY_RESP>
    <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas⇒
/QAS_LISTQUERY_RESP.VERSION_1">
      <QueryName>XRFWNFL</QueryName>
      <OwnerType>public</OwnerType>
      <Description>XRFWNFL</Description>
      <isConnectedQry>N</isConnectedQry>
    </PTQASWRK>
  </QAS_LISTQUERY_RESP>
</QAS_LISTQUERY_RESP_MSG>
```

QAS_QUERY_DETAILS_OPER

Use this service operation to return the complete details of an existing query in XML format.

Request Message: QAS_QUERY_DETAILS_REQ_MSG

Element Name	Description
QUERY_NAME Required element	Complete query name. Required.
IS_CONNECTED_QUERY Required element	Indicates that this is connected query. Valid values are <i>Y</i> and <i>N</i> . The default value is <i>N</i> .

Example Request:

This request will return the query details for the query *XRFWIN*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q⇒
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERY_DETAILS_REQ_MSG.VERS⇒
ION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERY_DETAILS_REQ_MSG>
      <QUERY_NAME>QUERY_WITH_EXPRESSION</QUERY_NAME>
    </qas:QAS_QUERY_DETAILS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_QUERY_DETAILS_RESP_MSG

The response will include the details for the query definition. This message has several groupings, and depending on the specific query, some groupings may appear multiple times, while other groupings do not apply. The elements are listed here in groups.

<Properties> Returns the properties of the query definition.

Properties Elements	Description
QUERY_NAME	Query name.
DESCRIPTION	Query description.
DESCRLONG	Long description
QUERY_OWNER	Query owner type.
CREATION_TIME	Time of query creation in PeopleSoft database.
LAST_UPDATE_TIME	Time of most recent update of the query in PeopleSoft database. Format is YYYY-MM-DD-HH.MM.SS.FF where FF is the fractional part of a second.

<Selects> Returns the select information of the query definition.

Select Elements	Description
SELECTNUM	SELECT number.
PARENTSELECTNUM	Number of parent SELECT.
SELECTTYPE	Type of selection, <i>Main, Subquery, or Union</i> .
QRYDISTINCT	Indicates whether this query is distinct.

<Records> Returns the record correlation and join information of the query definition.

Records Elements	Description
RCDNUM	Record number.
RCDNAME	Record name.
RCDSELECTNUM	Select number.
CORRNAME	Alias name such as <i>A,B,C</i> , and so on.

Records Elements	Description
JOINTYPE	Join type.
JOINRCDALIAS	Join record alias.
JOINFIELD	Join field.

<Fields> Returns the field of the query definition.

Fields Elements	Description
FIELDNUM	Field number in the corresponding select.
FIELDNAME	Field name.
DESCR	Description.
FORMAT	Field format as a combination of field type and field length.
AGGREGATE_TYPE	Aggregation type.
HEADING_TYPE	Heading type.
HEADING	Heading text.
FIELDSELECTNUM	Identifier of SELECT in which this field is included.
ORDERBYNUM	Order by number.
COLUMNNUM	Column number.
ORDERBYDIR	Direction of field ordering.
FIELDTYPE	Field type.
FIELDRCDALIAS	Alias of the record to which this field belongs.

<Criteria> Returns the criteria information of the query definition.

Criteria Elements	Description
CRTNUM	Criterion number in the corresponding select statement.
CRTTYPE	Specifies whether the HAVING clause is used.
CRTNAME	Criterion name.
CRTSELECTNUM	Select Number.

Criteria Elements	Description
NEGATION	Returns <i>True</i> if NOT is specified in the operator; otherwise, returns <i>False</i> .
CONDITION_TYPE	Condition type in criterion.
LEFT_PARENTHESIS_LEVEL	Left parenthesis level specified from 0 onwards.
RIGHT_PARENTHESIS_LEVEL	Right parenthesis level specified from 0 onwards.
CRTEXP1TYPE	Specifies whether the criterion is <i>Field</i> or <i>Expression</i> .
CRTEXP1RCDALIAS	Alias for record used in expression 1 criterion, such as A, B, and so on.
CRTEXP1FIELD	Field used in expression 1 criterion.
CRTEXP1TEXT	Expression text for criterion 1.
CRTEXP1NUM	Expression number for the criterion.
CRTEXP2RCDALIAS	Alias for record used in expression 2 criterion.
CRTEXP2FIELD	Field used in expression 2 criterion.
CRTEXP2TYPE	Expression 2 type in criterion.
CRTEXP2TEXT	Expression 2 text.
CRTLOGICALOPER	Logical operator that links the criteria.

<Expression> Returns the expression information of the query definition.

Expression Elements	Description
EXPNUM	Expression number.
EXPSELECTNUM	Expression's SELECT number.
EXPNAME	Expression name.
EXPTYPE	Expression type.
EXPLENGTH	Expression length.
EXPDECIMALPOS	Number of decimal places.
EXPTEXT	Expression text.

Expression Elements	Description
EXPAGGREGATE	Specifies whether the expression is an aggregate function.

<Prompts> Returns the prompt information of the query definition.

Prompt Elements	Description
PROMPT_NUM	Prompt number.
PROMPT_NAME	Unique prompt name.
PROMPT_UNIQUE_NAME	Unique prompt name.
PROMPT_FLDNAME	Name of the field used as prompt.
PROMPT_TABLE	Prompt table name.
PROMPT_EDITTYPE	Edit type.
PROMPT_FLDFORMAT	Field format.
PROMPT_FLDLENGTH	Field length for prompt.
PROMPT_FLDDECIMALPOS	Number of decimal places in prompt.
PROMPT_FLDDTYPE	Field type.
PROMPT_HEADING	Heading of prompt.
PROMPT_HEADINGTYPE	Type of prompt heading.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qcs:QAS_QUERY_DETAILS_RESP_MSG xmlns:qcs="http://xmlns.oracle.com/Enterprise=
/Tools/schemas/QAS_QUERY_DETAILS_RESP_MSG.VERSION_1">
      <QUERY_NAME>QUERY_WITH_EXPRESSION</QUERY_NAME>
      <DESCRIPTION>Example query</DESCRIPTION>
      <DESCRLONG>Sample query with expression, criterion and prompt</DESCRLONG>
      <QUERY_OWNER>QEDMO, Public</QUERY_OWNER>
      <CREATION_TIME>2009-05-15-09.48.15.000000</CREATION_TIME>
      <LAST_UPDATE_TIME>2009-05-15-09.48.15.546000</LAST_UPDATE_TIME>
      <PROMPT>
        <PROMPT_NUM>1</PROMPT_NUM>
        <PROMPT_NAME>QE_ORDER_NBR</PROMPT_NAME>
        <PROMPT_FLDNAME>QE_ORDER_NBR</PROMPT_FLDNAME>
        <PROMPT_UNIQUE_NAME>BIND1</PROMPT_UNIQUE_NAME>
        <PROMPT_EDITTYPE>Prompt table</PROMPT_EDITTYPE>
        <PROMPT_TABLE>QEORDER_HDR</PROMPT_TABLE>
        <PROMPT_FLDFORMAT>Upper</PROMPT_FLDFORMAT>
        <PROMPT_FLDDECIMALPOS>0</PROMPT_FLDDECIMALPOS>
        <PROMPT_FLDLENGTH>8</PROMPT_FLDLENGTH>
      </PROMPT>
    </qcs:QAS_QUERY_DETAILS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    <PROMPT_FLDTYPE>Character</PROMPT_FLDTYPE>
    <PROMPT_HEADING>Order</PROMPT_HEADING>
    <PROMPT_HEADINGTYPE>RFT Short</PROMPT_HEADINGTYPE>
</PROMPT>
<SELECT>
    <SELECTNUM>1</SELECTNUM>
    <PARENTSELECTNUM>0</PARENTSELECTNUM>
    <SELECTTYPE>Main</SELECTTYPE>
    <QRYDISTINCT>False</QRYDISTINCT>
</SELECT>
<RECORD>
    <RCDNUM>1</RCDNUM>
    <RCDNAME>QEORDER_DTL</RCDNAME>
    <RCDSELECTNUM>1</RCDSELECTNUM>
    <CORRNAME>A</CORRNAME>
</RECORD>
<FIELD>
    <FIELDNUM>1</FIELDNUM>
    <FIELDNAME>QE_ORDER_NBR</FIELDNAME>
    <DESCR>Order Number</DESCR>
    <FORMAT>CHAR 8</FORMAT>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Short</HEADING_TYPE>
    <HEADING>Order</HEADING>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <COLUMNNUM>1</COLUMNNUM>
    <FIELDTYPE>Character</FIELDTYPE>
    <FIELDRCDALIAS>A</FIELDRCDALIAS>
</FIELD>
<FIELD>
    <FIELDNUM>2</FIELDNUM>
    <FIELDNAME>QE_ORDER_LINE_NBR</FIELDNAME>
    <DESCR>Order Line Number</DESCR>
    <FORMAT>NUMBER 3</FORMAT>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Short</HEADING_TYPE>
    <HEADING>Order Line</HEADING>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <COLUMNNUM>2</COLUMNNUM>
    <FIELDTYPE>Number</FIELDTYPE>
    <FIELDRCDALIAS>A</FIELDRCDALIAS>
</FIELD>
<FIELD>
    <FIELDNUM>3</FIELDNUM>
    <FIELDNAME>QE_QTY</FIELDNAME>
    <DESCR>Quantity</DESCR>
    <FORMAT>SIGNED 17</FORMAT>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Short</HEADING_TYPE>
    <HEADING>Qty</HEADING>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <COLUMNNUM>3</COLUMNNUM>
    <FIELDTYPE>Signed Number</FIELDTYPE>
    <FIELDRCDALIAS>A</FIELDRCDALIAS>
</FIELD>
<FIELD>
    <FIELDNUM>4</FIELDNUM>
    <FIELDNAME>QE_PRICE</FIELDNAME>
    <DESCR>Price</DESCR>
    <FORMAT>NUMBER 9</FORMAT>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>RFT Short</HEADING_TYPE>
    <HEADING>Price</HEADING>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <ORDERBYNUM>0</ORDERBYNUM>

```

```

    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <COLUMNNUM>4</COLUMNNUM>
    <FIELDTYPE>Number</FIELDTYPE>
    <FIELDRCDALIAS>A</FIELDRCDALIAS>
  </FIELD>
  <FIELD>
    <FIELDNUM>5</FIELDNUM>
    <FIELDNAME>EXPR5_5</FIELDNAME>
    <DESCR/>
    <FORMAT>NUMBER 11</FORMAT>
    <AGGREGATE_TYPE>None</AGGREGATE_TYPE>
    <HEADING_TYPE>Text</HEADING_TYPE>
    <HEADING>A.QE_QTY*A.QE_PRICE</HEADING>
    <FIELDSELECTNUM>1</FIELDSELECTNUM>
    <ORDERBYNUM>0</ORDERBYNUM>
    <ORDERBYDIR>Ascending</ORDERBYDIR>
    <COLUMNNUM>5</COLUMNNUM>
    <FIELDTYPE>Number</FIELDTYPE>
    <FIELDRCDALIAS/>
  </FIELD>
  <CRITERION>
    <CRTNUM>1</CRTNUM>
    <CRTSELECTNUM>1</CRTSELECTNUM>
    <NEGATION>False</NEGATION>
    <LEFT_PARENTHESIS_LEVEL>0</LEFT_PARENTHESIS_LEVEL>
    <RIGHT_PARENTHESIS_LEVEL>0</RIGHT_PARENTHESIS_LEVEL>
    <CONDITION_TYPE>equal to</CONDITION_TYPE>
    <CRTEXP1TYPE>Field</CRTEXP1TYPE>
    <CRTEXP1RCDALIAS>A</CRTEXP1RCDALIAS>
    <CRTEXP1FIELD>QE_ORDER_NBR</CRTEXP1FIELD>
    <CRTEXP2TYPE>Prompt</CRTEXP2TYPE>
    <CRTEXP2TEXT>:1</CRTEXP2TEXT>
    <CRTLOGICALOPER>not used</CRTLOGICALOPER>
  </CRITERION>
  <EXPRESSION>
    <EXPNUM>1</EXPNUM>
    <EXPSELECTNUM>1</EXPSELECTNUM>
    <EXPTYPE>Number</EXPTYPE>
    <EXPTEXT>A.QE_QTY*A.QE_PRICE</EXPTEXT>
    <EXPLENGTH>11</EXPLENGTH>
    <EXPDECIMALPOS>2</EXPDECIMALPOS>
    <EXPAGGREGATE>False</EXPAGGREGATE>
  </EXPRESSION>
</qcs:QAS_QUERY_DETAILS_RESP_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Listing Fields

This section describes the service operation that is available for listing the fields in a query.

QAS_LISTQUERYFIELDS

Use this service operation to return a list of fields for a given query. This service operation will help the user discover the unique field names to use in the FilterFieldName when the query is executed.

Request Message: QAS_LISTQUERYFIELDS_REQ_MSG

<i>Element Name</i>	<i>Description</i>
QueryName Required element	Complete query name. Required.
OwnerType	Indicate whether it is a <i>public</i> or <i>private</i> query. If no value is specified, QAS will first look for a private query and if it is not found, will look for a public query.
IsconnectedQuery Required element	Indicate whether this is connected query. Required. Valid values are <i>Y</i> and <i>N</i> .

Example Request:

This request will return the fields for the query *XRFWIN*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTQUERYFIELDS_REQ_MSG.VERSION 1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTQUERYFIELDS_REQ.VERSION 1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_LISTQUERYFIELDS_REQ_MSG>
      <!--Zero or more repetitions:-->
      <qas:QAS_LISTQUERYFIELDS_REQ>
        <qas1:PTQASWRK class="R">
          <qas1:QueryName>XRFWIN</qas1:QueryName>
          <!--Optional:-->
          <qas1:OwnerType>Public</qas1:OwnerType>
          <qas1:isConnectedQuery>n</qas1:isConnectedQuery>
        </qas1:PTQASWRK>
      </qas:QAS_LISTQUERYFIELDS_REQ>
    </qas:QAS_LISTQUERYFIELDS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_LISTQUERYFIELDS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
ColumnNumber	Column number.
FieldName	Field name.
FieldType	Field type.
FieldLength	Field length.
FieldDecimal	Number of decimal positions.

Element Name	Description
HeadingText	Unique heading text.
UniqueFieldName	Unique field name. Unique field names are used to set up field filtering when the query is executed.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_LISTQUERYFIELDS_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools=
/schemas/QAS_LISTQUERYFIELDS_RESP_MSG.VERSION_1">
      <QAS_LISTQUERYFIELDS_RESP>
        <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch=
emas/QAS_LISTQUERYFIELDS_RESP.VERSION_1">
          <QueryName>XRFWIN</QueryName>
          <PTQASFIELDWRK class="R">
            <ColumnNumber>1</ColumnNumber>
            <FieldName>MENUNAME</FieldName>
            <FieldType>string</FieldType>
            <FieldLength>30</FieldLength>
            <FieldDecimal>0</FieldDecimal>
            <HeadingText>Menu Name</HeadingText>
            <UniqueFieldName>Menu Name</UniqueFieldName>
          </PTQASFIELDWRK>
        </PTQASWRK>
      </QAS_LISTQUERYFIELDS_RESP>
    </QAS_LISTQUERYFIELDS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Entering Prompts

This section describes the service operations that are available for discovering prompt and valid prompt values for a query. These include:

- QAS_LISTQUERYPROMPTS_OPER
- QAS_GETPROMPTTABLEVALUES_OPER
- QAS_GETXLAT_OPER

QAS_LISTQUERYPROMPTS_OPER

Use this service operation to retrieve a list of the prompts and associated prompt table for a specific query.

Request Message: QAS_LISTQUERYPROMPTS_REQ_MSG

<i>Element Name</i>	<i>Description</i>
QueryName Required element	Complete Query Name. Required.
OwnerType	Optionally, enter the query owner type. If no value is entered, both public and private are searched.
isConnectedQuery Required element	Indicate whether this is a connected query. Required. Valid values are Y and N.

Example Request:

This request will return the prompt field and prompt table for the query TWO_PROMPT_QUERY.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTQUERYPROMPTS_REQ_MSG.VERSION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTQUERYPROMPTS_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_LISTQUERYPROMPTS_REQ_MSG>
      <qas:QAS_LISTQUERYPROMPTS_REQ>
        <qas1:PTQASWRK class="R">
          <qas1:QueryName>TWO_PROMPT_QUERY</qas1:QueryName>
          <!--Optional:-->
          <qas1:OwnerType>PUBLIC</qas1:OwnerType>
          <qas1:isConnectedQuery>n</qas1:isConnectedQuery>
        </qas1:PTQASWRK>
      </qas:QAS_LISTQUERYPROMPTS_REQ>
    </qas:QAS_LISTQUERYPROMPTS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_LISTQUERYPROMPTS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
QueryName	Query name.
FieldName	If this prompt uses a translate table, the field name for the prompt is returned.
HeadingText	Heading text for the prompt.
FieldType	Prompt field type.
FormatText	Format for the text.
FieldLength	Field length for the prompt.

Element Name	Description
FieldDecimal	Number of decimal places in the prompt field.
UniquePromptName	The unique name used for the prompt. This value is case-sensitive.
EditType	The type of edit: <ul style="list-style-type: none"> • <i>Translate table</i> • <i>Yes/No table</i> • <i>Prompt table</i> • <i>No table edit</i>
PromptTable	If this prompt uses a prompt table, the table name is returned.

Example Response:

This response has two prompts, one for a prompt table and one for a translate table edit.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_LISTQUERYPROMPTS_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tool
s/schemas/QAS_LISTQUERYPROMPTS_RESP_MSG.VERSION_1">
      <QAS_LISTQUERYPROMPTS_RESP>
        <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch
emas/QAS_LISTQUERYPROMPTS_RESP.VERSION_1">
          <QueryName>TWO_PROMPT_QUERY</QueryName>
          <PTQASPRMPTWRK class="R">
            <FieldName>NODE_TYPE</FieldName>
            <HeadingText>Node Type</HeadingText>
            <FieldType>string</FieldType>
            <FormatText>upper</FormatText>
            <FieldLength>30</FieldLength>
            <FieldDecimal>0</FieldDecimal>
            <UniquePromptName>BIND1</UniquePromptName>
            <EditType>Translate table</EditType>
            <PromptTable/>
          </PTQASPRMPTWRK>
        </PTQASWRK>
        <PTQASPRMPTWRK class="R">
          <FieldName>PORTAL_TYPE</FieldName>
          <HeadingText>Portal</HeadingText>
          <FieldType>string</FieldType>
          <FormatText>upper</FormatText>
          <FieldLength>30</FieldLength>
          <FieldDecimal>0</FieldDecimal>
          <UniquePromptName>BIND2</UniquePromptName>
          <EditType>Prompt table</EditType>
          <PromptTable>PSPRDMDEFN</PromptTable>
        </PTQASPRMPTWRK>
      </PTQASWRK>
    </QAS_LISTQUERYPROMPTS_RESP>
  </QAS_LISTQUERYPROMPTS_RESP_MSG>
</soapenv:Body>
</soapenv:Envelope>
```

QAS_GETPROMPTTABLEVALUES_OPER

Use this service operation to return a list of field values for a given prompt table.

Request Message: QAS_GETPRMPTTBLVAL_REQ_MSG

<i>Element Name</i>	<i>Description</i>
PromptTableName Required element	Complete prompt record name. Required.
FieldValue	Optional search string for the key field value. If no value is entered, a list of all key values will be returned. Note: Multiple keys are not supported in PS Query. Tables with no keys are not supported in PS Query.
MaxRows	Optionally, enter the maximum number of rows to be returned. MaxRows is a number of length 10 with a decimal position of 0.

Example Request:

This request will return the first three rows of prompt table values for the table *PSMSGSETDEFN*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_GETPRMPTTBLVAL_REQ_MSG.VERSION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_GETPRMPTTBLVAL_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_GETPRMPTTBLVAL_REQ_MSG>
      <!--Zero or more repetitions:-->
      <qas:QAS_GETPRMPTTBLVAL_REQ>
        <qas1:PTQASWRK class="R">
          <qas1:PromptTableName>psmsgsetdefn</qas1:PromptTableName>
          <!--Optional:-->
          <qas1:MaxRows>3</qas1:MaxRows>
          <!--Optional:-->
          <qas1:FieldName></qas1:FieldName>
        </qas1:PTQASWRK>
      </qas:QAS_GETPRMPTTBLVAL_REQ>
    </qas:QAS_GETPRMPTTBLVAL_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_GETPRMPTTBLVALUES_RESP_MSG

<i>Element Name</i>	<i>Description</i>
FieldValue	Field value.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=>
```

```

oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_GETPRMPTTBVLVAL_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/
schemas/QAS_GETPRMPTTBVLVAL_RESP_MSG.VERSION_1">
      <QAS_GETPRMPTTBVLVAL_RESP>
        <PTQASFIELDWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tool
s/schemas/QAS_GETPRMPTTBVLVAL_RESP.VERSION_1">
          <FieldValue>1</FieldValue>
        </PTQASFIELDWRK>
      </QAS_GETPRMPTTBVLVAL_RESP>
      <QAS_GETPRMPTTBVLVAL_RESP>
        <PTQASFIELDWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tool
s/schemas/QAS_GETPRMPTTBVLVAL_RESP.VERSION_1">
          <FieldValue>2</FieldValue>
        </PTQASFIELDWRK>
      </QAS_GETPRMPTTBVLVAL_RESP>
      <QAS_GETPRMPTTBVLVAL_RESP>
        <PTQASFIELDWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tool
s/schemas/QAS_GETPRMPTTBVLVAL_RESP.VERSION_1">
          <FieldValue>3</FieldValue>
        </PTQASFIELDWRK>
      </QAS_GETPRMPTTBVLVAL_RESP>
    </QAS_GETPRMPTTBVLVAL_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

QAS_GETXLAT_OPER

Use this service operation to return the translate values for a given field.

Request Message: QAS_GETXLAT_REQ_MSG

Element Name	Description
FieldName	Complete field name. Required.

Example Request:

This request will return the translate value for *EFF_STATUS*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_GETXLAT_REQ_MSG.VERSION_1"
xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_GETXLAT_REQ.VERSI
ON_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_GETXLAT_REQ_MSG>
      <!--Zero or more repetitions:-->
      <qas:QAS_GETXLAT_REQ>
        <qas1:PTQASWRK class="R">
          <qas1:FieldName>EFF_STATUS</qas1:FieldName>
        </qas1:PTQASWRK>
      </qas:QAS_GETXLAT_REQ>
    </qas:QAS_GETXLAT_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_GETXLAT_RESP_MSG

<i>Element Name</i>	<i>Description</i>
FieldValue	Translate value.
ShortName	Short name for the translate value.
LongName	Long name for the translate value.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_GETXLAT_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas
/QAS_GETXLAT_RESP_MSG.VERSION_1">
      <QAS_GETXLAT_RESP>
        <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch
emas/QAS_GETXLAT_RESP.VERSION_1">
          <FieldValue>A</FieldValue>
          <ShortName>Active</ShortName>
          <LongName>Active</LongName>
        </PTQASWRK>
      </QAS_GETXLAT_RESP>
      <QAS_GETXLAT_RESP>
        <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch
emas/QAS_GETXLAT_RESP.VERSION_1">
          <FieldValue>I</FieldValue>
          <ShortName>Inactive</ShortName>
          <LongName>Inactive</LongName>
        </PTQASWRK>
      </QAS_GETXLAT_RESP>
    </QAS_GETXLAT_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Executing the Query

This section provides an overview of the XML string and discusses the service operations to execute a query. These include:

- QAS_EXECUTEQRYSYNC_OPER
- QAS_EXECUTEQRYSYNCPOLL_OPER
- QAS_EXECUTEQRYSYNC_ASYNC_OPER

XML String

When you request to execute a query, you can either select the query name if it exists in the database, or use the XML string to request an adhoc query. The same elements are used to define the XML string as defining the query to save. If you use XML string, it must contain at least one record, one field and one select.

<Records> Request must contain at least one record.

Records Elements	Description
RCDNUM	Record number.
RCDNAME	Record name.
RCDSELECTNUM	Select number.
CORRNAME	Alias Name such as <i>A,B,C</i> , and so on.
JOINTYPE	Type of join. Valid values are: <ul style="list-style-type: none"> • None • Hierarchy • Related • Tree • LeftOuter • RelatedLeftOuter Value is required only when the corresponding record is part of a JOIN in the query.
JOINRCDALIAS	Record with which join is done. Value is required only when the corresponding record is part of a JOIN in the query.
JOINFIELD	Field with which join is done. Value is required only when the corresponding record is part of a JOIN in the query.

<Fields> Request must contain at least one field.

Fields Elements	Description
FIELDNUM	Field number.
FIELDNAME	Field name.
FIELDRCDDNUM	Record number.
DESCR	Description.
AGGREGATE_TYPE	Aggregation type. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER. <hr/>
HEADING_TYPE	Heading type. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER. <hr/>

Fields Elements	Description
HEADING	If the HEADING_TYPE is <i>Heading</i> , this element is used to enter the heading text.
COLUMNNUM	Column number.
ORDERBYNUM	Order by number.
ORDERBYDIR	Direction of field ordering. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER.
FIELDSELECTNUM	Identifier of SELECT in which this field is included.
EXPRESSION_AS_FIELD	A valid expression number when an expression is used as a field.

<Criteria> Enter criteria information.

Criteria Elements	Description
CRTNUM	Criterion number.
CRTNAME	Criterion name.
CRTHAVINGFLAG	True if specifying a HAVING clause. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER.
CRTSELECTNUM	Selectnumber.
CRTNEGATION	Negation in criterion. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER.
CONDITION_TYPE	Condition type in criterion. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER.
LEFT_PARENTHESIS_LEVEL	Left parenthesis level specified from 0 onwards. Used for GROUP BY.
RIGHT_PARENTHESIS_LEVEL	Right parenthesis level specified from 0 onwards. Used for GROUP BY.
CRTEXP1TYPE	Expression 1 type in criterion.
CRTEXP1TEXT	Expression text for criterion 1.
CRTEXP1RCDALIAS	Record alias for record used in expression 1 in criterion.
CRTEXP1FIELD	Field used in expression 1 for criterion.

Criteria Elements	Description
CRTEXP2RCDALIAS	Record alias for record used in expression 2 for criterion.
CRTEXP2FIELD	Field used in expression 2 for criterion.
CRTEXP2TYPE	Expression 2 type in criterion. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER. <hr/>
CRTEXP2TEXT	Expression 2 text.
CRTLOGICALOPER	Logical operator that links the criteria. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER. <hr/>

<Expressions> Enter expression information.

Expressions Elements	Description
EXPNUM	Expression Number.
EXPSELECTNUM	Expression's SELECT Number.
EXPNAME	Expression Name.
EXPTYPE	Expression Type. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER. <hr/>
EXPLENGTH	Expression Length.
EXPDECIMALPOS	Number of Decimal Places.
EXPTEXT	Expression Text.
EXPAGGREGATE	Specifies if the expression is an aggregate function.

<Prompts> Enter prompt information.

Prompt Elements	Description
PROMPT_NUM	Prompt number.
PROMPT_NAME	Unique prompt name. This is the UniquePromptName value returned from the service operation QAS_LISTQUERYPROMPTS_OPER.
PROMPT_FLDNAME	Name of the field used as prompt.
PROMPT_UNIQUE_NAME	Unique prompt name.

Prompt Elements	Description
PROMPT_TABLE	Prompt table
PROMPT_EDITTYPE	Edit type. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER.
PROMPT_HEADING	Heading of prompt.
PROMPT_HEADINGTYPE	Type of prompt heading. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER.
PROMPT_FLDLENGTH	Field length for prompt.
PROMPT_FLDDECIMALPOS	Number of decimal positions in prompt.

<Select> XML String must include at least one select.

Select Elements	Description
SELECTNUM	SELECT number.
PARENTSELECTNUM	Number of parent SELECT. For main SELECT, this must be set to 0.
SELECTTYPE	Type of selection. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER.
QRYDISTINCT	Indicate whether this query is distinct. <hr/> Note: For valid values, see QAS_SAVE_QUERY_OPER.

QAS_EXECUTEQRYSYNC_OPER

Use this service operation to synchronously execute a query and receive the query results in the format selected in the service operation request. The response message depends on the request was for a file or a non file format.

Request Message: QAS_EXEQRY_SYNC_REQ_MSG

Element Name	Description
QueryName	Query name. Required unless you are using an XML string.
Required element unless you are using XML string	

Element Name	Description
isConnectedQuery Required element	Indicate whether this is a connected query. Valid values are <i>Y</i> or <i>N</i> in either upper or lower case. <hr/> Note: The default value is <i>N</i> . <hr/>
XMLString	XML string. <hr/> Note: XML string is used only for ad hoc queries. See XML String for the elements. <hr/>
OwnerType Required element	Query owner type, <i>public</i> or <i>private</i> in either uppercase or lowercase. Required.
BlockSizeKB Required element	Blocksize is used only for synchronous poll. For synchronous execution, the value should be set to 0 (zero). The default value is 0.
MaxRows Required element	The maximum number of rows to be fetched.
OutResultType Required element	Select the output type. Valid values are: <ul style="list-style-type: none"> • WEBROWSET • XMLP • EXCEL • HTML <hr/> Note: For connected queries, the output type must be XMLP. <hr/> See Output Format and Output Type .
OutResultFormat Required element	Select the output format. Valid values are: <ul style="list-style-type: none"> • FILE • NONFILE <hr/> Note: For connected queries, the output format must be FILE. <hr/>

<Prompts> These elements are used for Connected Query or queries containing prompts.

Prompt Elements	Description
PSQueryName	This field is used only for Connected Query. Enter the name of the query containing the prompt.
UniquePromptName	Unique prompt name defined in the query. This value is case-sensitive. Use the service operation QAS_LISTQUERYPROMPTS_OPER to find the unique prompt name.
FieldValue	Field value for the prompt. This value is case-sensitive. <hr/> Note: Date fields require the date format as YYYY-MM-DD. <hr/>

<FieldFilter> If you want to return only specific fields in the query, indicate each field you want returned.

FilterFieldName Element	Description
FilterFieldName	List of field names to be returned. This value is case sensitive and must be the unique field name as returned by the service operation QAS_LISTQUERYFIELDS.

Example Request:

This request will execute the query *MSGCAT_PROMPT*, the unique name for the prompt is *Set* and the prompt value is *2*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_EXEQRY_SYNC_REQ_MSG.VERSION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_EXEQRY_SYNC_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_EXEQRY_SYNC_REQ_MSG>
      <qas1:QAS_EXEQRY_SYNC_REQ>
        <QueryName>MSGCAT_PROMPT</QueryName>
        <isConnectedQuery>N</isConnectedQuery>
        <OwnerType>PUBLIC</OwnerType>
        <BlockSizeKB>0</BlockSizeKB>
        <MaxRow>2</MaxRow>
        <OutResultType>XMLP</OutResultType>
        <OutResultFormat>FILE</OutResultFormat>
        <Prompts>
          <!--Zero or more repetitions:-->
          <PROMPT>
            <PSQueryName></PSQueryName>
            <UniquePromptName>BIND1</UniquePromptName>
            <FieldValue>2</FieldValue>
          </PROMPT>
        </Prompts>
      </qas1:QAS_EXEQRY_SYNC_REQ>
    </qas:QAS_EXEQRY_SYNC_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_QUERYRESULTS_FILE_RESP

This response is used when the output format requested is *FILE*.

<i>Element Name</i>	<i>Description</i>
QueryResults	File URL

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_GETQUERYRESULTS_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools
/schemas/QAS_GETQUERYRESULTS_RESP_MSG.VERSION_1">
      <QAS_QUERYRESULTS_FILE_RESP xmlns="http://xmlns.oracle.com/Enterprise/Tool
s/schemas/QAS_QUERYRESULTS_FILE_RESP.VERSION_1">
        <FILEURL>http://ple-infodev-08.peoplesoft.com:8010/psreports/QEDMO/9999
957/MSGCAT_PROMPT.xml</FILEURL>
      </QAS_QUERYRESULTS_FILE_RESP>
    </QAS_GETQUERYRESULTS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_QUERYRESULTS_WRS_RESP

This response is used when the requested output format is *NONFILE* and the output type is *WEBBROWSET*.

<i>Element Name</i>	<i>Description</i>
QueryResults	Query Result in webrowset format.

Example Response for webrowset:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_GETQUERYRESULTS_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools
/schemas/QAS_GETQUERYRESULTS_RESP_MSG.VERSION_1">
      <webRowSet xmlns="http://java.sun.com/xml/ns/jdbc">
        <properties>
          <escape-processing>true</escape-processing>
          <fetch-direction>1000</fetch-direction>
          <fetch-size>0</fetch-size>
          <isolation-level>1</isolation-level>
          <key-columns/>
          <map/>
          <max-field-size>0</max-field-size>
          <max-rows>0</max-rows>
          <query-timeout>0</query-timeout>
          <read-only>true</read-only>
          <show-deleted>false</show-deleted>
          <table-name/>
        </properties>
        <metadata>
          <column-count>5</column-count>
          <column-definition>
            <column-index>1</column-index>
            <auto-increment>false</auto-increment>
            <case-sensitive>false</case-sensitive>
          </column-definition>
        </metadata>
      </webRowSet>
    </QAS_GETQUERYRESULTS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

```

<currency>>false</currency>
<nullable>0</nullable>
<signed>>false</signed>
<searchable>>false</searchable>
<column-display-size>0</column-display-size>
<column-label>Set</column-label>
<column-name>A.MESSAGE_SET_NBR</column-name>
<schema-name/>
<column-precision>0</column-precision>
<column-scale>0</column-scale>
<table-name/>
<catalog-name/>
<column-type>4</column-type>
<column-type-name>INTEGER</column-type-name>
</column-definition>
<column-definition>
<column-index>2</column-index>
<auto-increment>>false</auto-increment>
<case-sensitive>>false</case-sensitive>
<currency>>false</currency>
<nullable>0</nullable>
<signed>>false</signed>
<searchable>>false</searchable>
<column-display-size>0</column-display-size>
<column-label>Descr</column-label>
<column-name>A.DESCR</column-name>
<schema-name/>
<column-precision>0</column-precision>
<column-scale>0</column-scale>
<table-name/>
<catalog-name/>
<column-type>12</column-type>
<column-type-name>VARCHAR</column-type-name>
</column-definition>
<column-definition>
<column-index>3</column-index>
<auto-increment>>false</auto-increment>
<case-sensitive>>false</case-sensitive>
<currency>>false</currency>
<nullable>0</nullable>
<signed>>false</signed>
<searchable>>false</searchable>
<column-display-size>0</column-display-size>
<column-label>Msg</column-label>
<column-name>B.MESSAGE_NBR</column-name>
<schema-name/>
<column-precision>0</column-precision>
<column-scale>0</column-scale>
<table-name/>
<catalog-name/>
<column-type>4</column-type>
<column-type-name>INTEGER</column-type-name>
</column-definition>
<column-definition>
<column-index>4</column-index>
<auto-increment>>false</auto-increment>
<case-sensitive>>false</case-sensitive>
<currency>>false</currency>
<nullable>0</nullable>
<signed>>false</signed>
<searchable>>false</searchable>
<column-display-size>0</column-display-size>
<column-label>Message</column-label>
<column-name>B.MESSAGE_TEXT</column-name>
<schema-name/>
<column-precision>0</column-precision>
<column-scale>0</column-scale>
<table-name/>
<catalog-name/>
<column-type>12</column-type>
<column-type-name>VARCHAR</column-type-name>

```

```

</column-definition>
<column-definition>
  <column-index>5</column-index>
  <auto-increment>>false</auto-increment>
  <case-sensitive>>false</case-sensitive>
  <currency>>false</currency>
  <nullable>0</nullable>
  <signed>>false</signed>
  <searchable>>false</searchable>
  <column-display-size>0</column-display-size>
  <column-label>Severity</column-label>
  <column-name>B.MSG_SEVERITY</column-name>
  <schema-name/>
  <column-precision>0</column-precision>
  <column-scale>0</column-scale>
  <table-name/>
  <catalog-name/>
  <column-type>12</column-type>
  <column-type-name>VARCHAR</column-type-name>
</column-definition>
</metadata>
<data>
  <currentRow>
    <columnValue>2</columnValue>
    <columnValue>PeopleCode</columnValue>
    <columnValue>1</columnValue>
    <columnValue>Invalid use of !.</columnValue>
    <columnValue>E</columnValue>
  </currentRow>
  <currentRow>
    <columnValue>2</columnValue>
    <columnValue>PeopleCode</columnValue>
    <columnValue>2</columnValue>
    <columnValue>Literal is too long.</columnValue>
    <columnValue>E</columnValue>
  </currentRow>
</data>
</webRowSet>
</QAS_GETQUERYRESULTS_RESP_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_QUERYRESULTS_XMLP_RESP

This response is used when the requested output is *NONFILE* and the output type is *XMLP*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_GETQUERYRESULTS_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools
/schemas/QAS_GETQUERYRESULTS_RESP_MSG.VERSION_1">
      <query numRows="2" queryname="MSGCAT_PROMPT" xsi:noNamespaceSchemaLocation=
="">
        <row rownumber="1">
          <MESSAGE_SET_NBR>2</MESSAGE_SET_NBR>
          <DESCR>PeopleCode</DESCR>
          <MESSAGE_NBR>1</MESSAGE_NBR>
          <MESSAGE_TEXT>Invalid use of !.</MESSAGE_TEXT>
          <MSG_SEVERITY>E</MSG_SEVERITY>
        </row>
        <row rownumber="2">
          <MESSAGE_SET_NBR>2</MESSAGE_SET_NBR>
          <DESCR>PeopleCode</DESCR>
          <MESSAGE_NBR>2</MESSAGE_NBR>
          <MESSAGE_TEXT>Literal is too long.</MESSAGE_TEXT>
          <MSG_SEVERITY>E</MSG_SEVERITY>
        </row>
      </query>
    </QAS_GETQUERYRESULTS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

```

        </QAS_GETQUERYRESULTS_RESP_MSG>
    </soapenv:Body>
</soapenv:Envelope>

```

QAS_EXECUTEQRYSYNCPOLL_OPER

This service operation runs the query request in the Process Scheduler. This service operation should be used for long running queries or queries with large result sets. The query is executed in batch mode through Integration Broker asynchronous servers (Pub/Sub) and the Process Schedulers.

When a client request for QAS_EXECUTEQRYSYNCPOLL_OPER is received by the Integration Broker, the following occurs:

1. An asynchronous one way service operation which gets processed by Integration Broker.
2. The query instance ID (transaction ID) is returned to the requester.

The requester can use the query instance to poll for the status (QAS_QUERYSTATUS_OPER) and to retrieve the query results (QAS_GETQUERYRESULTS_OPER).

3. The service operation schedules the application engine program QASEXEQRYSYNCPOLL_OPER in the Process Scheduler.

The user ID in the request header must have permission to execute the QASEXEQRYSYNCPOLL_OPER application engine program, as well as permission for the service operation. The PeopleSoft delivered role *QAS Admin* grants the necessary permissions.

Note: Pub/Sub and Process Scheduler must be running on the application server for the QAS_EXECUTEQRYSYNCPOLL_OPER service operations to be queued and processed.

When the execute query sync poll request is processed, it will validate the query fields and query prompts. If the client application sending the request caches the query definition, the client application can assume the query definition has not changed and it is not necessary to make calls to ListQueryFields and ListQueryPrompts. If the client application receives an error, such as incorrect field name or incorrect prompt value, then the client application should assume that the query definition has been updated. If the query definition has been updated, then the client application will need to make calls to ListQueryFields and ListQueryPrompts.

Request Message: QAS_EXEQRYSYNCPOLL_REQ_MSG

<i>Element Name</i>	<i>Description</i>
QueryName Required element unless you are using XML string	Query name.
isConnectedQuery Required element	Indicate whether this is a connected query. Valid values are <i>Y</i> and <i>N</i> .
XMLString	XML string. <hr/> Note: See XML String.

Element Name	Description
OwnerType Required element	Query owner type.
BlockSizeKB Required element	<p>If the output format is nonfile, indicate the block size to use for chunking the response. BlockSizeKB is a number of length 10 with a decimal position of 0. Block size is in bytes.</p> <hr/> <p>Note: The number of calls made to GetQueryResult reduces as the block size value increases. Thus, increasing performance.</p> <hr/> <p>See Performance Considerations.</p>
MaxRows Required element	The maximum number of rows to be fetched
OutResultType Required element	<p>Select the output type. Valid values are:</p> <ul style="list-style-type: none"> • WEBROWSET • XMLP • EXCEL • HTML <hr/> <p>Note: For connected queries, the output type must be XMLP.</p> <hr/> <p>See Output Format and Output Type.</p>
OutResultFormat Required element	<p>Select the output format. Valid values are:</p> <ul style="list-style-type: none"> • FILE • NONFILE <hr/> <p>Note: For connected queries, the output format must be FILE.</p> <hr/>

<Prompts> These elements are used for Connected Query or queries containing prompts.

Prompt Elements	Description
PSQueryName	This field is used only for Connected Query. Enter the name of the query containing the prompt.

Prompt Elements	Description
UniquePromptName	Unique prompt name defined in the query. This value is case-sensitive. Use the service operation QAS_LISTQUERYPROMPTS_OPER to find the unique prompt name.
FieldValue	Field value for the prompt. This value is case-sensitive. Note: Date fields require the date format as YYYY-MM-DD.

<FieldFilter> If you want to return only specific fields in the query, indicate each field you want returned.

FilterFieldName Element	Description
FilterFieldName	List of field names to be returned. This value is case sensitive and must be the unique field name as returned by the service operation QAS_LISTQUERYFIELDS.

Example Request:

This request will execute the query XRFWIN with an output type of XMLP and output format of FILE.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_EXEQRY_SYNCPLL_REQ_MSG.VERSION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_EXEQRY_SYNCPLL_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_EXEQRY_SYNCPLL_REQ_MSG>
      <!--Zero or more repetitions:-->
      <qas1:QAS_EXEQRY_SYNCPLL_REQ>
        <QueryName>XRFWIN</QueryName>
        <isConnectedQuery>n</isConnectedQuery>
        <XMLString/>
        <OwnerType>public</OwnerType>
        <BlockSizeKB></BlockSizeKB>
        <MaxRow>3</MaxRow>
        <OutResultType>XMLP</OutResultType>
        <OutResultFormat>FILE</OutResultFormat>
      </qas1:QAS_EXEQRY_SYNCPLL_REQ>
    </qas:QAS_EXEQRY_SYNCPLL_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_EXEQRY_SYNCPLL_RESP_MSG

Element Name	Description
QueryInstance	Query instance ID. This instance ID is used for GetQueryStatus, CancelQuery, and GetQueryResult.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_EXEQRYSYNC POLL_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/=
schemas/QAS_EXEQRYSYNC POLL_RESP_MSG.VERSION_1">
      <QAS_EXEQRYSYNC POLL_RESP>
        <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch=
emas/QAS_EXEQRYSYNC POLL_RESP.VERSION_1">
          <QueryInstance>8995a01e-0a75-11dd-9c24-98a15db6aa18</QueryInstance>
        </PTQASWRK>
      </QAS_EXEQRYSYNC POLL_RESP>
    </QAS_EXEQRYSYNC POLL_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

QAS_EXECUTEQRYASNC_OPER

QAS supports running a query asynchronously. The request is received and executed. When the query has been completed, the results are asynchronously sent back to the requesting system.

Note: Pub/Sub must be running on the applications server for asynchronous service operations to be queued and processed.

For asynchronous requests, parameters are mandatory in the SOAP header in order to receive the response. This table lists the Reply To elements in the SOAP header:

These are standard parameters for any asynchronous request/response web service.

<i>Element</i>	<i>Description</i>
Address	Address where the http listener is running on the client application. The Address should be valid and should be accessible to Integration Broker in order to return the query result. If not, the results will never be returned.
ReferenceProperties	Any reference properties necessary for the client application.
PortType	Operations of the service.
Any additional elements	Any additional elements necessary for the client application.
MessageID	Unique message ID. Every request to execute query asynchronously should have a different value for MessageID.

This service operation is used to asynchronously request query execution and return the query results. The response message depends on the result type requested (OutResultType).

Request Message: QAS_EXEQRY_ASYNC_REQ_MSG

<i>Element Name</i>	<i>Description</i>
QueryName	Query name.
isConnectedQuery	Indicate whether this is a connected query. Valid values are <i>Y</i> and <i>N</i> .
XMLString	XML string. Note: See XML String for the elements.
OwnerType	Query owner type.
BlockSizeKB Required element	If the output format is nonfile, indicate the block size to use for chunking the response. BlockSizeKB is a number of length 10 with a decimal position of 0. Block size is in bytes. Note: If the BlockSize is 0 all query results are retrieved in 1 block.
MaxRows	The maximum number of rows to be fetched
OutResultType	Select the output type. Valid values are: <ul style="list-style-type: none"> • WEBROWSET • XMLP • EXCEL • HTML Note: For connected queries, the output type must be XMLP.
OutResultFormat	Select the output format. Valid values are: <ul style="list-style-type: none"> • FILE • NONFILE Note: For connected queries, the output format must be FILE.

<Prompts> These elements are used for Connected Query or queries containing prompts. .

<i>Prompt Elements</i>	<i>Description</i>
PSQueryName	This field is used only for Connected Query. Enter the name of the query containing the prompt.

Prompt Elements	Description
UniquePromptName	Unique prompt name defined in the query. This value is case-sensitive. Use the service operation QAS_LISTQUERYPROMPTS_OPER to find the unique prompt name.
FieldValue	Field value for the prompt. This value is case-sensitive. Note: Date fields require the date format as YYYY-MM-DD.

<FieldFilter> If you want to return only specific fields in the query, indicate each field you want returned.

FilterFieldName Element	Description
FilterFieldName	List of field names to be returned. This value is case sensitive and must be the unique field name as returned by the service operation QAS_LISTQUERYFIELDS.

Example Request:

This is an example request to run the query MSGCAT_PROMPT and notify the client application when the request has finished.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:addr="http://schemas.xmlsoap.org/ws/2003/03/addressing" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_EXEQRY_ASYNC_REQ_MSG.VERSION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_EXEQRY_ASYNC_REQ.VERSION_1">
  <soapenv:Header>
    <add:ReplyTo>
      <add:Address>http://ple-indodev-08:8082/qasrefapp/callbackservice</add:Address>
    </add:ReplyTo>
  </soapenv:Header>
  <soapenv:Body>
    <qas:QAS_EXEQRY_ASYNC_REQ_MSG>
      <qas1:QAS_EXEQRY_ASYNC_REQ>
        <QueryName>xrfwin</QueryName>
        <isConnectedQuery>n</isConnectedQuery>
        <OwnerType>public</OwnerType>
        <BlockSizeKB>0</BlockSizeKB>
        <MaxRow></MaxRow>
        <OutResultType>xmlp</OutResultType>
        <OutResultFormat>NONFILE</OutResultFormat>
      </qas1:QAS_EXEQRY_ASYNC_REQ>
    </qas:QAS_EXEQRY_ASYNC_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

For Webrowset , the Response Message is QAS_EXECUTEQRYSYNC_RESP_MSG: QAS_QUERYRESULTS_WSR_RESP

<i>Element Name</i>	<i>Description</i>
QueryResults	Query Result in webrowset format

Example Response:

For File, the Response Message is QAS_QUERYRESULTS_FILE_RESP.

<i>Element Name</i>	<i>Description</i>
FILEURL	File URL

Retrieving Query Results

The service operation QAS_GETQUERYRESULTS_OPER is used only to retrieve the query results for the QAS_EXECUTEQRYSYNCPOLL_OPER service call.

Query results are stored either in message segments or in an XML file. Segments can contain rows of data (nonrowset-based messages). When the query result is stored in the XML file, QAS will return the URL of the file location.

QAS_GETQUERYRESULTS_OPER

This service operation is used to retrieve the query results.

Request Message: QAS_GETQUERYRESULTS_REQ_MSG

<i>Element Name</i>	<i>Description</i>
BlockNumber	<p>The number of blocks to be retrieved.</p> <p>Query result is retrieved starting from block 1 in sequential order until the final block is retrieved.</p> <hr/> <p>Note: The block that contains status <i>finalBlockRetrieved</i> is the final block.</p> <hr/>
QueryInstance	Query instance ID.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_GETQUERYRESULTS_REQ_MSG.VERSION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_GETQUERYRESULTS_REQ.VERSION_1">
  <soapenv:Header/>
```

```

<soapenv:Body>
  <gas:QAS_GETQUERYRESULTS_REQ_MSG>
    <!--Zero or more repetitions:-->
    <gas:QAS_GETQUERYRESULTS_REQ>
      <gas1:PTQASWRK class="R">
        <gas1:BlockNumber>1</gas1:BlockNumber>
        <gas1:QueryInstance>8995a01e-0a75-11dd-9c24-98a15db6aa18</gas1:Query=
Instance>
      </gas1:PTQASWRK>
    </gas:QAS_GETQUERYRESULTS_REQ>
  </gas:QAS_GETQUERYRESULTS_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_GETQUERYRESULTS_RESP_MSG

Depending on the output format requested, the response will be either a file URL or the query result in webrowset or XMLP format.

Element Name	Description
FILEURL	Returns FILE URL.
status	<p>If the output format is file, the return status will be either <i>success</i> or <i>failure</i>.</p> <p>If the output format is non file the return status will be one of the following:</p> <ul style="list-style-type: none"> <i>running</i> or <i>queued</i> indicates that clients have to continue polling for the same block until a <i>statusBlockRetrieved</i> or <i>finalBlockRetrieved</i> is returned. <i>blockRetrieved</i> indicates that the clients have received 1 block of query result and will have to poll for the next block. <i>finalBlockRetrieved</i> indicates that there are no more blocks to retrieve and clients can stop polling for results.

Example Response:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_GETQUERYRESULTS_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools
/schemas/QAS_GETQUERYRESULTS_RESP_MSG.VERSION_1">
      <QAS_QUERYRESULTS_FILE_RESP xmlns="http://xmlns.oracle.com/Enterprise/Tool
s/schemas/QAS_QUERYRESULTS_FILE_RESP.VERSION_1">
        <FILEURL>http://ple-infodev-09.peoplesoft.com:8010/psreports/QEDMO/9999
911/XRFWIN.xml</FILEURL>
        <status>success</status>
      </QAS_QUERYRESULTS_FILE_RESP>
    </QAS_GETQUERYRESULTS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Canceling a Query

The service operation `QAS_CANCELQUERY_OPER` is used to cancel a query after execution. You need to cancel a query using the query instance ID to clean up the `PSQASRUN` table. The QAS Administration page can also be used to clear any orphaned rows in the `PSQASRUN` table.

See [Using QAS Administration](#).

QAS_CANCELQUERY_OPER

This service operation is used to cancel an executed query.

Request Message: QAS_CANCELQUERY_REQ_MSG

<i>Element Name</i>	<i>Description</i>
QueryInstanceID	Query Instance ID

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_CANCELQUERY_REQ_MSG.VERSION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_CANCELQUERY_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_CANCELQUERY_REQ_MSG>
      <qas:QAS_CANCELQUERY_REQ>
        <qas1:PTQASWRK class="R">
          <qas1:QueryInstanceID>ab621810-5437-11dd-83cd-a32b27c3af6a</qas1:QueryInstanceID>
        </qas1:PTQASWRK>
      </qas:QAS_CANCELQUERY_REQ>
    </qas:QAS_CANCELQUERY_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_CANCELQUERY_RESP_MSG

<i>Element Name</i>	<i>Description</i>
Status	Return status. Valid values are: <ul style="list-style-type: none"> • idNotFound Used when the queryInstanceID isn't correct. • fail • success

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema-1">
```

```

1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_CANCELQUERY_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch
emas/QAS_CANCELQUERY_RESP_MSG.VERSION_1">
      <QAS_CANCELQUERY_RESP>
        <PTQASSTATWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools
/schemas/QAS_CANCELQUERY_RESP.VERSION_1">
          <Status>success</Status>
        </PTQASSTATWRK>
      </QAS_CANCELQUERY_RESP>
    </QAS_CANCELQUERY_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Retrieving Query Status

The service operation QAS_QUERYSTATUS_OPER is used to return the status of query execution. This service operation returns the QAS status, not the Process Scheduler status.

See "Viewing the Status of Processes" (PeopleTools 8.53: PeopleSoft Process Scheduler).

QAS_QUERYSTATUS_OPER

This service operation returns current query status.

Request Message: QAS_QUERYSTATUS_REQ_MSG

<i>Element Name</i>	<i>Description</i>
QueryInstanceID	Query instance ID.

Example Request:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERYSTATUS_REQ_MSG.VERSIO
N_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_QUERYSTATUS_R
EQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_QUERYSTATUS_REQ_MSG>
      <qas:QAS_QUERYSTATUS_REQ>
        <qas1:PTQASWRK class="R">
          <qas1:QueryInstanceID>64ef3e36-5cf6-11dd-a1b6-cf9c0ed84831</qas1:Que
ryInstanceID>
        </qas1:PTQASWRK>
      </qas:QAS_QUERYSTATUS_REQ>
    </qas:QAS_QUERYSTATUS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_QUERYSTATUS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
Status	Returns query status. Values are: <ul style="list-style-type: none"> • idNotFound • running • killed • posting • last • queued • success • error
NumCols	Number of columns.
TotalRows	Total number of rows.
TotalBlocks	Always returns 0.
TotalBytes	Total number of bytes.

Example Response:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_QUERYSTATUS_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch
emas/QAS_QUERYSTATUS_RESP_MSG.VERSION_1">
      <QAS_QUERYSTATUS_RESP>
        <PTQASSTATWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools
/schemas/QAS_QUERYSTATUS_RESP.VERSION_1">
          <Status>running</Status>
          <NumColumns>0</NumColumns>
          <TotalBlocks>0</TotalBlocks>
          <TotalBytes>0</TotalBytes>
          <TotalRows>0</TotalRows>
        </PTQASSTATWRK>
      </QAS_QUERYSTATUS_RESP>
    </QAS_QUERYSTATUS_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```


Chapter 5

Accessing PeopleSoft Application Tables

Understanding QAS Security

This section discusses:

- Query security
- Service Operation security
- WS-Security
- Process Profile

This section also discusses QAS security flow.

Query Security

PeopleSoft Query uses query access group trees to control security of the tables in your PeopleSoft database. You define a hierarchy of record components, based on logical or functional groupings, and then give users access to one or more branches of the tree. Users can use PeopleSoft Query to retrieve information only from record definitions they have access to based on the query access tree assignment.

Related Links

"Using Query Access Group Trees" (PeopleTools 8.53: PeopleSoft Query)

Service Operation Security

QAS service operations are delivered with User/Password Required enabled and WS Security Req Verification set to *Encrypt and Digitally Sign or HTTPS*.

Client applications using QAS service operations must either digitally encrypt and sign the request or send the request over HTTPS.

Service operations are secured by means of permission lists. PeopleSoft applications deliver the permission list *PTPT2200* (QAS access), which has full access to all QAS service operations and the application engine program QASEXEQRY. The role *QAS Admin* contains the permission list PTPT2200. Any users assigned the role QAS Admin can access the QAS service operations.

WS-Security

Web services security (WS-Security) is implemented on the integration gateway for inbound and outbound integrations with third-party systems. QAS service operations use WS-Security.

See [Web Services Overview](#).

Process Profile

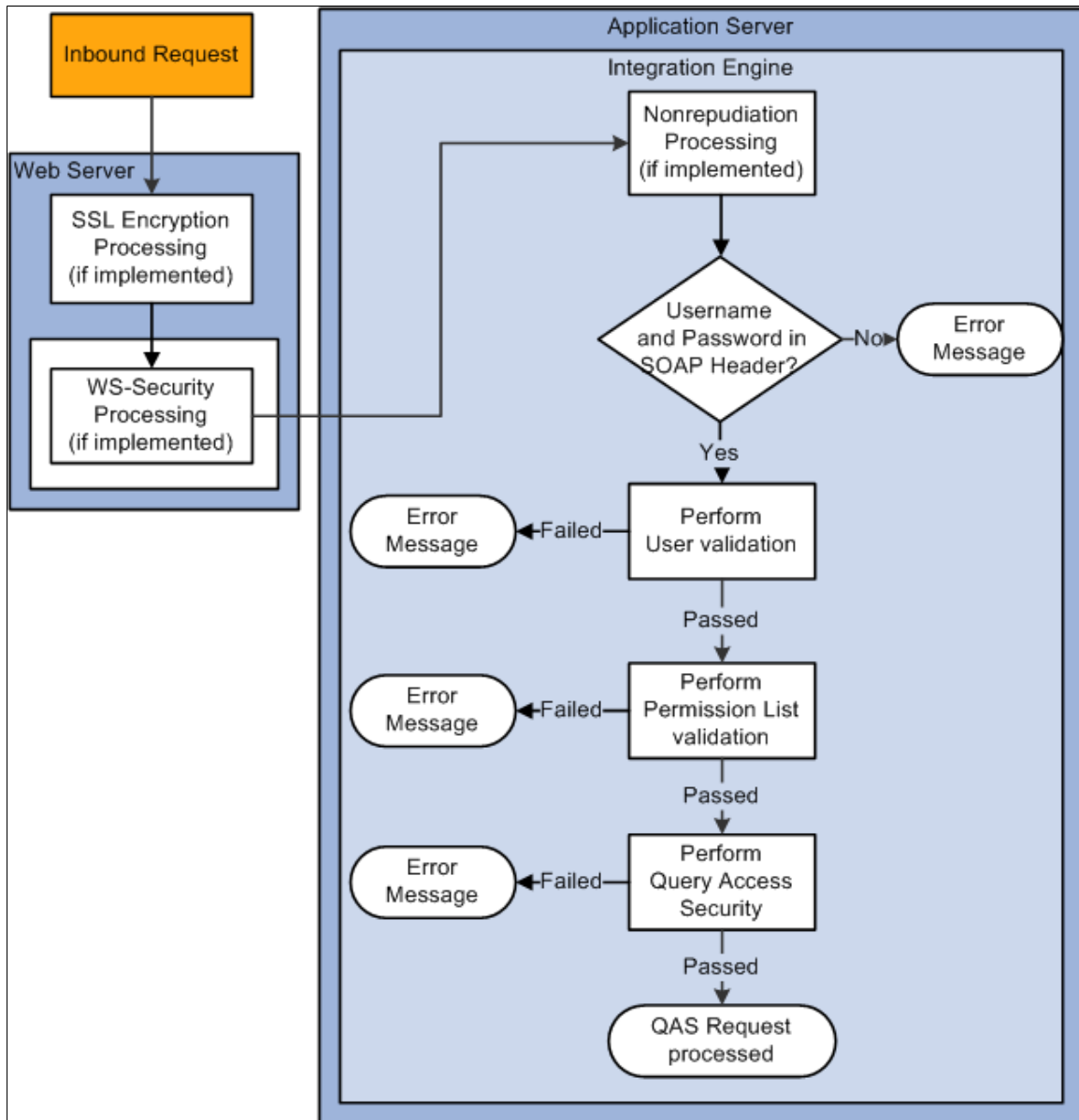
The service operation QAS_EXECUTEQRYSYNCPOLL_OPER schedules the application engine program QASEXEQRYSYNCPOLL to run in Process Scheduler, therefore the user initiating the request must have permission to run QASEXEQRYSYNCPOLL in the Process Profile.

See QASEXEQRYSYNCPOLL_OPER.

QAS Security Flow

Image: QAS request from a third-party security flow

This diagram illustrates the QAS request inbound flow from a third-party system in the Integration Broker



When any transaction arrives at the integration gateway, the PeopleSoft system checks for the existence of a WS-Security SOAP header. If it exists, the integration gateway validates the digital signature if it exists, and decrypts the UsernameToken and optional password to restore the user ID information to clear text

format. The integration gateway then passes the user ID information, and UsernameToken password if provided by the sender, to the application server, where additional security processing is performed.

If a user name and password are supplied in the SOAP header, the user is validated in the PeopleSoft system.

If no user ID and password are supplied, the request is rejected.

The PeopleSoft system then validates whether the user's permission list grants access to the QAS service operation.

If the user is authorized to the service operation, then Query Access security is used and the request is processed.

QAS Security Service Operations

Query access security is defined on permission lists. Roles contain one or more permission lists and the user is assigned roles. Several service operations are available that a third party can use to list roles and role users.

QAS_AUTHTOKEN_OPER

This service operation is used to retrieve the user ID for a PStoken. This service operation is used when a Business Object Enterprise (BOE) report is run through the Process Scheduler. The PStoken is sent in the HTTP header over HTTPS. BOE will use this service operation to determine the user ID requesting the report.

Request Message: QAS_AUTHTOKEN_REQ_MSG

<i>Element Name</i>	<i>Description</i>
PSTOKEN	PeopleSoft authorization token.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:qas="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_AUTHTOKEN_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_AUTHTOKEN_REQ_MSG>
      <PSTOKEN>owAAAAQDAgEBAAAavAIAAAAAAAsAAAABABTaGRyAk4AbQg4AC4AMQAwABRkm1RLE0zCq6JFYA=>
oVWo7oK06qVGMAAAAFaFNkYXRhV3icy2VgYGBhZmJkBNJ7mBjAgCuQwZXBhcGXwZ+BzZXBj8GdQQakEs/gA=>
xRxZnAE0iZGDAZAaMmgCySNgKQRgxmYbcqgByUNwaQLUMYQrNaAgQEAb08LPQ==; http%3a%2f%2fple-i=>
nfodev-08.peoplesoft.com%3a8010%2f%2f%2frefresh=list:=>
; HPTabName=DEFAULT</PSTOKEN>
    </qas:QAS_AUTHTOKEN_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_AUTHTOKEN_RESP_MSG

<i>Element</i>	<i>Description</i>
LoginUser	Returns the user ID for the PSToken.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qss:QAS_AUTHTOKEN_RESP_MSG xmlns:qss="http://xmlns.oracle.com/Enterprise/Too
ls/schemas/QAS_AUTHTOKEN_RESP_MSG.VERSION_1">
      <LoginUser>QEDMO</LoginUser>
    </qss:QAS_AUTHTOKEN_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

QAS_LISTROLE_OPER

Use this service operation to get a list of roles, along with descriptions.

Request Message: QAS_LISTROLE_REQ_MSG

<i>Element name</i>	<i>Description</i>
SearchString	Search string used for specifying the role name or the first few characters of the role name. If no value is entered, all roles will be returned. This value is case-sensitive.

Example Request:

This is an example of a request to select all roles that begin with *QAS*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q=
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTROLE_REQ_MSG.VERSION_1"
xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTROLE_REQ.VER
SION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_LISTROLE_REQ_MSG>
      <!--Zero or more repetitions:-->
      <qas:QAS_LISTROLE_REQ>
        <qas1:PTQASWRK class="R">
          <!--Optional:-->
          <qas1:SearchString>QAS</qas1:SearchString>
        </qas1:PTQASWRK>
      </qas:QAS_LISTROLE_REQ>
    </qas:QAS_LISTROLE_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_LISTROLE_RESP_MSG

<i>Element Name</i>	<i>Description</i>
RoleName	Role name.
Description	Role description.

Example Response:

```
<?xml version="1.0"?>
<QAS_LISTROLE_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_
LISTROLE_RESP_MSG.VERSION_1">
  <QAS_LISTROLE_RESP>
    <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas
/QAS_LISTROLE_RESP.VERSION_1">
      <RoleName>QAS Admin</RoleName>
      <Description>QAS Administrators</Description>
    </PTQASWRK>
  </QAS_LISTROLE_RESP>
</QAS_LISTROLE_RESP_MSG>
```

QAS_LISTUSERROLES_OPER

Use this service operation to get a list of roles for a given user, along with descriptions.

Request Message: QAS_LISTUSERROLES_REQ_MSG

<i>Element Name</i>	<i>Description</i>
UserName	Complete user name. Required and case-sensitive.
Required element	
SearchString	Search string used for specifying the role name or the first few characters of the role name. If no value is entered, all roles for the user will be returned. This value is case-sensitive.

Example Request:

This is an example of a request to select all roles for the user *PSADMIN*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTUSERROLES_REQ_MSG.VERS
ION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTUSERROL
ES_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_LISTUSERROLES_REQ_MSG>
      <!--Zero or more repetitions:-->
      <qas:QAS_LISTUSERROLES_REQ>
        <qas1:PTQASWRK class="R">
          <qas1:UserName>PSADMIN</qas1:UserName>
          <!--Optional:-->
          <qas1:SearchString></qas1:SearchString>
        </qas1:PTQASWRK>
      </qas:QAS_LISTUSERROLES_REQ>
    </qas:QAS_LISTUSERROLES_REQ_MSG>
```

```
</soapenv:Body>
</soapenv:Envelope>
```

Response Message: QAS_LISTUSERROLES_RESP_MSG

Element Name	Description
RoleName	Role name.
Description	Role description.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <QAS_LISTUSERROLES_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/s
chemas/QAS_LISTUSERROLES_RESP_MSG.VERSION_1">
      <QAS_LISTUSERROLES_RESP>
        <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch
emas/QAS_LISTUSERROLES_RESP.VERSION_1">
          <RoleName>PeopleSoft Administrator</RoleName>
          <Description>PeopleSoft Admin Privileges</Description>
        </PTQASWRK>
      </QAS_LISTUSERROLES_RESP>
    </QAS_LISTUSERROLES_RESP>
    <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/sch
emas/QAS_LISTUSERROLES_RESP.VERSION_1">
      <RoleName>PeopleSoft User</RoleName>
      <Description>PeopleSoft User</Description>
    </PTQASWRK>
  </QAS_LISTUSERROLES_RESP>
</QAS_LISTUSERROLES_RESP_MSG>
</soapenv:Body>
</soapenv:Envelope>
```

QAS_LISTUSER_OPER

Use this service operation to get a list of users, along with descriptions.

Request Message: QAS_LISTUSER_REQ_MSG

Element Name	Description
SearchString	Search string used for specifying the user name or the first few characters of the user name. If no value is entered, all users will be returned. This value is case-sensitive.

Example Request:

This is an example of a request to select all users that begin with *PS*.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTUSER_REQ_MSG.VERSION_1"
" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTUSER_REQ.VER
SION_1">
  <soapenv:Header/>
  <soapenv:Body>
```

```

<qas:QAS_LISTUSER_REQ_MSG>
  <qas:QAS_LISTUSER_REQ>
    <qas1:PTQASWRK class="R">
      <!--Optional:-->
      <qas1:SearchString>PS</qas1:SearchString>
    </qas1:PTQASWRK>
  </qas:QAS_LISTUSER_REQ>
</qas:QAS_LISTUSER_REQ_MSG>
</soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_LISTUSER_RESP_MSG

Element Name	Description
UserName	User name.
Description	User description.

Example Response:

```

<?xml version="1.0"?>
<QAS_LISTUSER_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_→
LISTUSER_RESP_MSG.VERSION_1">
  <QAS_LISTUSER_RESP>
    <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas→
/QAS_LISTUSER_RESP.VERSION_1">
      <UserName>PSADMIN</UserName>
      <Description>PeopleSoft Administrator</Description>
    </PTQASWRK>
  </QAS_LISTUSER_RESP>
</QAS_LISTUSER_RESP_MSG>

```

QAS_LISTROLEUSERS_OPER

Use this service operation to get a list of users for a given role, along with descriptions.

Request Message: QAS_LISTROLEUSERS_REQ_MSG

Element Name	Description
RoleName	Complete role name. Required and case-sensitive.
Required element	
SearchString	Optional search string used for specifying the user name or the first few characters of the user name. If no value is entered, all users for the role will be returned. This value is case-sensitive.

Example Request:

This is an example of a request to select all users that begin with *PS* and have the role *PeopleSoft Administration*.

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q→
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTROLEUSERS_REQ_MSG.VERS→
ION_1" xmlns:qas1="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LISTROLEUSE→

```

```

RS_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <qas:QAS_LISTROLEUSERS_REQ_MSG>
      <!--Zero or more repetitions:-->
      <qas:QAS_LISTROLEUSERS_REQ>
        <qas1:PTQASWRK class="R">
          <qas1:RoleName>PeopleSoft Administrator</qas1:RoleName>
          <!--Optional:-->
          <qas1:SearchString>PS</qas1:SearchString>
        </qas1:PTQASWRK>
      </qas:QAS_LISTROLEUSERS_REQ>
    </qas:QAS_LISTROLEUSERS_REQ_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: QAS_LISTROLEUSERS_RESP_MSG

<i>Element Name</i>	<i>Description</i>
UserName	User name.
Description	User description.

Example Response:

```

<?xml version="1.0"?>
<QAS_LISTROLE_RESP_MSG xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_→
LISTROLE_RESP_MSG.VERSION_1">
  <QAS_LISTROLE_RESP>
    <PTQASWRK class="R" xmlns="http://xmlns.oracle.com/Enterprise/Tools/schemas→
/QAS_LISTROLE_RESP.VERSION_1">
      <RoleName>QAS Admin</RoleName>
      <Description>QAS Administrators</Description>
    </PTQASWRK>
  </QAS_LISTROLE_RESP>
</QAS_LISTROLE_RESP_MSG>

```

QAS_LOGIN_OPER

This service operation is available for a client application to sign on to the PeopleSoft database and use QAS service operations to create and execute queries.

To use this service operation, the user must install and configure certificates.

See "Understanding SSL/TLS and Digital Certificates" (PeopleTools 8.53: Security Administration).

Request Message: QAS_LOGIN_REQ_MSG

<i>Element Name</i>	<i>Description</i>
UserVerificationAttempt	Do not enter a value.

Example Request:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q→
as="http://xmlns.oracle.com/Enterprise/Tools/schemas/QAS_LOGIN_REQ_MSG.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>

```



```

    <qas:UserVerificationAttempt></qas:UserVerificationAttempt>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message:

Element Name	Description
IsValidUser	Returns <i>Y</i> if the user is validated.

Example Response:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <qss:QAS_LOGIN_RESP_MSG xmlns:qss="http://xmlns.oracle.com/Enterprise/Tools/s=
chemas/QAS_LOGIN_RESP_MSG.VERSION_1">
      <IsValidUser>Y</IsValidUser>
    </qss:QAS_LOGIN_RESP_MSG>
  </soapenv:Body>
</soapenv:Envelope>

```

Using QAS Administration

The QAS Administration page is used to monitor QAS query execution. To access the QAS administration page, the user must have permission to access the QAS Administration page (PSQASADMIN).

After executing a query, the client application is responsible for canceling the query, which will delete the row from the PSQASRUN table. If the rows are not deleted by the client application, the QAS Administrator can delete the rows using the QAS Administration page.

To access the QAS Administration page, select PeopleTools, Utilities, Administration, QAS Administration (PSQASADMIN).

Image: QAS Administration page

This example illustrates the fields and controls on the QAS Administration page.

QAS Administration								
Query Execution Status								
Status	Customize Find First 1-5 of 5 Last							
User ID	Instance ID	Query Name	Owner Type	Run Status	Total Number of Blocks	Total Number of bytes	Date Time of Query Creation	Clear
QEMGR	8da33ade-4fb3-11de-927d-eefd6364b513	XRFWIN	Public	success	1	19420	06/02/2009 1:25:46.000000PM	Clear
QEDMO	eaf87c92-4fc4-11de-927d-eefd6364b513	MESSAGES_FOR_MSGSET_1_20	Public	success	1	7878	06/02/2009 3:30:04.000000PM	Clear
QEDMO	a435f126-4fc5-11de-927d-eefd6364b513	MESSAGES_FOR_MSGSET	Public	success	1	16560	06/02/2009 3:35:15.000000PM	Clear
VP1	b4824465-4fc5-11de-927d-eefd6364b513	MESSAGES_FOR_MSGSET	Public	posting	0	0	06/02/2009 3:35:42.000000PM	Clear
QEMGR	7de39260-4fb3-11de-927d-eefd6364b513	XRFWIN	Public	error	0	0	06/02/2009 1:25:19.000000PM	Clear

This page displays the run status for QAS service operations that execute queries on the PeopleSoft system. Depending on the execution type and output format, you will see various run statuses.

This table lists the run statuses by output format.

<i>Output Format</i>	<i>Status</i>	<i>Description</i>
FILE	running	The report is running in Process Scheduler.
FILE	posting	The report was posted to the report repository.
FILE or NONFILE	error	The query encountered an error. If the query does not exist or the user does not have access to the query, an error will occur.
NONFILE	success	The query data is stored in the Integration Broker runtime tables.

Use the Clear button to delete entries from the page.

Understanding Process Scheduler Service

Process Scheduler Service Operations

To enable external applications to access PeopleSoft Process Scheduler, a number of service operations are available. Depending on the application, the external application will need to use several service operations in combination. The service operations for Process Scheduler are part of the service PROCESSREQUEST.

Process Scheduler service operations are grouped into the following categories to help describe the service operations:

- Schedule requests
- Monitor requests

Schedule Requests

Service operations to schedule requests are used by client applications to schedule process requests to run in the PeopleSoft application. This table lists the service operations:

Service Operation Name	Description
PRCS_GETPROCESSNAMES	Returns a list of process names based on process type.
PRCS_GETPARAMS	Returns the runtime parameters registered for a process in the process definition.
PRCS_GETPROMPT	Returns a list of process types, distribution status, run status, server names, operator IDs or recurrence definitions, time zones, folder names depending on the request.
PRCS_SCHEDULE	Schedules a job or process using all the parameters required to schedule the job or process. Returns the process instance.
PRCS_SCHEDULE_PSQUERY	Schedules a non-unique process query.
PRCS_SCHEDULE_CONQUERY	Schedules a non- unique process connected query.
PRCS_SCHEDULE_BIPUBLISHER	Schedules a non-unique process BIPublisher.
PRCS_SCHEDULE_NVISION	Schedules a non-unique process nVision report request.
PRCS_RUNJOBSETNOW	Schedules a JobSet based on settings defined in the Scheduled JobSet Definition component. kalaiyarasu

Monitor Processes

Service operations to monitor requests and retrieve reports are used by client applications to monitor process requests in the PeopleSoft application. This table lists the service operations:

Service Operation Name	Description
PRCS_GETREQUEST	Returns the process details for a given process instance. The details include the run status, distribution status, output information, distribution information, runtime information, and recurrence information.
PRCS_GETREPORT	Returns report and log file details for the reports posted to the report repository.
PRCS_FINDREQUESTS	Used to search for submitted process requests based on a combination of different filter criteria.
PRCS_UPDATEREQUEST	Used to update the process request.
PRCS_GETSERVERCONSTRAINTS	Retrieves server specific details, such as count of Max API aware, Max concurrent for each process type, Max concurrent for each process category.
PRCS_GETOUTPUTPARAMS	Retrieves the output type and output format for a given process or job name.
PRCS_PRINT_JOB_HTML_REPORT	Retrieves an HTML formatted report displaying all items in a job or jobset as defined in the Job Definition component.
PRCS_PRINT_JOB_REQUEST_REPORT	Retrieves an HTML formatted report displaying the current status of a specific process, job, and jobset.
PRCS_PRINT_SCHEDULE_HTML_REPORT	Retrieves an HTML formatted report displaying all items in a job or jobset as defined in the Scheduled Jobset Definition component.

Process Scheduler Service

This section provides an overview of PROCESSREQUEST service and discusses:

- WSDL Document
- Message Schema
- SOAP Template

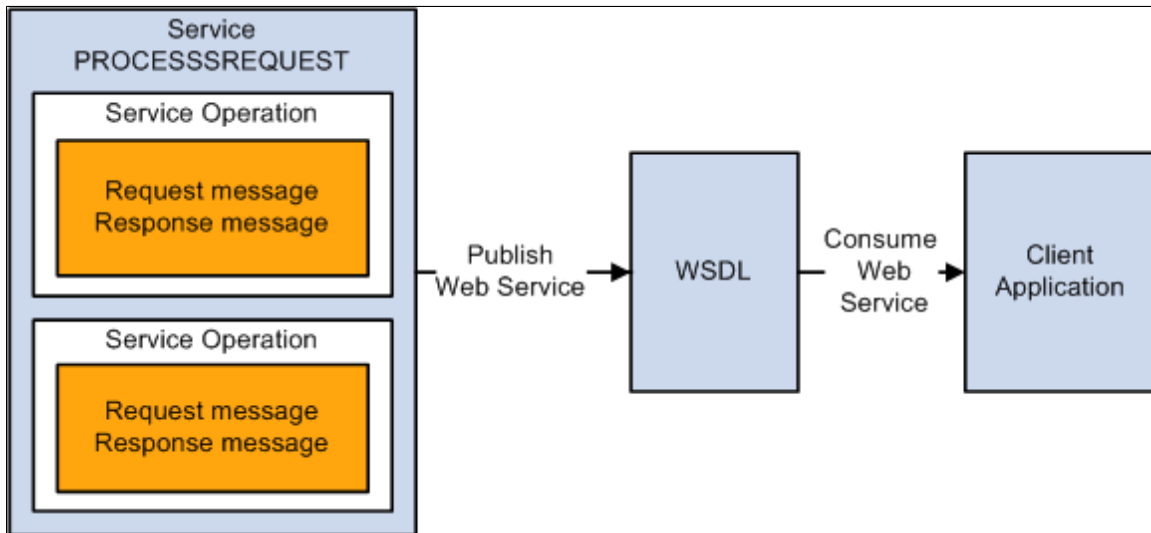
Understanding PROCESSREQUEST Service

Services are used to logically group a set of service operations. The service PROCESSREQUEST contains all of the service operations for Process Scheduler. When you publish a web service, a WSDL document is created containing all of the selected service operations as well as the endpoints based on the Service Configuration.

See "Using the Service Configuration Page to Set Service Configuration Properties" (PeopleTools 8.53: PeopleSoft Integration Broker Administration).

Image: Providing PROCESSREQUEST web service

This diagram illustrates how a client application consumes the web service



The Service page provides links to Provide Web Service and to Generate SOAP Template. The Provide Web Service wizard generates a WSDL document containing all of the service operations selected. This WSDL document is consumed by the client application.

The published web service is stored in the WSDL Repository. Client applications consume the WSDL by accessing the WSDL URL from the repository. The WSDL for PROCESSREQUEST is delivered and updated with the appropriate WSDL Repository location whenever the target location is updated on the Service Configuration page (Select PeopleTools, Integration Broker, Configuration, Service Configuration).

WSDL Document

To view the generated WSDL Repository for the PROCESSREQUEST, click the View WSDL link on the service page.

Image: WSDL Repository page for PROCESSREQUEST service

This example illustrates the fields and controls on the WSDL Repository page for PROCESSREQUEST service.

The screenshot shows the WSDL Repository interface for the PROCESSREQUEST service. It includes a 'Service: PROCESSREQUEST' label, a 'WSDL' section with a 'View WSDL' link, and an 'Exported Service Operations' table. The table lists five operations with their respective routing, request, response, and fault messages.

Operation	Routing External Alias	Request Message	Response Message	Fault Message
PRCS_FINDREQUESTS	PRQ_FINDREQUESTS.v1	PRQ_FINDREQUESTS_REQ.1	PRQ_FINDREQUESTS_RES.1	
PRCS_GETPARAMS	PRQ_GETPARAMS.v1	PRQ_GETPARAMS_REQ.1	PRQ_GETPARAMS_RES.1	
PRCS_GETPROCESSNAMES	PRCS_GETPROCESSNAMES.v1	PRQ_GETPROCESSNAMES_REQ.1	PRQ_GETPROCESSNAMES_RES.1	
PRCS_GETPROMPT	PRQ_GETPROMPT.v1	PRQ_GETPROMPT_REQ.1	PRQ_GETPROMPT_RES.1	
PRCS_GETREPORT	PRQ_GETREPORT.v1	PRQ_GETREPORT_REQ.1	PRQ_GETREPORT_RES.1	

All of the service operations contained in the WSDL are listed in the Exported Service Operations section. Use the View WSDL link to view the WSDL document.

Message Schema

The WSDL document is created by means of the request and response message schemas for each service operation. To view the message schema, select PeopleTools, Integration Broker, Integration Setup, Messages and select the message to view. Select the Schema tab to view the schema.

SOAP Template

For each service operation, you can use the Generate SOAP Template utility to generate SOAP message templates for request messages, response messages, and fault messages found in the WSDL document. The generated SOAP template will include WS-Security SOAP Header tags. You can then use the templates to test SOAP messages in your third-party application.

Process Scheduler Service Operation Security

This section provides an overview of Process Scheduler web service security.

Process Scheduler Web Service Security

The service operations for PROCESSREQUEST service follow the same user-based security as an online user. All Process Scheduler service operations require a user ID and password that must be digitally encrypted. Based on the user ID, the request is validated based on the request type.

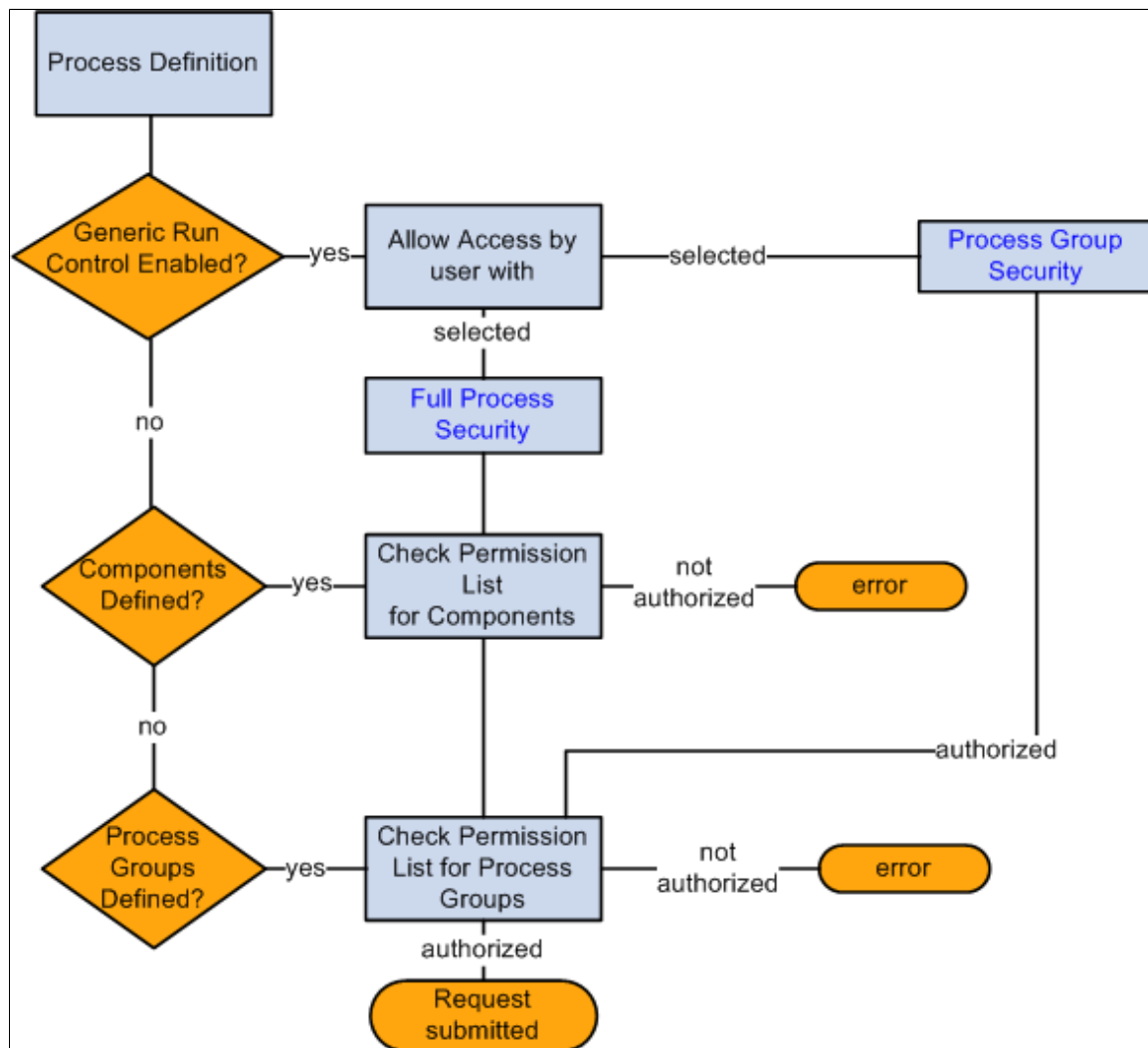
Process Request

Security is defined for a process in the process definition. Security can be controlled by access to the component containing the run control and by process security group. The user who is requesting this web service operation can schedule the processes only if the user is authorized to submit the request.

This diagram shows the security process flow to schedule a process request:

Image: Schedule process request security flow

This example illustrates the fields and controls on the Schedule process request security flow.



In the preceding diagram, the user submits the PRCS_SCHEDULE service operation, which includes a SOAP header identifying the user. The process scheduler validates the user by checking the security defined in the process definition:

1. If the process definition has Enable Generic Prompting on the Run Control Parameters page, the value for Allow Access by user with is used to validate permissions. If the generic run control is not enabled, the process scheduler will use the process security defined on the Process Definition Options page.
 - *Full Process Security*: The user must have permission for both component and process group.
 - *Process Group Security*: The user must have access to the process security group.

2. On the Process Definition Options page, the process may contain both components and process groups. If components are included, the user must have security access to at least one of the components.
3. The user must have security access to the process group.

See "Setting Process Definition Options" (PeopleTools 8.53: PeopleSoft Process Scheduler).

Monitor Request

User's access to process request via process monitor is determined by the process profile permission assigned to the person who submitted the request. In the process profile, options are available to restrict other users from viewing and updating process requests run by the owner (current user).

The options available are:

- *Owner*: Only the user who submits the process can see or update the process request from process monitor.
- *All*: All users can see or update the process request from the process monitor.
- *None*: No user, including the owner, can view or update the process request from process monitor.

Note: User IDs that contain the role *ProcessSchedulerAdmin* can monitor all processes.

See "Setting Up PeopleSoft Process Scheduler Privileges and Profiles" (PeopleTools 8.53: PeopleSoft Process Scheduler).

Retrieve Reports

User's access to retrieve reports is determined by the distribution in the report request. When a report request is submitted, you can specify the report distribution by users or roles. If the request does not contain distribution informations, the distribution defined in the Process definition is used.

Note: User IDs that contain the role *ProcessSchedulerAdmin* or *ReportDistAdmin* can retrieve all reports.

Chapter 7

Using Process Scheduler Service Operations

Understanding Process Scheduler Service Operations

A third-party application can schedule and monitor PeopleSoft processes or jobs using the process scheduler service operations. If the process requires parameters, the parameters need to be defined on the Run Control page of the Process or Job definition.

See "Setting Process Definition Options" (PeopleTools 8.53: PeopleSoft Process Scheduler).

Guidelines for Completing Service Operation Requests

The following guidelines apply across all process scheduler requests:

- All requests should contain a SOAP header containing user ID and password.
- Parameters are case sensitive.
- Leading and trailing blanks are not allowed in XML values.
- All requests should conform to the message schema defined for the service operation messages.

Scheduling a Request

Client applications can use service operations to find a process for specified process type, retrieve the run time parameters, prompt values for a specified parameters, and schedule the process request. This section describes the service operations available to help in scheduling the request. These include:

- PRCS_GETPROMPT
- PRCS_GETPROCESSNAMES
- PRCS_GETPARAMS
- PRCS_SCHEDULE
- PRCS_SCHEDULE_PSQUERY
- PRCS_SCHEDULE_CONQUERY
- PRCS_SCHEDULE_BIPUBLISHER
- PRCS_RUNJOBSETNOW

PRCS_GETPROMPT

Use this service operation to retrieve the prompts for a process.

Request Message: PRQ_GETPROMPT_REQ

<i>Element Name</i>	<i>Description</i>
ParameterName	Valid values are: <ul style="list-style-type: none"> • RunStatus • DistributionStatus • ServerName • ProcessTypes • OperatorId • RecurrenceName • Time zones All values are case-sensitive.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETPROMPT_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:GetPromptValues>
      <ParameterName>RunStatus</ParameterName>
    </prq:GetPromptValues>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_GETPROMPT_RES

<i>Element Name</i>	<i>Description</i>
Prompt Value	Returns a list of values for the request prompt.

Example Response:

This response returns the values for Run Status.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:GetPromptValuesResponse xsi:schemaLocation="http://xmlns.oracle.com/Ente
rprise/Tools/schemas/PRQ_GETPROMPT_RES.VERSION_1 PRQ_GETPROMPT_RES.VERSION_1.xsd" x=
mlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETPROMPT_RES.VERSIO
N_1">
      <PromptValue>Cancel</PromptValue>
      <PromptValue>No Success</PromptValue>
    </prq:GetPromptValuesResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    <PromptValue>Posted</PromptValue>
    <PromptValue>Not Posted</PromptValue>
    <PromptValue>Resend</PromptValue>
    <PromptValue>Posting</PromptValue>
    <PromptValue>Generated</PromptValue>
    <PromptValue>Pending</PromptValue>
    <PromptValue>Warning</PromptValue>
    <PromptValue>Blocked</PromptValue>
    <PromptValue>Restart</PromptValue>
    <PromptValue>Delete</PromptValue>
    <PromptValue>Error</PromptValue>
    <PromptValue>Hold</PromptValue>
    <PromptValue>Queued</PromptValue>
    <PromptValue>Initiated</PromptValue>
    <PromptValue>Processing</PromptValue>
    <PromptValue>Cancelled</PromptValue>
    <PromptValue>Success</PromptValue>
  </prq:GetPromptValuesResponse>
</soapenv:Body>
</soapenv:Envelope>

```

PRCS_GETPROCESSNAMES

Use this service operation to retrieve a list of process names.

Request Message: PRQ_GETPROCESSNAMES_REQ

<i>Element Name</i>	<i>Description</i>
ProcessType	<p>Process type. Process type is case-sensitive.</p> <p>Use service operation PRCS_GETPROMPT with ParameterName <i>ProcessTypes</i> for a list of valid process types.</p>

Example Request:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETPROCESSNAMES_REQ.VERSIO
N_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:GetProcessNames>
      <ProcessType>XML Publisher</ProcessType>
    </prq:GetProcessNames>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: PRQ_GETPROCESSNAMES_RES

<i>Element Name</i>	<i>Description</i>
ProcessName	Returns a list of process names.

Example Response:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:GetProcessNamesResponse xsi:schemaLocation="http://xmlns.oracle.com/Ente

```

```
rprise/Tools/schemas/PRQ_GETPROCESSNAMES_RES.VERSION_1 PRQ_GETPROCESSNAMES_RES.VERSI
ION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETPROCE
SSNAMES_RES.VERSION_1">
  <ProcessName>PSXPQRYRPT</ProcessName>
  <ProcessName>QE_ETEXT</ProcessName>
  <ProcessName>QE_GLX8100</ProcessName>
  <ProcessName>QE_I9</ProcessName>
  <ProcessName>QE_LOCALE</ProcessName>
  <ProcessName>QE_TIMEZONE</ProcessName>
  <ProcessName>QE_UNICODE</ProcessName>
  <ProcessName>QE_XMD_BURST</ProcessName>
  <ProcessName>QE_XMD_XLS</ProcessName>
  <ProcessName>QE_XMLDOC</ProcessName>
  <ProcessName>XRFWIN</ProcessName>
</prq:GetProcessNamesResponse>
</soapenv:Body>
</soapenv:Envelope>
```

PRCS_GETPARAMS

Use this service operation to retrieve parameter values for a job or process. The job or process must have generic prompting enabled and runtime parameters defined.

Request Message: PRQ_GETPARAMS_REQ

<i>Element Name</i>	<i>Description</i>
JobName	Job name in upper case.
ProcessName	Process name in uppercase.
ProcessType	Process type. Use service operation PRCS_GETPROMPT with ParameterName <i>ProcessTypes</i> for a list of valid process types.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETPARAMS_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:GetRunTimeParametersRequest>
      <!--Optional:-->
      <ProcessName>PORTAL_CSS</ProcessName>
      <!--Optional:-->
      <ProcessType>Application Engine</ProcessType>
    </prq:GetRunTimeParametersRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_GETPARAMS_RES

<i>Element Name</i>	<i>Description</i>
ProcessName	Process name.
ProcessType	Process type.

Element Name	Description
RecordName	Prompt record name.
RecordFieldName	Prompt field name.
ParameterName	Parameter label.
ApplicationClassMethod	Application class method.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:GetRunTimeParametersResponse xsi:schemaLocation="http://xmlns.oracle.com
/Enterprise/Tools/schemas/PRQ_GETPARAMS_RES.VERSION_1 PRQ_GETPARAMS_RES.VERSION_1.x
sd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETPARAMS_RES.V
ERSION_1">
      <ProcessParameters>
        <ProcessName>PORTAL_CSS</ProcessName>
        <ProcessType>Application Engine</ProcessType>
        <RecordName>PORTAL_CSS_RUN</RecordName>
        <RecordFieldName>PORTAL_NAME</RecordFieldName>
        <ParameterName>Portal Name</ParameterName>
        <ApplicationClassMethod>.</ApplicationClassMethod>
      </ProcessParameters>
      <ProcessParameters>
        <ProcessName>PORTAL_CSS</ProcessName>
        <ProcessType>Application Engine</ProcessType>
        <RecordName>PORTAL_CSS_RUN</RecordName>
        <RecordFieldName>PORTAL_FLDRP_FLAG</RecordFieldName>
        <ParameterName>Delete invalid security</ParameterName>
        <ApplicationClassMethod>.</ApplicationClassMethod>
      </ProcessParameters>
    </prq:GetRunTimeParametersResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

PRCS_SCHEDULE

Use this service operation to schedule a job or process. Most of the elements are optional and required only if you want to override the default values for the process.

Request Message: PRQ_SCHEDULE_REQ

Element Name	Description
RunControlId	Run control ID.
Required	You can enter an existing run control ID or create a new one.
ProcessType	Process type.
Required	Use service operation PRCS_GETPROMPT with ParameterName <i>ProcessType</i> for a list of valid process types.

Element Name	Description
ProcessName Either a ProcessName or a JobName is required.	Process name. Use service operation PRCS_GETPROCESSNAMES if you are not sure of the process name. <hr/> Note: Process name is case-sensitive.
JobName Either a ProcessName or a JobName is required.	Job name. <hr/> Note: Job name is case-sensitive.
ServerName	Server name. Use service operation PRCS_GETPROMPT with ParameterName <i>ServerName</i> for a list of valid servers.
RunDateTime	Custom run date time in the format 2008-11-17 01.11.37.000000.
TimeZone	Time zone.
Recurrence	Recurrence Use service operation PRCS_GETPROMPT with ParameterName <i>RecurrenceName</i> for a list of valid recurrence definitions.
OutputDestinationType	If this element is omitted, the default output type is used. To override the default output, enter a valid output type for the process. See "Selecting Output Types and Formats" (PeopleTools 8.53: PeopleSoft Process Scheduler).
OutputDestinationFormat	If this element is omitted, the default output format is used. To override the default format, enter a valid output format for the process. See "Selecting Output Types and Formats" (PeopleTools 8.53: PeopleSoft Process Scheduler).
OutputDestinationString	If the output type is <i>File</i> or <i>Printer</i> , specify the file path or printer destination for the output.
ReportFolderName	Report folder name. This folder name applies to the main job or process. For a process within a job, enter the override for job item node (<JobItems> section).

Element Name	Description
ProcessFileName	Dependant file name. Use this value to override the file name in the process definition.
RetentionDays	Retention days. Note: Retention days applies to a process. For a process within a job, enter the override for job item (<JobItems> section).

<PrceDistributionOption> Use these elements to enter distribution overrides at the process level.

Process Distribution Option Elements	Description
PrceDistributionIdType	Specify either <i>User</i> or <i>Role</i> .
PrceDistributionId	Specify the corresponding distribution ID for the DistributionIdType.

<PrceOutputOption> Use these elements to modify the output options at the process level.

Process Output Option Elements	Description
PrceOutputDestinationType	If this element is omitted, the default output type is used. To override the default output, enter a valid output type for the process. See "Selecting Output Types and Formats" (PeopleTools 8.53: PeopleSoft Process Scheduler).
PrceOutputDestinationFormat	If this element is omitted, the default output format is used. To override the default format, enter a valid output format for the process. See "Selecting Output Types and Formats" (PeopleTools 8.53: PeopleSoft Process Scheduler).
PrceOutputDestinationString	If the output type is <i>File</i> or <i>Printer</i> , specify the file path or printer destination for the output.

<PrceEmailOption> Use these elements to enter email overrides at the process level.

Process Email Option Elements	Description
PrceEmailSubject	Enter the text that appears in the subject line of the email. If this element is omitted, the default text message is used.

Process Email Option Elements	Description
PrceEmailText	Enter text to appear in the body of the email. If this element is omitted, the default text message is used
PrceEmailAddresses	Enter a list of email addresses, separated by semicolons, to which the email should be sent.
PrceEmailAttachLog	Enter <i>true</i> to include log files resulting from the Structured Query Report (SQR) program as attachments to the email file (SQR only).
PrceEmailWebReport	Enter <i>true</i> to send an email with a link to the completed report output. This option is available only when the output type for the request is Web.

<JobItem> If the request is for a job, use these elements to enter the job information.

Job Item Elements	Description
JobName	Job name.
ProcessItemLevel	Specify the job item's process level within the main job as a number.
JobSequenceNumber	Specify the parent item's sequence number as a number.
SeqNoInItemLevel	Specify the item's sequence number as a number.
ReportFolderName	Enter the name of the folder in which the report results are posted.
RetentionDays	Enter the number of days before the files that are on the report repository are deleted. If no value is entered, the default retention days for the process is used.

<DistributionOption> Use these elements to modify the distribution list for job items.

Job Item Distribution Option Elements	Description
DistributionIdType	Specify either <i>User</i> or <i>Role</i> .
DistributionId	Specify the corresponding distribution ID for the DistributionIdType.

<OutputOption> Use these elements to modify the output options for job items.

Job Item Output Option Elements	Description
OutputDestinationType	If this element is omitted, the default output type is used. To override the default output, enter a valid output type for the process. See "Selecting Output Types and Formats" (PeopleTools 8.53: PeopleSoft Process Scheduler).
OutputDestinationFormat	If this element is omitted, the default output format is used. To override the default format, enter a valid output format for the process. See "Selecting Output Types and Formats" (PeopleTools 8.53: PeopleSoft Process Scheduler).
OutputDestinationString	If the output type is <i>File</i> or <i>Printer</i> , specify the file path or printer destination for the output.

<EmailOption> Use these elements to enter email options for job items.

Job Item Email Option Elements	Description
EmailSubject	Enter the text that appears in the subject line of the email. If this element is omitted, the default text message is used.
EmailText	Enter text to appear in the body of the email. If this element is omitted, the default text message is used
EmailAddresses	Enter a list of email addresses, separated by semicolons, to which the email should be sent.
EmailAttachLog	Enter <i>true</i> to include log files resulting from the Structured Query Report (SQR) program as attachments to the email file (SQR only).
EmailWebReport	Enter <i>true</i> to send an email with a link to the completed report output. This option is available only when the output type for the request is Web.

<RunTimeParameters> Use these elements to enter runtime parameters.

RunTime Parameter Elements	Description
ProcessName	Process name. Process name is case-sensitive.
ProcessType	Process type.

RunTime Parameter Elements	Description
ParameterName	Parameter name.
ParameterValue	Parameter value.

Example Request:

This is an example request for a process with no runtime parameters or overrides:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:Schedule>
      <RunControlId>TEST1</RunControlId>
      <ProcessType>Application Engine</ProcessType>
      <!--Optional:-->
      <ProcessName>AEMINITEST</ProcessName>
    </prq:Schedule>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_SCHEDULE_RES

Element	Description
ProcessInstance	Process instance number.
JobName	Job name.
ProcessName	Name of process within the job.
ProcessType	Process type.
JobSequenceNumber	Sequence number within the job.
ProcessItemLevel	Process item level within the job.
ProcessInstance	Process instance number for the job item.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:ScheduleResponse xsi:schemaLocation="http://xmlns.oracle.com/Enterprise/=
Tools/schemas/PRQ_SCHEDULE_RES.VERSION_1 PRQ_SCHEDULE_RES.VERSION_1.xsd" xmlns:prq=
"http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_RES.VERSION_1">
      <ProcessInstance>33</ProcessInstance>
    </prq:ScheduleResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

Schedule a Process with Prompts

This is an example of a request for a process that has two parameters:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:Schedule>
      <RunControlId>SYNC</RunControlId>
      <ProcessType>Application Engine</ProcessType>
      <!--Optional:-->
      <ProcessName>PORTAL_CSS</ProcessName>
      <!--Zero or more repetitions:-->
      <RunTimeParameter>
        <ProcessName>PORTAL_CSS</ProcessName>
        <ProcessType>Application Engine</ProcessType>
        <ParameterName>Portal Name</ParameterName>
        <ParameterValue>EMPLOYEE</ParameterValue>
      </RunTimeParameter>
      <RunTimeParameter>
        <ProcessName>PORTAL_CSS</ProcessName>
        <ProcessType>Application Engine</ProcessType>
        <ParameterName>Delete invalid security</ParameterName>
        <ParameterValue>Y</ParameterValue>
      </RunTimeParameter>
    </prq:Schedule>
  </soapenv:Body>
</soapenv:Envelope>
```

Schedule a Job with Overrides for Output

This is an example to schedule a job with overrides for report distribution, output type and format for one of the processes within the job.:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:Schedule>
      <RunControlId>TEST1</RunControlId>
      <ProcessType>PSJob</ProcessType>
      <!--Optional:-->
      <JobName>3SQR</JobName>
      <JobItem>
        <JobName>XRFIELDS</JobName>
        <!--Optional:-->
        <ProcessItemLevel>1</ProcessItemLevel>
        <!--Optional:-->
        <JobSequenceNumber>1</JobSequenceNumber>
        <!--Optional:-->
        <SeqNoInItemLevel>1</SeqNoInItemLevel>
        <!--Optional:-->
        <DistributionOption>
          <DistributionIdType>User</DistributionIdType>
          <DistributionId>QEDMO</DistributionId>
        </DistributionOption>
        <OutputOption>
          <OutputDestinationType>WEB</OutputDestinationType>
          <OutputDestinationFormat>PDF</OutputDestinationFormat>
        </OutputOption>
      </JobItem>
    </prq:Schedule>
  </soapenv:Body>
</soapenv:Envelope>
```

The response for a job will include the process instance ID at each level:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:
soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/
2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:ScheduleResponse xsi:schemaLocation="http://xmlns.oracle.com/Enterprise
/Tools/schemas/PRQ_SCHEDULE_RES.VERSION_1 PRQ_SCHEDULE_RES.VERSION_1.xsd" xmlns:
prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_RES.VERSION_1">
      <ProcessInstance>21</ProcessInstance>
      <JobItem>
        <JobName>3SQR</JobName>
        <ProcessName>3SQR</ProcessName>
        <ProcessType>PSJob</ProcessType>
        <JobSequenceNumber>0</JobSequenceNumber>
        <ProcessItemLevel>0</ProcessItemLevel>
        <ProcessInstance>21</ProcessInstance>
      </JobItem>
      <JobItem>
        <JobName>3SQR</JobName>
        <ProcessName>XRFIELDS</ProcessName>
        <ProcessType>SQR Report</ProcessType>
        <JobSequenceNumber>1</JobSequenceNumber>
        <ProcessItemLevel>1</ProcessItemLevel>
        <ProcessInstance>22</ProcessInstance>
      </JobItem>
      <JobItem>
        <JobName>3SQR</JobName>
        <ProcessName>XRFMENU</ProcessName>
        <ProcessType>SQR Report</ProcessType>
        <JobSequenceNumber>2</JobSequenceNumber>
        <ProcessItemLevel>1</ProcessItemLevel>
        <ProcessInstance>23</ProcessInstance>
      </JobItem>
      <JobItem>
        <JobName>3SQR</JobName>
        <ProcessName>XRFRCFL</ProcessName>
        <ProcessType>SQR Report</ProcessType>
        <JobSequenceNumber>3</JobSequenceNumber>
        <ProcessItemLevel>1</ProcessItemLevel>
        <ProcessInstance>24</ProcessInstance>
      </JobItem>
    </prq:ScheduleResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

PRCS_SCHEDULE_PSQUERY

This service operation schedules PSQuery process.

Request Message: PRQ_SCHEDULE_PSQUERY_REQ

<i>Element Name</i>	<i>Description</i>
QueryName	Specifies a string representing the name of a query.
RunControlId	Specifies a new or an existing run control ID.
PromptName	Specifies the name of the prompt for this query.
PromptValue	Specifies prompt value for its matching prompt names.

Element Name	Description
FieldName	Specifies the field name for the query tree prompt.
ExpressionText	Specifies the query expression text for the query tree prompt.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_PSQUERY_REQ.VERSION_1"
ON_1">
<soapenv:Header xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <wsse:Security soap:mustUnderstand="1" xmlns:soap="http://schemas.xmlsoap.org/w
sdl/soap/" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wsse
curity-secext-1.0.xsd">
    <wsse:UsernameToken wsu:Id="UsernameToken-1" xmlns:wsu="http://docs.oasis-ope
n.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
      <wsse:Username>QE_QUERY</wsse:Username>
      <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-ws
s-username-token-profile-1.0#PasswordText">QE_QUERY</wsse:Password>
    </wsse:UsernameToken>
  </wsse:Security>
</soapenv:Header>
<soapenv:Body>
  <prq:SchedulePSQuery>
    <QueryName>QE_TREE_OPT_PROMPTS</QueryName>
    <Prompts>
      <PromptName>SetID</PromptName>
      <PromptValue>QEDM1</PromptValue>
    </Prompts>
    <TreePrompts>
      <FieldName>A.QE_JOBCODE</FieldName>
      <ExpressionText>,,QE_JOBCODES,1999-01-01,ALL_JOBS</ExpressionText>
    </TreePrompts>
    <TreePrompts>
      <FieldName>C.DEPTID</FieldName>
      <ExpressionText>SHARE,,DEPARTMENTS,1900-01-01,DEPARTMENTS</ExpressionTe
xt>
    </TreePrompts>
    <RunControlId>QE_TREE_OPT_PROMPTS</RunControlId>
  </prq:SchedulePSQuery>
</soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_SCHEDULE_PSQUERY_RES

Element Name	Description
ProcessInstance	Specifies a number representing a process instance for running the report.

Example Response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:SchedulePSQueryResponse xsi:schemaLocation="http://xmlns.oracle.com/Ente
rprise/Tools/schemas/PRQ_SCHEDULE_PSQUERY_RES.VERSION_1 PRQ_SCHEDULE_PSQUERY_RES.VE
RSION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDU
LE_PSQUERY_RES.VERSION_1">
      <ProcessInstance>11078</ProcessInstance>
```

```

    </prq:SchedulePSQueryResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

PRCS_SCHEDULE_CONQUERY

This service operation schedules non unique connected query process.

Request Message: PRQ_SCHEDULE_CONQUERY_REQ

Element Name	Description
ConQueryName	Specifies the connected query name.
RunControlId	Specifies a new or an existing run control ID.
QueryName	Specifies the name of a query that participates in the connected query.
PromptName	Specifies name of the prompt for this query.
PromptValue	Specifies prompt value for its matching query prompt.
DistributionIdType	Specify either <i>User</i> or <i>Role</i> .

Example Request:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_CONQUERY_REQ.VERS=
ION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:ScheduleConQuery>
      <ConQueryName>APX1090</ConQueryName>
      <RunParam>
        <QueryName>APY1090_MATCH_EXCEPTIONS</QueryName>
        <Prompts>
          <PromptName>MatchStatus</PromptName>
          <PromptValue>Matched</PromptValue>
        </Prompts>
        <Prompts>
          <PromptName>RunControlId</PromptName>
          <PromptValue>PAYCYCLE</PromptValue>
        </Prompts>
      </RunParam>
      <RunControlId>APX1090</RunControlId>
    </prq:ScheduleConQuery>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: PRQ_SCHEDULE_CONQUERY_RES

Element Name	Description
ProcessInstance	Specifies a number representing a process instance for running the report.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:ScheduleConQueryResponse xsi:schemaLocation="http://xmlns.oracle.com/Ent
erprise/Tools/schemas/PRQ_SCHEDULE_CONQUERY_RES.VERSION_1 PRQ_SCHEDULE_CONQUERY_RES
.VERSION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCH
EDULE_CONQUERY_RES.VERSION_1">
      <ProcessInstance>11073</ProcessInstance>
    </prq:ScheduleConQueryResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

PRCS_SCHEDULE_BIPUBLISHER

This service operation schedules non unique BI Publisher processe.

Request Message: PRQ_SCHEDULE_BIPUBLISHER_REQ

<i>Element Name</i>	<i>Description</i>
ReportName	Specifies the name of the report.
RunControlId	Specifies a new or an existing run control ID.
PromptName	Specifies name of the prompt for the report.
PromptValue	Specifies prompt value for its matching query prompt.
DistributionIdType	Specify either <i>User</i> or <i>Role</i> .

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_PSQUERY_REQ.VERSI
ON_1">
  <soapenv:Header xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
    <wsse:Security soap:mustUnderstand="1" xmlns:soap="http://schemas.xmlsoap.org/w
sdl/soap/" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wsse
curity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken-1" xmlns:wsu="http://docs.oasis-ope
n.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>VP1</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-ws
s-username-token-profile-1.0#PasswordText">VP1</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body>
    <prq:ScheduleBIPublisher>
      <ReportName>AMDE5001</ReportName>
      <RunParam>
        <Prompts>
          <PromptName>OPRID</PromptName>
          <PromptValue>VP1</PromptValue>
        </Prompts>
        <Prompts>
          <PromptName>RUN_CNTL_ID</PromptName>
```

```

        <PromptValue>test</PromptValue>
    </Prompts>
</RunParam>
    <RunControlId>AMDE5001</RunControlId>
</prq:ScheduleBIPublisher>
</soapenv:Body>
</soapenv:Envelope>

```

Response Message: PRQ_SCHEDULE_BIPUBLISHER_RES

<i>Element Name</i>	<i>Description</i>
ProcessInstance	Specifies a number representing a process instance for running the report.

Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:ScheduleBIPublisherResponse xsi:schemaLocation="http://xmlns.oracle.com/En
terprise/Tools/schemas/PRQ_SCHEDULE_BIPUBLISHER_RES.VERSION_1 PRQ_SCHEDULE_BIPUBL
ISHER_RES.VERSION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schema
s/PRQ_SCHEDULE_BIPUBLISHER_RES.VERSION_1">
      <ProcessInstance>10570</ProcessInstance>
    </prq:ScheduleBIPublisherResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

PRCS_RUNJOBSETNOW

This service operation schedules a JobSet based on the settings as defined in the Scheduled JobSet Definition component.

Request Message: PRQ_RUNJOBSETNOW_REQ

<i>Element Name</i>	<i>Description</i>
MainSchedName	Specifies the name of the jobset schedule.
MainJobName	Specifies the name of job within the jobset schedule.

Example Request

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_SCHEDULE_PSQUERY_REQ.VERSI
ON_1">
  <soapenv:Header xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
    <wsse:Security soap:mustUnderstand="1" xmlns:soap="http://schemas.xmlsoap.org/w
sdl/soap/" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wsse
curity-secext-1.0.xsd">
      <wsse:UsernameToken wsu:Id="UsernameToken-1" xmlns:wsu="http://docs.oasis-ope
n.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsse:Username>VP1</wsse:Username>
        <wsse:Password Type="http://docs.oasis-open.org/wss/2004/01/oasis-200401-ws
s-username-token-profile-1.0#PasswordText">VP1</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>

```



```

</soapenv:Header>
<soapenv:Body>
  <prq:RunJobSetNow>
    <MainSchedName>js1</MainSchedName>
    <MainJobName>j1</MainJobName>
  </prq:RunJobSetNow>
</soapenv:Body>
</soapenv:Envelope>

```

Response Message: PRQ_RUNJOBSETNOW_RES

Element Name	Description
ProcessInstance	Specifies a number representing a process instance for running the report.

Example Response

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:RunJobSetNowResponse xsi:schemaLocation="http://xmlns.oracle.com/Enterpr
ise/Tools/schemas/PRQ_RUNJOBSETNOW_RES.VERSION_1 PRQ_RUNJOBSETNOW_RES.VERSION_1.xsd
" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_RUNJOBSETNOW_RES.
VERSION_1">
      <ProcessInstance>10554</ProcessInstance>
    </prq:RunJobSetNowResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

Monitoring a Request

Client applications can use service operations to monitor a process and get a report. This section describes the service operations available to monitor the request. These include:

- PRCS_FINDREQUESTS
- PRCS_GETREQUEST
- PRCS_UPDATEREQUEST
- PRCS_GETREPORT
- PRCS_GETSERVERCONSTRAINTS
- PRCS_GETOUTPUTPARAMS
- PRCS_PRINT_JOB_HTML_REPORT
- PRCS_PRINT_JOB_REQUEST_REPORT
- PRCS_PRINT_SCHEDULE_HTML_REPORT

PRCS_FINDREQUESTS

Use this service operation to find process requests.

Request Message: PRQ_FINDREQUESTS_REQ

Element Name	Description
ProcessInstanceFrom	Enter the process instance number for the beginning of a range of instances by which to limit the process requests that are returned.
ProcessInstanceTo	Enter the process instance number for the end of a range of instances by which to limit the process requests that are returned.
OperatorId	Enter the ID of the user who submitted the process. Leave blank to view all of the processes that you are authorized to view. Use service operation PRCS_GETPROMPT with ParameterName <i>OperatorId</i> for a list of valid user IDs.
ProcessType	Enter a process type to limit the returned list to a specific process type.
ProcessName	Enter a process name to limit the returned list to a specific process name. Process name is case-sensitive.
ServerRun	Enter the name of a server to limit the returned list to a specific server.
RunStatus	Enter the run status to limit the returned list to a specific status. Use service operation PRCS_GETPROMPT with ParameterName <i>RunStatus</i> for a list of valid run statuses.
DistributionStatus	Enter the distribution status to limit the returned list to a specific distribution status. Use service operation PRCS_GETPROMPT with ParameterName <i>DistributionStatus</i> for a list of valid distribution statuses.
ProcessRequestCount	Enter a number to limit the returned list.

Element Name	Description
IsProcessCount	Use this parameter to receive all data and process count. Specify <i>0</i> to receive all data but no ProcessCount. Specify <i>1</i> to receive all data and ProcessCount.

<FilterType> The filter type can be either Last or Date Range

For <Last> specify an interval of time by which to limit the process requests that appear in the list.

Last Elements	Description
FilterValue	Enter a custom numerical value.
FilterUnit	Enter the interval. Valid values are: <ul style="list-style-type: none"> • <i>All</i> • <i>Days</i> • <i>Hours</i> • <i>Minutes</i>

Date Range

For <CustomDates> specify dates to get a list of requests created between two specific dates.

Date Range Elements	Description
FromDate	Enter a from date.
ToDate	Enter a to date.

Example Request:

```
<?xml version="1.0"?>
<soapenv:Envelope xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:s=
oapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:wsa="http://schemas.xmlsoa
p.org/ws/2003/03/addressing/" xmlns:xsd="http://www.w3.org/2001/XMLSchema/" xmlns:x=
si="http://www.w3.org/2001/XMLSchema-instance/">
  <soapenv:Header xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
    <wsse:Security soap:mustUnderstand="1" xmlns:soap="http://schemas.xmlsoap.org/w
sdl/soap/" xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wsse
curity-secext-1.0.xsd">
      <wsse:UsernameToken>
        <wsse:Username>QEDMO</wsse:Username>
        <wsse:Password>QEDMO</wsse:Password>
      </wsse:UsernameToken>
    </wsse:Security>
  </soapenv:Header>
  <soapenv:Body xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
    <prq:FindRequests xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/P
RQ_FINDREQUESTS_REQ.VERSION_1">
      <OperatorId>QEDMO</OperatorId>
```

```

<ProcessType>SQR Report</ProcessType>
<ProcessName></ProcessName>
<RunStatus>Success</RunStatus>
<DistributionStatus>Posted</DistributionStatus>
<FilterType>
  <CustomDates>
    <FromDate>2009-02-01</FromDate>
    <ToDate>2009-02-10</ToDate>
  </CustomDates>
</FilterType>
</prq:FindRequests>
</soapenv:Body>
</soapenv:Envelope>

```

Response Message: PRQ_FINDREQUESTS_RES

Element Name	Description
ProcessInstance	Process instance.
ProcessName	Process name.
ProcessType	Process type.
RunControlID	Run control ID.
ServerNameRequested	Name of the server requested for the process.
ServerNameRun	Name of the server where the process was run.
OperatorID	ID that submitted the process.
RequestDateTime	Date and time that the process request was submitted.
RunDateTime	The date and time that the process request was created.
BeginDateTime	The date and time that the process started.
EndDateTime	The date and time that the process ended.
OutputDestinationFormat	Output format.
OutputDestinationType	Output type.
Recurrence	Recurrence.
RunStatus	Run status.
DistributionStatus	Distribution status.

Element Name	Description
ContentId	Content ID is the Report ID assigned to the individual report by the Process Scheduler. The Content ID is used to view reports in the Report Repository. Note: For bursted reports all the associated content IDs will be retrieved and will be a part of the response.
IsProcessCount	Process count.

Example Response:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:FindRequestsResponse xsi:schemaLocation="http://xmlns.oracle.com/Enterpr
ise/Tools/schemas/PRQ_FINDREQUESTS_RES.VERSION_1 PRQ_FINDREQUESTS_RES.VERSION_1.xsd"
  xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_FINDREQUESTS_RES.
VERSION_1">
      <ProcessRequestItem>
        <ProcessInstance>9999941</ProcessInstance>
        <ProcessName>DDDAUDIT</ProcessName>
        <ProcessType>SQR Report</ProcessType>
        <RunControlId>2</RunControlId>
        <ServerNameRequested>PSNT</ServerNameRequested>
        <ServerNameRun>PSNT</ServerNameRun>
        <OperatorId>QEDMO</OperatorId>
        <RequestDateTime>2009-02-03-13.12.33.921000</RequestDateTime>
        <RunDateTime>2009-02-03-13.12.19.000000</RunDateTime>
        <BeginDateTime>2009-02-03-13.13.00.000000</BeginDateTime>
        <EndDateTime>2009-02-03-13.13.12.640000</EndDateTime>
        <OutputDestinationFormat>PDF</OutputDestinationFormat>
        <OutputDestinationType>WEB</OutputDestinationType>
        <RecurrenceName/>
        <RunStatus>Success</RunStatus>
        <DistributionStatus>Posted</DistributionStatus>
        <ContentId>9999905</ContentId>
      </ProcessRequestItem>
      <ProcessRequestItem>
        <ProcessInstance>9999948</ProcessInstance>
        <ProcessName>SYSAUDIT</ProcessName>
        <ProcessType>SQR Report</ProcessType>
        <RunControlId>2</RunControlId>
        <ServerNameRequested>PSNT</ServerNameRequested>
        <ServerNameRun>PSNT</ServerNameRun>
        <OperatorId>QEDMO</OperatorId>
        <RequestDateTime>2009-02-05-09.53.32.296000</RequestDateTime>
        <RunDateTime>2009-02-05-09.52.57.000000</RunDateTime>
        <BeginDateTime>2009-02-05-09.54.33.000000</BeginDateTime>
        <EndDateTime>2009-02-05-09.56.28.640000</EndDateTime>
        <OutputDestinationFormat>PDF</OutputDestinationFormat>
        <OutputDestinationType>WEB</OutputDestinationType>
        <RecurrenceName/>
        <RunStatus>Success</RunStatus>
        <DistributionStatus>Posted</DistributionStatus>
        <ContentId>9999914</ContentId>
      </ProcessRequestItem>
    </prq:FindRequestsResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

PRCS_GETREQUEST

This service operation is used to get the process request status.

Request Message: PRQ_GETREQUEST_REQ

<i>Element Name</i>	<i>Description</i>
ProcessInstance	Process instance number.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETREQUEST_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:GetRequest>
      <ProcessInstance>9999951</ProcessInstance>
    </prq:GetRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_GETREQUEST_RES

<i>Element Name</i>	<i>Description</i>
ProcessInstance	Process instance.
ProcessName	Process name.
ProcessType	Process type.
RunControlID	Run control ID.
ServerNameRun	Server name.
OperatorID	ID of the user who submitted the process.
RequestDateTime	The date and time that the process request was submitted.
RunDateTime	The date and time that the process request was scheduled to run.
BeginDateTime	The date and time that the process started.
EndDateTime	The date and time that the process ended.
OutputDestinationFormat	Output format.
OutputDestinationType	Output type.
RunStatus	Run status.

Element Name	Description
DistributionStatus	Distribution status.
ContentId	Report ID assigned in the Report Repository. Note: For bursted reports all the associated content IDs will be retrieved and will be a part of the response.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:GetRequestResponse xsi:schemaLocation="http://xmlns.oracle.com/Enterpris
e/Tools/schemas/PRQ_GETREQUEST_RES.VERSION_1 PRQ_GETREQUEST_RES.VERSION_1.xsd" xmlns:
s:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETREQUEST_RES.VERSION_
1">
      <ProcessRequestItem>
        <ProcessInstance>9999951</ProcessInstance>
        <ProcessName>DDDAUDIT</ProcessName>
        <ProcessType>SQR Report</ProcessType>
        <RunControlId>TEST</RunControlId>
        <ServerNameRun>PSNT</ServerNameRun>
        <OperatorId>QEMGR</OperatorId>
        <RequestDateTime>2009-02-05-11.32.37.125000</RequestDateTime>
        <RunDateTime>2009-02-05-11.32.36.000000</RunDateTime>
        <BeginDateTime>2009-02-05-11.33.01.000000</BeginDateTime>
        <EndDateTime>2009-02-05-11.33.25.468000</EndDateTime>
        <OutputDestinationFormat>PDF</OutputDestinationFormat>
        <OutputDestinationType>WEB</OutputDestinationType>
        <RunStatus>Success</RunStatus>
        <DistributionStatus>Posted</DistributionStatus>
        <ContentId>9999917</ContentId>
      </ProcessRequestItem>
    </prq:GetRequestResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

PRCS_UPDATEREQUEST

This service operation is used to update a request.

Request Message: PRQ_UPDATEREQUEST_REQ

Element Name	Description
ProcessInstance	The process instance to update.

Element Name	Description
RunStatus	Indicate the new status. Valid values are: <ul style="list-style-type: none"> Deleted Cancelled Hold Restart

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_UPDATEREQUEST_REQ.VERSION_
1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:UpdateProcessRequest>
      <ProcessInstance>9999940</ProcessInstance>
      <RunStatus>Deleted</RunStatus>
    </prq:UpdateProcessRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_UPDATEREQUEST_RES

Element Name	Description
RequestStatus	Returns the status of either <i>Success</i> or <i>Failure</i> .

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:UpdateProcessRequestResponse xsi:schemaLocation="http://xmlns.oracle.com/
Enterprise/Tools/schemas/PRQ_UPDATEREQUEST_RES.VERSION_1 PRQ_UPDATEREQUEST_RES.VER
SION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_UPDATER
EQUEST_RES.VERSION_1">
      <RequestStatus>Success</RequestStatus>
    </prq:UpdateProcessRequestResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

PRCS_GETREPORT

This service operation is used to retrieve a report from the report repository.

Request Message: PRQ_GETREPORT_REQ

<i>Element Name</i>	<i>Description</i>
ContentID	The content ID for the report. Use service operation PRCS_GETREQUEST to retrieve the content ID for the report.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETREPORT_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:GetReport>
      <ContentId>9999917</ContentId>
    </prq:GetReport>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message:

<i>Element Name</i>	<i>Description</i>
Content ID	Content ID.
Report URL	Returns the URL to the report, log, and trace files.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:GetReportResponse xsi:schemaLocation="http://xmlns.oracle.com /Enterpris
e/Tools/schemas/PRQ_GETREPORT_RES.VERSION_1 PRQ_GETREPORT_RES.VERSION_1.xsd" xmlns:
prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETREPORT_RES.VERSION_1">
      <Report>
        <ContentID>9999917</ContentID>
        <ReportDetails>
          <ReportURL>//ple-infodev-08.peoplesoft.com:8010/psreports/QEDMO/9999
917/DDDAUDIT_9999951.PDF</ReportURL>
          <ReportDescription>Portable Document Format</ReportDescription>
        </ReportDetails>
        <ReportDetails>
          <ReportURL>//ple-infodev-08.peoplesoft.com:8010/psreports/QEDMO/9999
917/DDDAUDIT_9999951.out</ReportURL>
          <ReportDescription>Trace File</ReportDescription>
        </ReportDetails>
        <ReportDetails>
          <ReportURL>//ple-infodev-08.peoplesoft.com:8010/psreports/QEDMO/9999
917/SQR_DDDAUDIT_9999951.log</ReportURL>
          <ReportDescription>Message Log</ReportDescription>
        </ReportDetails>
      </Report>
    </prq:GetReportResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

PRCS_GETSERVERCONSTRAINTS

This service operation retrieves server specific constraints data from the database. As a response, Max load and current load information will be returned in the XML response message.

Request Message: PRQ_GETSERVERCONSTRAINTS_REQ

<i>Element Name</i>	<i>Description</i>
ServerName	Specifies the server name for the service operation PRCS_GETSERVERCONSTRAINTS.
ProcessType	Specifies the process type for the service operation PRCS_GETSERVERCONSTRAINTS.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETSERVERCONSTRAINTS_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:GetServerConstraints>
      <!--Optional:-->
      <ServerName>PSNT</ServerName>
    </prq:GetServerConstraints>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_GETSERVERCONSTRAINTS_RES

<i>Element Name</i>	<i>Description</i>
RaiseAlert	Specifies the server suspended state.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:GetServerConstraintsResponse xsi:schemaLocation="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETSERVERCONSTRAINTS_RES.VERSION_1 PRQ_GETSERVERCONSTRAINTS_RES.VERSION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GETSERVERCONSTRAINTS_RES.VERSION_1">
      <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>Application Engine</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
      </ServerInfo>
      <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>COBOL SQL</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
      </ServerInfo>
      <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>Crw Online</ProcessType>
      </ServerInfo>
    </prq:GetServerConstraintsResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```

        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>Crystal</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>Cube Builder</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>Data Mover</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>Essbase Cube Builder</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>Optimization Engine</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>2</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>PSJob</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>SQR Process</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>SQR Report</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>SQR Report For WF Delivery</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>Winword</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>XML Publisher</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>

```

```

        <ProcessType>nVision-Report</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
    <ServerInfo>
        <ServerName>PSNT</ServerName>
        <ProcessType>nVision-ReportBook</ProcessType>
        <CurrentLoad>0</CurrentLoad>
        <MaxLoad>3</MaxLoad>
    </ServerInfo>
</prq:GetServerConstraintsResponse>
</soapenv:Body>
</soapenv:Envelope>
<

```

PRCS_GETOUTPUTTYPE

This service operation retrieves output types.

Request Message: PRCS_GET_OUTPUTTYPES_REQ

<i>Element Name</i>	<i>Description</i>
ProcessType	Specifies the name of a predefined process type as a string.
ProcessName	Specifies the name of a predefined process as a string.
LangCode	Specifies the language code associated with the task to be enqueued.

Example Request:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GET_OUTPUTTYPES_REQ.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:GetOutputTypes>
      <ProcessType>AppEngine</ProcessType>
    </prq:GetOutputTypes>
  </soapenv:Body>
</soapenv:Envelope>

```

Response Message: PRCS_GET_OUTPUTTYPES_RES

<i>Element Name</i>	<i>Description</i>
OutputTypeValue	Specifies the output type as a string.
IsActive	Specifies the active type.
IsDefaultType	Specifies the default type.

Example Response:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
openc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200

```

```

1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:GetOutputTypesResponse xsi:schemaLocation="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GET_OUTPUTTYPES_RES.VERSION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GET_OUTPUTTYPES_RES.VERSION_1">
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>-</ProcessName>
        <OutputTypeValue>FILE</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>-</ProcessName>
        <OutputTypeValue>WINDOW</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>-</ProcessName>
        <OutputTypeValue>EMAIL</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>-</ProcessName>
        <OutputTypeValue>WEB</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>1</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSCONQRS</ProcessName>
        <OutputTypeValue>FILE</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSCONQRS</ProcessName>
        <OutputTypeValue>WINDOW</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSCONQRS</ProcessName>
        <OutputTypeValue>WEB</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>1</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSQUERY</ProcessName>
        <OutputTypeValue>FILE</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSQUERY</ProcessName>
        <OutputTypeValue>WINDOW</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
      </OutputTypeInfo>
      <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSQUERY</ProcessName>

```

```

        <OutputTypeValue>EMAIL</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
    </OutputTypeInfo>
    <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSQUERY</ProcessName>
        <OutputTypeValue>WEB</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>1</IsDefaultType>
    </OutputTypeInfo>
    <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSQUERY</ProcessName>
        <OutputTypeValue>FEED</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
    </OutputTypeInfo>
    <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSXPQRYRPT</ProcessName>
        <OutputTypeValue>FILE</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
    </OutputTypeInfo>
    <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSXPQRYRPT</ProcessName>
        <OutputTypeValue>PRINTER</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
    </OutputTypeInfo>
    <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSXPQRYRPT</ProcessName>
        <OutputTypeValue>WINDOW</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
    </OutputTypeInfo>
    <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSXPQRYRPT</ProcessName>
        <OutputTypeValue>EMAIL</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>0</IsDefaultType>
    </OutputTypeInfo>
    <OutputTypeInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSXPQRYRPT</ProcessName>
        <OutputTypeValue>WEB</OutputTypeValue>
        <IsActive>1</IsActive>
        <IsDefaultType>1</IsDefaultType>
    </OutputTypeInfo>
</prq:GetOutputTypesResponse>
</soapenv:Body>
</soapenv:Envelope>

```

PRCS_GETOUTPUTFORMATS

This service operation retrieves output format for a given process.

Request Message: PRQ_GET_OUTPUTFORMATS_REQ

<i>Element Name</i>	<i>Description</i>
ProcessType	Specifies the process type. Process type is case sensitive.

Element Name	Description
ProcessName	Specifies the process name.
OutDestType	Specifies the output type to filter the response.
LangCode	Specifies the language code associated with the task to be enqueued.

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
prq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_GET_OUTPUTFORMATS_REQ.VERSION_1"
ION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:GetOutputFormats>
      <ProcessType>AppEngine</ProcessType>
      <!--Optional:-->
      <ProcessName>PSCONQRS</ProcessName>
      <!--Optional:-->
      <LangCode>FRA</LangCode>
    </prq:GetOutputForma t s>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_GET_OUTPUTFORMATS_RES

Element Name	Description
OutputTypeValue	Specifies the output type value.
OutputFormatValue	Specifies the output format value.
IsDefaultType	Specifies the default value of the output format.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:GetOutputFormatsResponse xsi:schemaLocation="http://xmlns.oracle.com/Ent
erprise/Tools/schemas/PRQ_GET_OUTPUTFORMATS_RES.VERSION_1.xsd" xmlns:prq="http://xm
lns.oracle.com/Enterprise/Tools/schemas/PRQ_GET_OUTPUTFORMATS_RES.VERSION_1">
      <OutputFormatInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSCONQRS</ProcessName>
        <OutputTypeValue>FILE</OutputTypeValue>
        <OutputFormatValue>XML Format (*.xml)</OutputFormatValue>
        <IsActive>1</IsActive>
        <IsDefaultType>1</IsDefaultType>
      </OutputFormatInfo>
      <OutputFormatInfo>
        <ProcessType>AppEngine</ProcessType>
        <ProcessName>PSCONQRS</ProcessName>
        <OutputTypeValue>WINDOW</OutputTypeValue>
        <OutputFormatValue>XML Format (*.xml)</OutputFormatValue>
        <IsActive>1</IsActive>
        <IsDefaultType>1</IsDefaultType>
    </prq:GetOutputFormatsResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    </OutputFormatInfo>
  <OutputFormatInfo>
    <ProcessType>AppEngine</ProcessType>
    <ProcessName>PSCONQRS</ProcessName>
    <OutputTypeValue>WEB</OutputTypeValue>
    <OutputFormatValue>XML Format (*.xml)</OutputFormatValue>
    <IsActive>1</IsActive>
    <IsDefaultType>1</IsDefaultType>
  </OutputFormatInfo>
</prq:GetOutputFormatsResponse>
</soapenv:Body>
</soapenv:Envelope>

```

PRCS_PRINT_JOB_HTML_REPORT

This service operations retrieves an HTML formatted report in a tree displaying all items in a job or jobset, as defined in the Job Definition component.

Request Message: PRQ_PRINTJOBHTMLREPORT_REQ

<i>Element Name</i>	<i>Description</i>
JobName	Specifies the name of the job.

Example Request:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_PRINT_JOBHTMLREPORT_REQ.VE
RSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:PrintJobHTMLReport>
      <JobName>3SQR</JobName>
    </prq:PrintJobHTMLReport>
  </soapenv:Body>
</soapenv:Envelope>

```

PRQ_PRINTJOBHTMLREPORT_RES

<i>Element Name</i>	<i>Description</i>
HTMLReport	Returns an HTML formatted report in a tree displaying all items in a job or jobset.

Example Response:

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Server</faultcode>
      <faultstring>null</faultstring>
      <detail>
        <IBResponse type="error">
          <DefaultTitle>Integration Broker Response</DefaultTitle>
          <StatusCode>20</StatusCode>
          <MessageID>554</MessageID>
          <DefaultMessage>Encryption and Digital Signed or Https required for =>
Service Operation PRCS_PRINT_JOBHTMLREPORT. (158,554)</DefaultMessage>
          <MessageParameters>
            <Parameter>PRCS_PRINT_JOBHTMLREPORT</Parameter>

```



```

        </MessageParameters>
    </IBResponse>
</detail>
</SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

PRCS_PRINT_JOB_REQUEST_REPORT

This service operation retrieves an html formatted report displaying the current status of a specific process, job, and jobset.

Request Message: PRQ_PRINTJOBREQUESTREPORT_REQ

<i>Element Name</i>	<i>Description</i>
ProcessInstance	Specifies the process instance number of the scheduled job, jobset or process.
ItemInstance	Specifies the item instance number of the scheduled job or jobset.
PrintJobTree	<p>Specify whether to have the job tree displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • <i>1</i> to display the tree. • <i>0</i> to hide the tree. <hr/> <p>Note: The default value is <i>1</i>.</p>
PrintDistList	<p>Specify whether you want the list of Users and Roles who will be the recipient of a report generated for a job item displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • <i>1</i> to display the list. • <i>0</i> to hide the list. <hr/> <p>Note: The default value is <i>0</i>.</p>
PrintSystemMessage	<p>Specify whether you want to have the message specified from the Process Definition or Job Definition page displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • <i>1</i> to display the system message. • <i>0</i> to hide the system message. <hr/> <p>Note: The default value is <i>0</i>.</p>

Element Name	Description
PrintApplicationMessage	<p>Specify whether you want the application messages displayed in the HTML report. These are the application messages that can be viewed from the Message Log subpage of the Process Monitor Detail page. This parameter takes a string value:</p> <ul style="list-style-type: none"> • <i>1</i> to display the application message. • <i>0</i> to hide the application message. <hr/> <p>Note: The default value is 0.</p>
PrintParamList	<p>Specify whether you want the parameter list for a job item displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • <i>1</i> to display the parameter list. • <i>0</i> to hide the parameter list. <hr/> <p>Note: The default value is 0.</p>

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_PRINT_JOBREQUESTREPORT_REQ=
.VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:PrintJobRequestReport>
      <ProcessInstance>10901</ProcessInstance>
      <PrintSystemMessage>1</PrintSystemMessage>
      <PrintApplicationMessage>1</PrintApplicationMessage>
      <PrintParamList>1</PrintParamList>
    </prq:PrintJobRequestReport>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_PRINT_JOBREQUESTREPORT_RES

Element Name	Description
HTML Report	Returns HTML report.

Example Response:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200=
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:PrintJobRequestReportResponse xsi:schemaLocation="http://xmlns.oracle.co=
m/Enterprise/Tools/schemas/PRQ_PRINT_JOBREQUESTREPORT_RES.VERSION_1 PRQ_PRINT_JOBRE=
QUESTREPORT_RES.VERSION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/=
schemas/PRQ_PRINT_JOBREQUESTREPORT_RES.VERSION_1">
      <HTMLReport><![CDATA[<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transiti=
onal//EN">
```

```

<HTML><HEAD><TITLE>Job Report</TITLE>
<META content="text/html; charset=windows-1252" http-equiv=Content-Type>
<STYLE type=text/css>
.PSHEADER {BACKGROUND-COLOR: rgb(51,51,153); COLOR: rgb(255,255,255); FONT-FAMILY: =>
"Arial","sans-serif"; FONT-SIZE: 9pt; FONT-STYLE: normal; FONT-WEIGHT: bold}
.PSEVENROW {BACKGROUND-COLOR: rgb(255,255,255); COLOR: rgb(0,0,0); FONT-FAMILY: "Ar=>
ial","sans-serif"; FONT-SIZE: 9pt; FONT-STYLE: normal; FONT-WEIGHT: normal}
.PSODDROW {BACKGROUND-COLOR: rgb(226,226,226); COLOR: rgb(0,0,0); FONT-FAMILY: "Ari=>
al","sans-serif"; FONT-SIZE: 9pt; FONT-STYLE: normal; FONT-WEIGHT: normal}
</STYLE>
<META content="MSHTML 5.00.3103.1000" name=GENERATOR></HEAD>
<BODY>

<span class='PSTEXT' ><strong></strong></span><br />
<p> </p><TABLE cellpadding=2 cellspacing=0 width="100%">
  <TBODY>
    <TR>
      <TD>
        </TD>
      </TR></TBODY></TABLE>
<BR></BODY></HTML>]]</HTMLReport>
  </prq:PrintJobRequestReportResponse>
</soapenv:Body>
</soapenv:Envelope>

```

PRCS_PRINT_SCHEDULE_HTML_REPORT

This service operation retrieves an HTML formatted report displaying all items in a job or jobset as defined in the Scheduled Jobset Definition component.

Request Message: PRQ_PRINTSCHEDULEHTMLREPORT_REQ

<i>Element Name</i>	<i>Description</i>
MainSchedName	Specifies the name of the jobset schedule.
MainJobName	Specifies the name of job within the jobset schedule.
PrintJobTree	<p>Specify whether to have the job tree displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • <i>1</i> to display the tree. • <i>0</i> to hide the tree. <hr/> <p>Note: The default value is <i>1</i>.</p>
PrintDistList	<p>Specify whether you want the list of Users and Roles who will be the recipient of a report generated for a job item displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • <i>1</i> to display the list. • <i>0</i> to hide the list. <hr/> <p>Note: The default value is <i>0</i>.</p>

Element Name	Description
PrintNotifyList	<p>Specify whether you want the list of Users and Roles who will be notified for the status of a job item displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • 1 to display the list. • 0 to hide the list. <hr/> <p>Note: The default value is 0.</p>
PrintMessageList	<p>Specify whether you want the messages that will be emailed upon completion of a job item displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • 1 to display the list. • 0 to hide the list. <hr/> <p>Note: The default value is 0.</p>
PrintParamList	<p>Specify whether you want the parameter list for a job item displayed in the HTML report. This parameter takes a string value:</p> <ul style="list-style-type: none"> • 1 to display the list. • 0 to hide the list. <hr/> <p>Note: The default value is 0.</p>

Example Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:p=
rq="http://xmlns.oracle.com/Enterprise/Tools/schemas/PRQ_PRINT_SCHEDHTMLREPORT_REQ.→
VERSION_1">
  <soapenv:Header/>
  <soapenv:Body>
    <prq:PrintScheduleHTMLReport>
      <MainSchedName>sct</MainSchedName>
      <MainJobName>3SQR</MainJobName>
      <PrintJobTree>1</PrintJobTree>
      <PrintDistList>1</PrintDistList>
    </prq:PrintScheduleHTMLReport>
  </soapenv:Body>
</soapenv:Envelope>
```

Response Message: PRQ_PRINTSCHEDULEHTMLREPORT_RES

Element Name	Description
HTMLReport	Returns HTML report.

Example Response:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:s=
oapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/200
1/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <prq:PrintScheduleHTMLReportResponse xsi:schemaLocation="http://xmlns.oracle.c
om/Enterprise/Tools/schemas/PRQ_PRINT_SCHEDULEHTMLREPORT_RES.VERSION_1_PRQ_PRINT_SCHE
DHTMLREPORT_RES.VERSION_1.xsd" xmlns:prq="http://xmlns.oracle.com/Enterprise/Tools/
schemas/PRQ_PRINT_SCHEDULEHTMLREPORT_RES.VERSION_1">
      <HTMLReport><![CDATA[<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transiti
onal//EN">
<HTML><HEAD><TITLE>Job Report</TITLE>
<META content="text/html; charset=windows-1252" http-equiv=Content-Type>
<STYLE type=text/css>
.PSHEADER {BACKGROUND-COLOR: rgb(51,51,153); COLOR: rgb(255,255,255); FONT-FAMILY: =>
"Arial","sans-serif"; FONT-SIZE: 9pt; FONT-STYLE: normal; FONT-WEIGHT: bold}
.PSEVENROW {BACKGROUND-COLOR: rgb(255,255,255); COLOR: rgb(0,0,0); FONT-FAMILY: "Ar=>
rial","sans-serif"; FONT-SIZE: 9pt; FONT-STYLE: normal; FONT-WEIGHT: normal}
.PSODDROW {BACKGROUND-COLOR: rgb(226,226,226); COLOR: rgb(0,0,0); FONT-FAMILY: "Ari=>
al","sans-serif"; FONT-SIZE: 9pt; FONT-STYLE: normal; FONT-WEIGHT: normal}
</STYLE>
<META content="MSHTML 5.00.3103.1000" name=GENERATOR></HEAD>
<BODY>

<span class='PSTEXT' ><strong>sct</strong></span><br />


<span class='PSTEXT' ><strong>3SQR</strong></span><br />



<span class='PSTEXT' >1 XRFIELDS: Cross Reference Field Listing (SQR Report)</spa=>
n><br />



<span class='PSTEXT' >2 XRFMENU: Menu Listing Report (SQR Report)</span><br />



<span class='PSTEXT' >3 XRFRCFL: Cross Reference - Records and (SQR Report)</span>=>
<br />
<p> </p><TABLE cellpadding=2 cellspacing=0 width="100%">
  <TBODY>
    <TR>
      <TD>
        <TABLE border=1 cellpadding=3 cellspacing=0 width="100%">
          <TBODY>
            <tr>
              <th class='PSHEADER' colspan="10" bordercolor=rgb(51,51,153)><p align="left">=>
Job Name: 3SQR - SQR Multi-process Job</p></th>
            </tr>
            <tr>
              <th class='PSHEADER' colspan="3" bordercolor=rgb(51,51,153)><p align="left">M=>
ode: Serial</p></th>
              <th class='PSHEADER' colspan="7" bordercolor=rgb(51,51,153)> </th>
            </tr>
            <tr>
              <th class='PSHEADER' width="5%">Seq.</th>
              <th class='PSHEADER' width="10%">Process Name</th>
              <th class='PSHEADER' width="15%">Description</th>
              <th class='PSHEADER' width="10%">Process Type</th>
              <th class='PSHEADER' width="10%">Run Control ID</th>
              <th class='PSHEADER' width="5%">Type</th>
              <th class='PSHEADER' width="8%">Output
Format</th>
              <th class='PSHEADER' width="10%">Destination</th>

```

```

    <th class='PSHEADER' width="8%">Server Option</th>
    <th class='PSHEADER' width="5%">Run Time</th>
</tr>
<tr>
    <td class='PSEVENROW' align="center" width="5%">1</td>
    <td class='PSEVENROW' width="10%">XRFIELDS</td>
    <td class='PSEVENROW' width="15%">Cross Reference Field Listing</td>
    <td class='PSEVENROW' align="center" width="10%">SQR Report</td>
    <td class='PSEVENROW' align="center" width="5%">t1</td>
    <td class='PSEVENROW' align="center" width="5%">Web</td>
    <td class='PSEVENROW' align="center" width="8%">Acrobat (*.pdf)</td>
    <td class='PSEVENROW' align="center" width="10%">Distribution List</td>
    <td class='PSEVENROW' align="center" width="8%">Any Server </td>
    <td class='PSEVENROW' align="center" width="5%"> </td>
</tr>
<tr>
    <td class='PSEVENROW' colspan="2"> </td>
    <td class='PSEVENROW' colspan="8">
        <p><strong>Distribution List:</strong></p>
        <table border="1" cellpadding="3" cellspacing="0" bordercolor="#CCCCCC"> =>
            <tr>
                <td class='PSEVENROW' ><strong>Type</strong></td>
                <td class='PSEVENROW' ><strong>Name</strong></td>
            </tr>
            <tr>
                <td class='PSEVENROW' >User </td>
                <td class='PSEVENROW' >VP1 </td>
            </tr>
        </table>
    </td></tr>
<tr>
    <td class='PSODDROW' align="center" width="5%">2</td>
    <td class='PSODDROW' width="10%">XRFMENU</td>
    <td class='PSODDROW' width="15%">Menu Listing Report</td>
    <td class='PSODDROW' align="center" width="10%">SQR Report</td>
    <td class='PSODDROW' align="center" width="5%">t1</td>
    <td class='PSODDROW' align="center" width="5%">Web</td>
    <td class='PSODDROW' align="center" width="8%">Acrobat (*.pdf)</td>
    <td class='PSODDROW' align="center" width="10%">Distribution List</td>
    <td class='PSODDROW' align="center" width="8%">Any Server </td>
    <td class='PSODDROW' align="center" width="5%"> </td>
</tr>
<tr>
    <td class='PSODDROW' colspan="2"> </td>
    <td class='PSODDROW' colspan="8">
        <p><strong>Distriution List:</strong></p>
        <table border="1" cellpadding="3" cellspacing="0" bordercolor="#CCCCCC"> =>
            <tr>
                <td class='PSODDROW' ><strong>Type</strong></td>
                <td class='PSODDROW' ><strong>Name</strong></td>
            </tr>
            <tr>
                <td class='PSODDROW' >User </td>
                <td class='PSODDROW' >VP1 </td>
            </tr>
        </table>
    </td></tr>
<tr>
    <td class='PSEVENROW' align="center" width="5%">3</td>
    <td class='PSEVENROW' width="10%">XRFRCFL</td>
    <td class='PSEVENROW' width="15%">Cross Reference - Records and</td>
    <td class='PSEVENROW' align="center" width="10%">SQR Report</td>
    <td class='PSEVENROW' align="center" width="5%">t1</td>
    <td class='PSEVENROW' align="center" width="5%">Web</td>
    <td class='PSEVENROW' align="center" width="8%">Acrobat (*.pdf)</td>
    <td class='PSEVENROW' align="center" width="10%">Distribution List</td>
    <td class='PSEVENROW' align="center" width="8%">Any Server </td>
    <td class='PSEVENROW' align="center" width="5%"> </td>
</tr>
<tr>
    <td class='PSEVENROW' colspan="2"> </td>

```

```

    <td class='PSEVENROW' colspan="8">
    <p><strong>Distribution List:</strong></p>
    <table border="1" cellpadding="3" cellspacing="0" bordercolor="#CCCCCC"> =>
    <tr>
    <td class='PSEVENROW' ><strong>Type</strong></td>
    <td class='PSEVENROW' ><strong>Name</strong></td>
    </tr>
    <tr>
    <td class='PSEVENROW' >User      </td>
    <td class='PSEVENROW' >VP1      </td>
    </tr>
    </table>
    </td></tr>
</TBODY></TABLE>
<p> </p>      </TD></TR></TBODY></TABLE>
<BR></BODY></HTML>]]</HTMLReport>
    </prq:PrintScheduleHTMLReportResponse>
  </soapenv:Body>
</soapenv:Envelope>

```

