PeopleTools 8.59: Feed Publishing Framework

April 2021

PeopleTools 8.59: Feed Publishing Framework

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 that have purchased support 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

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.

Hosted PeopleSoft Online Help

You can access the hosted PeopleSoft Online Help on the Oracle Help Center. The hosted PeopleSoft Online Help 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. The hosted PeopleSoft Online Help is available in English only.

To configure the context-sensitive help for your PeopleSoft applications to use the Oracle Help Center, see Configuring Context-Sensitive Help Using the Hosted Online Help Website.

Locally Installed Help

If you're setting up an on-premise PeopleSoft environment, and your organization has firewall restrictions that prevent you from using the hosted PeopleSoft Online Help, you can install the online help locally. See Configuring Context-Sensitive Help Using a Locally Installed Online Help Website.

Downloadable PeopleBook PDF Files

You can access downloadable PDF versions of the help content in the traditional PeopleBook format on the Oracle Help Center. 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 families 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 family. Whether you are implementing a single application, some combination of applications within the product family, or the entire product family, 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: 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.

Typographical Convention	***Description***
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)

Translations and Embedded Help

PeopleSoft 9.2 software applications include translated embedded help. With the 9.2 release, PeopleSoft aligns with the other Oracle applications by focusing our translation efforts on embedded help. We are not planning to translate our traditional online help and PeopleBooks documentation. Instead we offer very direct translated help at crucial spots within our application through our embedded help widgets. Additionally, we have a one-to-one mapping of application and help translations, meaning that the software and embedded help translation footprint is identical—something we were never able to accomplish in the past.

Using and Managing the PeopleSoft Online Help

Select About This Help in the left navigation panel on any page in the PeopleSoft Online Help to see information on the following topics:

- Using the PeopleSoft Online Help
- Managing Hosted online help
- Managing locally installed PeopleSoft Online Help

Understanding PeopleTools Modules

This PeopleBook refers to the following modules:

- Feed Publishing Framework
- Integration Broker
- PeopleSoft Query
- Workflow
- Process Scheduler

Feed Publishing Framework

The Feed Publishing Framework is a module of Oracle's PeopleTools. It consists of unified interfaces for content owners and administrative users to create, configure, and maintain feed definitions of various types; interfaces for end users to discover related feeds and search feed definitions; a set of application programming interfaces (APIs) and code samples to assist application developers in creating new type of feeds and integrating them with existing features; and multiple language support.

Integration Broker

Integration Broker facilitates exposing PeopleSoft business logic as services and consuming external web services by Oracle's PeopleSoft applications. Integration Broker also supports synchronous and asynchronous messaging among PeopleSoft applications and with third-party systems. Integration Broker uses a variety of communication protocols while managing message structure, message content, and transport disparities.

PeopleSoft Query

Oracle's PeopleSoft Query is an end user reporting tool. With Query Manager, you can extract the precise information that you are looking for by using visual representations of your PeopleSoft database, without writing SQL statements. The queries that you write can be as simple or as complex as necessary. They can be one-time queries or queries that you use repeatedly.

Workflow

Workflow enables you to efficiently automate the flow of information throughout your enterprise, crossing both application and functional boundaries. PeopleSoft Workflow technology consists of a powerful set of tools that enables you to automate time-consuming business processes. You can merge the activities of multiple users into flexible business processes to increase efficiency, cut costs, and keep up with rapidly changing customer and competitive challenges.

Process Scheduler

Process Scheduler is a centralized tool that enables application developers, system administrators, and application users to manage PeopleSoft batch processes. Using the PeopleSoft Pure Internet Architecture, you can access a list of processes through a web browser, and queue and run a process request.

PeopleTools Related Links

PeopleTools 8.59 Home Page

PeopleTools Elasticsearch Home Page

"PeopleTools Product/Feature PeopleBook Index" (PeopleTools 8.59: Getting Started with PeopleTools)

PeopleSoft Hosted Online Help

PeopleSoft Information Portal

PeopleSoft Spotlight Series

PeopleSoft Training and Certification | Oracle University

My Oracle Support

Oracle Help Center

Contact Us

Send your suggestions to psoft-infodev_us@oracle.com. Please include the applications update image or PeopleTools release that you're using.

Follow Us

YouTube

Twitter@PeopleSoft_Info.

PeopleSoft Blogs

LinkedIn

Chapter 1

Getting Started With the Feed Publishing Framework

Feed Publishing Framework Overview

The Feed Publishing Framework provides user interfaces and application programming interfaces (APIs) for feed definition creation and maintenance, feed searching and subscription, as well as feed document generation and delivery.

Feeds

A web feed, news feed, or feed is a data format that makes rapidly changing content available to users. A feed contains entries, which might be headlines, full-text articles, excerpts, summaries, digital media, and links to content on a web site, along with various metadata. PeopleSoft applications, like news agencies, external web sites, and other content distributors, publish or syndicate web feeds to which users can subscribe. Examples of feeds in PeopleSoft applications include worklists, queries, discussions, and so on.

Web feeds work by using the pull technology model. Typically, when using web feeds, content providers publish feed links on their site, which you register in an aggregator program (also called a feed reader or a news reader) that runs on your own computer. Feed readers can be independent programs or browser extensions that provide configurable integrated feed reader functionality. Web-based feed readers seldom require additional software installation and make the user's feeds available on any computer with internet access. Microsoft Outlook and Mozilla Thunderbird are two commonly used secure feed readers.

When directed, the aggregator polls all the servers in its feed list to ascertain if new content exists; if so, the aggregator retrieves the new content. You can configure feed readers to check for new or updated content at regular intervals. When you no longer want to receive updated content, you remove the feed from the reader.

Atom and Really Simple Syndication (RSS) are the two primary syndication formats. Atom also provides a standardized way to export an entire blog, or parts of it, for backup or for importing into other blogging systems. Some web sites let people choose between RSS- or Atom-formatted web feeds; others offer only RSS or only Atom. In particular, many blog and wiki sites offer their web feeds in the Atom format.

Most web pages, including PeopleSoft application pages, use this feed icon to indicate that a feed is available in either the RSS 1.0, RSS 2.0, or Atom 1.0 format.

Note: PeopleSoft applications offer feeds in Atom 1.0 format only.

Feed Publishing Framework

The Feed Publishing Framework provides:

- Feed Publishing Wizard, a simplified guide for content owners and administrative users to create, configure, and maintain feed definitions and feed templates of various types.
- Publish as Feed hidden pages to allow for feed publication directly from the data component.
- Interfaces for end users to search for feeds and to discover related feeds.
- Management of feed definitions, feed categories, feed templates and feed data types as PeopleTools managed objects.
- A set of APIs and code samples to assist application developers in creating new types of feeds and integrating them with existing features.

With the Feed Publishing Framework, you can develop and publish feeds from any PeopleSoft or non-PeopleSoft data. Then, with the delivered user interfaces, users can find and subscribe to your feeds.

Feed Publishing Framework Implementation

This section provides an overview of feed publishing framework implementation.

Prerequisites

The Feed Publishing Framework relies on other PeopleTools components, such as Integration Broker, for you to be able to publish and consume feeds.

You must configure your PeopleSoft system as follows:

Step	***Reference***
Configure and activate the Integration Broker gateway and target nodes.	See Configuring the Integration Broker Gateway and Target Nodes.
Configure the Integration Broker service target locations.	See Configuring Integration Broker Service Target Locations. **Note:** If you plan to use secure authentication, then you must also configure Secure Sockets Layer (SSL) on your system. See "Implementing WebLogic SSL Keys and Certificates" (PeopleTools 8.59: System and Server Administration).
The default user of the ANONYMOUS node must be a valid user.	See Configuring the Default User of the ANONYMOUS Node.
Set authentication for the default local node to password or certificate. In addition, define the content URI text and portal URI text of the default local node.	See Configuring the Default Local Node.
Define the content URI text and portal URI text of each local host node.	See Setting URI Text for Local Host Nodes.

Step	***Reference***
For Oracle WebLogic Server, you must configure the server to disable its own authentication.	**Note:** By default, the delivered config.xml file is set to disable Oracle WebLogic Server's own authentication. No additional configuration is required unless you have changed this authentication setting. Disabling Oracle WebLogic Server's authentication allows authentication to be passed through and handled by the PeopleSoft servlet. See Configuring Oracle WebLogic Server to Disable Authentication.
Assign users the roles and permission necessary to use feeds.	Feed administrators require the PTPT1300 - Portal Administrator permission list, which is included in the Portal Administrator role. Users who need to search for and view feeds require the PTPT1000 - PeopleSoft User permission list, which is included in the PeopleSoft User role. See "Setting Page Permissions" (PeopleTools 8.59: Security Administration). See "Assigning Permissions to Roles" (PeopleTools 8.59: Security Administration). See "Setting Roles" (PeopleTools 8.59: Security Administration).
Using psadmin, activate the Pub/Sub servers on the application server.	This step is required only when scheduled feed messages are published to Integration Broker queues. See "Accessing the Application Server Options" (PeopleTools 8.59: System and Server Administration).
Configure Process Scheduler.	This step is required only when scheduled feed messages are published to Integration Broker queues through an Application Engine program. See "Using the Process Scheduler Menu" (PeopleTools 8.59: System and Server Administration).

Using PeopleTools-Delivered Feed Data Types

PeopleTools delivers predefined feed data types for query feeds, worklist feeds, Integration Broker generic message feeds, and lists of published feeds. Follow these steps to use an existing feed data type:

Step	***Reference***
Publish a *list of feeds* feed for each feed data type using the Define Feed Data Types page. Also publish a list of feeds feed for the FEED data type to create a *master list* of feed lists.	See Publishing a List of Feeds Feed.
Create query feeds.	Create query feeds from Query Manager pages. See Understanding Query Feeds.

Step	***Reference***
Create worklist feeds.	Create worklist feeds from the Worklist and Worklist Details pages. See Understanding Worklist Feeds.
Create feeds for Integration Broker asynchronous, one-way service operations, also known as Integration Broker generic message feeds.	Create Integration Broker generic message feeds from the Define IB Generic Message Feed page. See Understanding Integration Broker Generic Message Feeds.

Developing New Feed Data Types

Use these steps to develop new feed data types:

Step	***Reference***
Analyze requirements for the new feed data type.	See Analyzing Requirements for New Feed Data Types.
Implement a feed data source application class.	See Creating the Feed Data Source Application Class.
Define the feed data type.	See Creating the Feed Data Type Definition.
(Optional) Update the property maintenance page to include a Publish as Feed link.	See Adding the Publish as Feed Link to a Page in the Component. **Note:** Because any feed definition can be published through the Feed Publishing Wizard, it is optional to update the property maintenance page to add the Publish as Feed link.
(Optional) Add standard Publish as Feed pages to a component as hidden pages.	See Adding the Four Standard Publish as Feed Pages. **Note:** Because any feed definition can be published through the Feed Publishing Wizard, it is optional to update the property maintenance component to add the Publish as Feed pages.
Update the view content page to include the related feeds hover menu.	See Adding the Related Feeds Hover Menu to Pages.

Chapter 2

Understanding the Feed Publishing Framework

Feed Publishing Framework

The Feed Publishing Framework provides:

- Feed Publishing Wizard, a simplified guide for content owners and administrative users to create, configure, and maintain feed definitions and feed templates of various types.
- Publish as Feed hidden pages to allow for feed publication directly from the data component.
- Interfaces for end users to search for feeds and to discover related feeds.
- Management of feed definitions, feed categories, feed templates and feed data types as PeopleTools managed objects.
- A set of APIs and code samples to assist application developers in creating new types of feeds and integrating them with existing features.

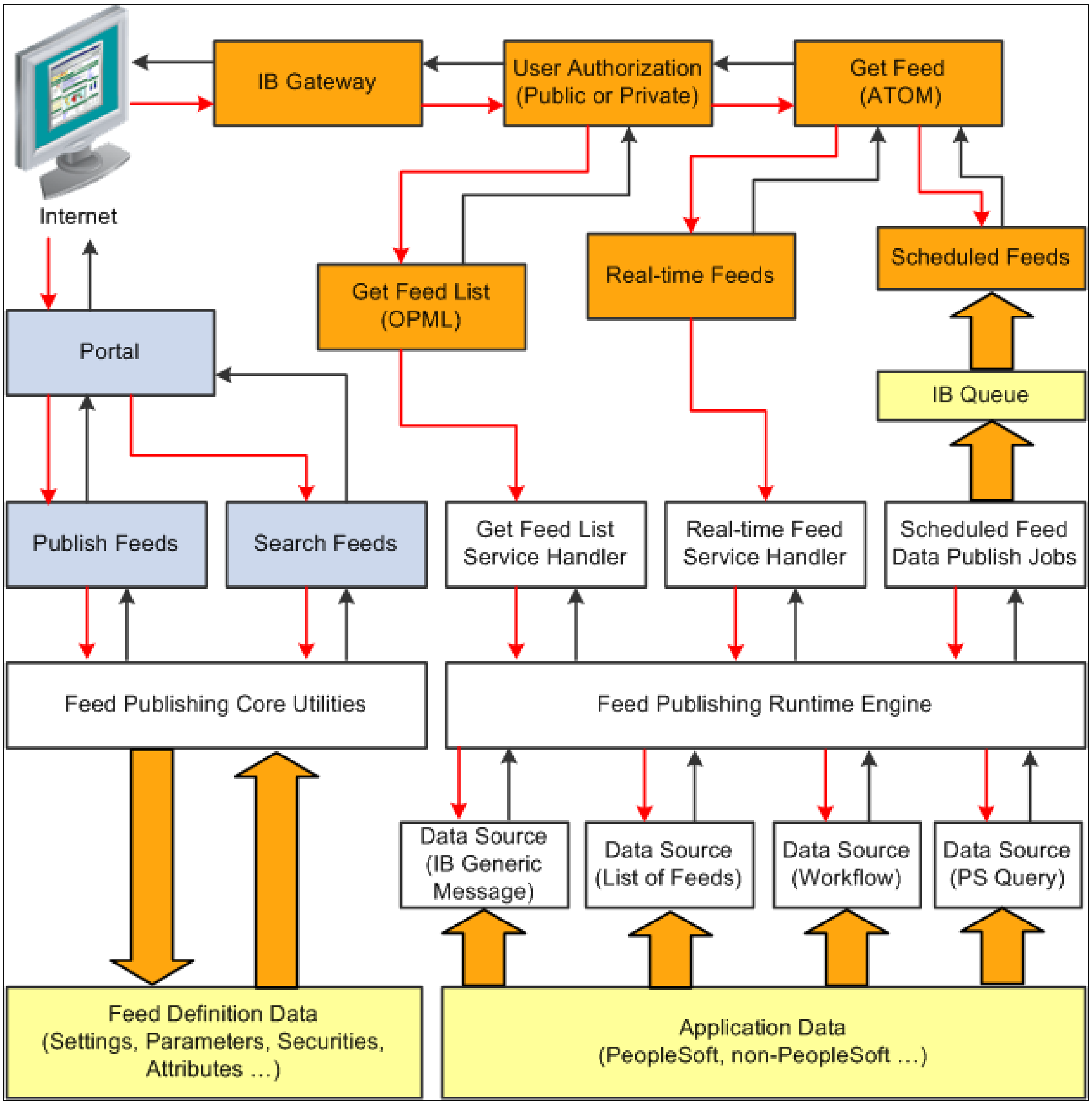
  Multiple language support.

The following diagram shows the Feed Publishing Framework architecture. For this diagram, use this legend:

- Orange boxes - Integration Broker components
- Light blue boxes - Portal components or GUI
- Yellow boxes - Data storage
- White boxes - Feed Publishing Framework core
- Red lines - Incoming request
- Black lines - Response
- Gold arrows - Data to and from database

Image: Feed Publishing Framework architecture

Feed Publishing Framework architecture


Internet
IB Gateway
User Authorization (Public or Private)
Get Feed (ATOM)
Portal
Get Feed List (OPML)
Real-time Feeds
Scheduled Feeds
IB Queue
Publish Feeds
Search Feeds
Get Feed List Service Handler
Real-time Feed Service Handler
Scheduled Feed Data Publish Jobs
Feed Publishing Core Utilities
Feed Publishing Runtime Engine
Data Source (IB Generic Message)
Data Source (List of Feeds)
Data Source (Workflow)
Data Source (PS Query)
Feed Definition Data (Settings, Parameters, Securities, Attributes ...)
Application Data (PeopleSoft, non-PeopleSoft ...)

This section discusses:

- Feed publishing runtime engine.
- Feed document generation and delivery.
- Feed document properties.
- Feed data type application classes.
- Creation of new feed data types.

Feed Publishing Runtime Engine

The centerpiece of the feed document generation is the runtime engine. The runtime engine:

- Handles the feed identification and authorization.
- Sets the data source settings and data source parameter value.
- Executes the data source object to collect data.
- Transforms feed data to an Atom 1.0 feed document.

Runtime Engine Process Flow

When the runtime engine receives a request, it first locates the feed definition based on the feed ID, and then it determines whether the current user has access to the feed. If the user has permission, the process continues. If the user does not have permission, the process stops and the system displays an error.

Next, the runtime engine creates an instance of the feed data type application class associated with the specific feed definition, restores the Data Source Setting values and the Data Source Parameter values saved with the feed definition, and sets the runtime information, such as the requested language. At this time, the runtime engine also evaluates the runtime values of data source parameters with usage types of *Admin Specified, System Variable,* and *User Specified.*

Then, the runtime engine instructs the feed object to collect the data and return the feed data in an Atom 1.0 feed document. The runtime engine does not have direct knowledge about the feed data. All business logic relevant to data collection are encapsulated in the feed data type application class PeopleCode.

By default, the currently authenticated user of the request is used for gathering the feed data. Different users requesting the same feed may receive different feed data based on their permission. This behavior can be overridden by specifying the Feed Authorization options when you create a feed definition; that is, a user ID can be specified alone with a selection that specifies whether to use this user to generate feed documents for all requests of that feed or only for anonymous requests. This option is a per feed definition, and it will be stored with the feed definition.

Feed Document Generation and Delivery

By default, all feeds managed by the framework will be served as real-time feeds through Integration Broker framework by using the GetFeed service operation.

GetFeed Service Operation

A GetFeed service operation (PTFP_GETFEED) uses the runtime engine to locate the feed definition and generate feed document in Atom 1.0 format. Transformation can be done in the service operation level to get feed documents in other formats. Integration Broker manages user authentication, caching, and feed data type service operation-level security.

The primary task of the GetFeed service operation handler is to collect the request information and then forward it to the runtime engine. The handler also catches the exceptions raised by the runtime engine and reports them in the specific way applicable to Integration Broker.

By default, the GetFeed service operation only has one required parameter, the feed ID, and three optional parameters, the language code, the portal name, and the node name. The feed URL given out by the framework contains only these parameters, when applicable.

When a feed request comes in, the Integration Broker copies the values of the query parameters to the corresponding fields defined in the default feed request message definition. Then, the service operation passes this information to the runtime engine for further processing. If other query parameters exist in the request URL, then the service operation collects and passes them to the runtime engine. These additional query parameter values override the values of Data Source Parameters for which the usage types are *User Specified*.

Any feed data type can override this default behavior by providing its own feed request message definition extended from the default one, as well as by using a different method for generating the feed URL to include more default query parameters. Regardless, the feed ID query parameter should always be a required parameter. We do not recommend the use of query parameters for data source settings or non-user-specified data source parameters.

Feed Document Properties

The runtime engine returns an Atom 1.0 feed document. The properties of the feed definition become the feed header properties of the feed document. Each item in the feed-format-neutral object collection becomes one entry of the feed document. A list of the data mapping relationships follows.

Feed Element Mapping

The feed data source uses a feed-format-neutral object collection to collect data. This table describes the mapping between the Atom 1.0 feed-level elements and the feed definition properties:

Atom 1.0 Feed-Level Element	***Feed Definition Property***
author	Author.
category	Feed data type and feed category.
contributor	Contributor.
generator	Latest entry in PSRELEASE table.
icon	Icon.
id	System-generated URL to open the feed.
link	Alternate link: The content web page URL associated with the feed, which is provided by the data source. Self link: System-generated URL to open the feed.
logo	Logo.
rights	Copyright.
subtitle	Feed description.
title	Feed title.
update	Current date and time of request.

This table describes the mapping between the Atom 1.0 entry-level elements and properties of items in the feed-format-neutral object collection. The real values of these properties are determined by the data source at runtime:

Atom 1.0 Entry-Level Element	***Item Property***
author	Author.
category	Category. **Note:** This item allows multiple entries.
content	FullContent, if available.
contributor	Contributor. **Note:** This item allows multiple entries.
id	GUID, if available, or content URL.
link	Alternate link: contentURL. Other links: enclosure, if available. **Note:** This item allows multiple entries.
published	Date and time published.
rights	Copyright.
source	NA
summary	Description.
title	Title.
updated	Date and time updated.

Note: If any property contains empty values, the corresponding element is not added.

Feed Data Type Application Classes

The feed data type application class (PTFP_FEED:DataSource:DataSource) bridges the data and the feed definition. It has two roles:

- At design time, it provides information to the framework about how to define the feed definition for this type of data.
- At runtime, the framework uses the application class to collect feed data.

Design Time Role of Application Classes

At design time, the feed data type application class provides information about a specific type of data to the framework, including a list of data source settings, list of data source parameters and their default values, data security, and name of the default feed service definition. The feed data type application class handles events such as processes that occur when you delete a feed definition. The framework uses all of this information in the feed definition creation and maintenance.

Every feed data type may have zero to three data source settings. The data source settings uniquely define the feed data source of the given type of data. You must define and store the values of the data source settings with the feed definition. The feed data type application class can also provide the prompt information for each data source setting.

Every feed data type may have zero or more data source parameters. The data source parameters are used to fine tune the feed or personalize the feed; for example, a news publication feed should include all child sections or a workspace feed should not include discussion data. Values of data source parameters are determined at runtime based on the usage type, fixed value, system variable value, user-specified value, and so on. The default values of data source parameters are stored with the feed definition. The Feed Data Type application class could provide the description, prompt information, default values, and default usage type for each data source parameter.

Runtime Role of Application Classes

At runtime, the runtime engine finds the feed definition and the associated feed data type application class based on the requested feed ID. It creates an instance of the feed data type application class associated with the specific feed definition, restores the data source setting values and the data source parameter values saved with the feed definition, evaluates the runtime values of those data source parameters based on their usage type, and sets other runtime information. It then executes the object to retrieve the content data of the feed.

The feed data type application class uses a feed-format-neutral object collection to temporarily store the data. It does not transform data to a feed document directly. This design enables application developers to expand the object model or use their own data objects. The advantages of using a feed-format-neutral object collection instead of the feed-format-specific XML document are:

- It shields you from having to deal with the complex details of a specific feed format.
- It minimizes the possibility of using a wrong XML element or structure.
- It standardizes the use of specific elements within the feed.
- It enables easier migration to newer feed format standards or a completely different feed format.

Related Links

Creating the Feed Data Source Application Class

"DataSource Class" (PeopleTools 8.59: PeopleCode API Reference)

Creation of New Feed Data Types

The following list provides an overview of the steps to create a new feed data type:

1. Analyze the requirements for the new feed data type.
2. Create the feed data source application class by extending the abstract base DataSource class.

3. Create the feed data type definition.
4. (Optional) Update the property maintenance component.

   Note: Because any feed definition can be published through the Feed Publishing Wizard, it is optional to update the property maintenance component to add the Publish as Feed pages and Publish as Feed link to a component.

5. Update the view content component or pagelet — that is, deploy the related feeds hover menu.

These steps are described in detail elsewhere in this book.

Related Links

Analyzing Requirements for New Feed Data Types

Delivered Feed Data Types

This section discusses the delivered feed data types.

List of Feeds (FEED)

A *list of feeds* feed enables feed administrators to generate a feed that displays a list of all feeds of a specific feed data type.

See Publishing a List of Feeds Feed.

Integration Broker Generic Message Feeds (GENERICFEED)

Integration Broker generic message feeds enable administrators to expose Integration Broker messages used in asynchronous, one-way service operations as feeds.

See Understanding Integration Broker Generic Message Feeds.

PeopleSoft Query Feeds (PSQUERY)

PeopleSoft Query feeds enable query administrators to expose query outputs as feeds.

Note: Any user with access to Query Manager can publish query feeds.

See Understanding Query Feeds.

Worklist Feeds (WORKLIST)

Worklist feeds enable workflow administrators to expose worklists as feeds.

See Understanding Worklist Feeds.

Feed Types and Options

This section provides an overview of feed types and options.

Real-Time Feeds

Real-time feeds are dynamic—that is, they are produced when the user requests them. Real-time feeds are created using Integration Broker synchronous service operations. These service operations are similar to other Integration Broker service operations except that the service operation handler returns an ATOM_1_0 message.

Whenever the HTTP listening connector gets a request for a real-time feed, it invokes the appropriate synchronous service operation. It uses either the PS_TOKEN or basic authentication credentials. User authentication and service operation authorization are handled by Integration Broker; feed authorization is handled by the Feed Publishing Framework. If a user has access to the feed, then the service operation handler adjudicates any HTTP request parameters passed to it, generates an Atom feed, and returns it in an ATOM_1_0 response message.

Scheduled Feeds

Scheduled feeds are published asynchronously and stored as messages in Integration Broker queues.

Scheduled feeds can be further classified into up-front feeds and generic feeds. When using up-front feeds, the messages published to the Integration Broker queues are feed messages. When using generic feeds, the messages published to Integration Broker queues are either PeopleSoft rowset or XML messages. They are not feed (Atom) XML messages.

When a user requests the feed, the GetFeed (PTFP_GETPREPUBFEED) synchronous service operation is invoked by the HTTP listening connector. The GetFeed service operation handler fetches the appropriate feed messages from Integration Broker queues and collates them into a single feed message. The GetFeed service operation handler collates the messages for up-front feeds into a single feed message; for generic feeds, it wraps the feed element tags to the Integration Broker messages and then collates them into a single feed message.

Note: Unlike real-time feeds, for which you can create your own service operation to deliver the feed, scheduled feeds always use the PTFP_GETPREPUBFEED service operation to deliver feeds.

Paged Feeds

A *paged feed* is a feed that has been split into pages (also known as segments) to improve system performance in delivering large feed documents and to improve performance for consuming a feed. A paged feed is presented with first, last, next, and previous links to allow access to additional pages in the feed document.

Paged feeds are supported for scheduled feeds only. The framework supports paged feeds via Integration Broker message segments. %MaxMessageSize is recommended when creating Integration Broker message segments for paged feeds.

When setting the paging property for a feed, select either *Segmented* or *No Paging* to determine how the framework displays the complete feed:

- Segmented
  - This option is designed for feeds intended for crawlers or system synchronization.
  - This option is not supported by most feed readers or clients.
  - Feed links (first, next, previous, last) are added to the feed XML.
  - Feed entries are not restricted by the Max Rows Limit parameter.

    See Setting Feed Publishing Framework Options.
- No paging
  - This option is designed for feeds intended for end user viewing.
  - This option is supported by all feed readers or clients.
  - Feed entries are restricted by the Max Rows Limit parameter.

The following table describes how paged feed options and Integration Broker message segment options affect the output of the framework:

Integration Broker Message Option	***Feed Option - Segmented***	***Feed Option - No Paging***
Segmented	• Each message segment becomes a page in the feed XML. • Oldest message contents appear first. • The Max Rows Limit is not applicable.	• All message or message segments appear in a single feed XML. • Latest message contents appear first. • The Max Rows Limit applies.
Non-segmented	• Each message becomes a page in the feed XML. • Oldest message contents appear first. • The Max Rows Limit is not applicable.	• All message content entries appear in a single feed XML. • Latest message contents appear first. • The Max Rows Limit applies.

Related Links

Publishing Integration Broker Generic Message Feeds
"DSPARAMETER_SF_PAGING" (PeopleTools 8.59: PeopleCode API Reference)
"SF_PAGINGOPTION_NOPAGING" (PeopleTools 8.59: PeopleCode API Reference)
"SF_PAGINGOPTION_SEGMENTED" (PeopleTools 8.59: PeopleCode API Reference)

Incremental Feeds

An *incremental feed* is a feed that has been published and updated with time stamps that allow the feed content to be delivered incrementally. An incremental feed allows the Feed Publishing Framework to deliver only the feed content that has changed since the user last requested the feed.

Note: The incremental feed option is incompatible with the paged feed option. For the delivered feed data types that can be specified as paged, the *Incremental* option is disabled when *Segmented* is selected.

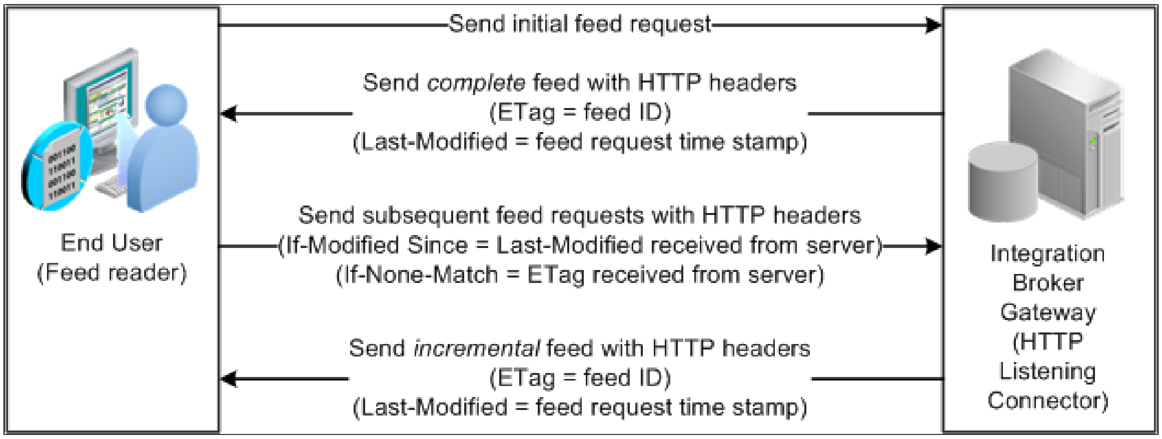
Incremental feeds save network bandwidth by using HTTP conditional GET headers. The HTTP conditional GET headers supported are:

- ETag
- If-None-Match
- Last-Modified
- If-Modified-Since

Note: Not all feed readers support incremental feeds, which requires that the reader retain the feed request time stamp and present that data as part of the next feed request.

Image: Incremental feeds

The following diagram illustrates how HTTP conditional headers are used with incremental feeds. With the initial feed request, the feed reader does not include any HTTP conditional headers. In the response, the PeopleSoft system sends the *complete* feed data and includes two HTTP conditional headers: ETag equals the feed ID and Last-Modified equals the feed request time stamp. When the feed reader makes a subsequent feed request, it includes two HTTP conditional headers: If-None-Match equals the ETag sent by the PeopleSoft system and If-Modified-Since equals Last-Modified sent by the system. In the response, the PeopleSoft system sends just the *incremental* feed data and includes the same two HTTP conditional headers as the initial response; Last-Modified now represents the latest feed request time stamp.


Send initial feed request
Send *complete* feed with HTTP headers
(ETag = feed ID)
(Last-Modified = feed request time stamp)
Send subsequent feed requests with HTTP headers
(If-Modified Since = Last-Modified received from server)
(If-None-Match = ETag received from server)
Send *incremental* feed with HTTP headers
(ETag = feed ID)
(Last-Modified = feed request time stamp)
End User
(Feed reader)
Integration
Broker
Gateway
(HTTP
Listening
Connector)

As delivered, the Feed Publishing Framework provides support for scheduled, incremental feeds. For the feed data source, you must define the DSPARAMETER_INCREMENTAL data source parameter and assign an appropriate value to it. For real-time, incremental feeds, you must also define the DSPARAMETER_INCREMENTAL data source parameter. In addition, you must implement the PeopleCode to deliver an incremental feed in the data source's execute method.

Related Links

Publishing Integration Broker Generic Message Feeds
"DSPARAMETER_INCREMENTAL" (PeopleTools 8.59: PeopleCode API Reference)
"INCREMENTALOPTION_NO" (PeopleTools 8.59: PeopleCode API Reference)
"INCREMENTALOPTION_YES" (PeopleTools 8.59: PeopleCode API Reference)

Feed Security

This section provides an overview of feed security.

Security for Publishing Feeds

Feeds can be published by two different methodologies: the Feed Publishing Wizard and Publish as Feed pages. Security is different for each of these methodologies:

- Users with access to the PTPT1300 (Portal Administrators) permission list have access to the Feed Publishing Wizard by default.
- The hidden Publish as Feed pages are accessed through the Publish as Feed link found on the property maintenance component for each type of PeopleSoft data that can be published as a feed. Therefore, access to publishing feeds in this manner depends on authorized access to the components and pages where the link is found.

Security for Viewing Feeds

The two levels of feed security are:

- Feed-level security.
- Data-level security.

Feed-Level Security

The Feed Publishing Framework manages feed-level security. Feed-level security determines which feeds are visible to the user when accessing the My Feeds page or any related hover menus. You can configure feed security to be:

- Public.

  Public feeds run under context of the default user that is associated with the ANONYMOUS node.

  See "Defining Node Parameters" (PeopleTools 8.59: Integration Broker Administration).

- Realtime.

  Every time a user accesses a feed during search or execution, the data source object determines whether the current user has access to the feed. This security option has an advantage in that the feed security is always in sync with the data source. This security option can greatly affect performance of feed searches and should only be used when the data security is constantly changing, or the data security could not be defined using role or permission list based security model.

- Permission list and roles.

  You assign access to the feed based on permission lists and roles.

Data-Level Security

By default, the authenticated user who requests a feed is used for generating the feed data. Data-level security is checked by each application class that supports a feed data type when the runtime engine executes it to collect feed data. It is always checked in real time. Users who have access to a feed but not the data will receive a feed document that contains no entries. Different users who subscribe to the same feed might receive different feed data, depending on their permissions. You can sync the feed data security to the feed definition using the Publish Feed Definition pages.

Important! Developers are responsible for building data-level security into the data source application class logic; data-level security is not automatic.

In the Feed Publishing Wizard, you can override the default feed authorization by specifying a user ID and password to be used for requests for this feed. Doing so ensures every user who requests the feed sees the same result. This setting can be useful for public feeds.

Warning! Use care when selecting a user ID for this override as this can potentially give unauthorized users access to feed data that they would not normally be able to see.

See Step 4: Specifying Publishing Options.

Security for Creating Feed Data Types

Security for creating new feed data types is based on permission lists. To create new feed data types, the user must be authorized to access pages in the PTFP_DATATYPE component on the PTFP_FEED_PUBLISHING menu.

Note: Users with access to the PTPT1300 (Portal Administrators) permission list have access to these pages by default.

See "Understanding Permission Lists" (PeopleTools 8.59: Security Administration).

Feed Publication

The Feed Publishing Framework provides two methods for publishing feeds:

- Feed Publishing Wizard.
- Publish as Feed links and pages.

In a PeopleSoft system, administrative users and content owners with appropriate authorization can use either method to create, view, edit, and delete feed definitions related to a particular data source.

See Methods for Publishing Feeds.

My Feeds

End users can search and view feeds by using the My Feeds link, which can be found in the menu navigation. Search for feeds specific to the user and then click the feed document link to view it in a new browser window. You can also add the feed URL to feed readers, or you can export the search results to an OPML (Outline Processor Markup Language) file and save the list for later use.

See Using the My Feeds Page.

Chapter 3

Configuring Your PeopleSoft System to Support Feeds

Configuring the Integration Broker Gateway and Target Nodes

To use the Feed Publishing Framework, you must first configure your PeopleSoft system to support feed publication and consumption.

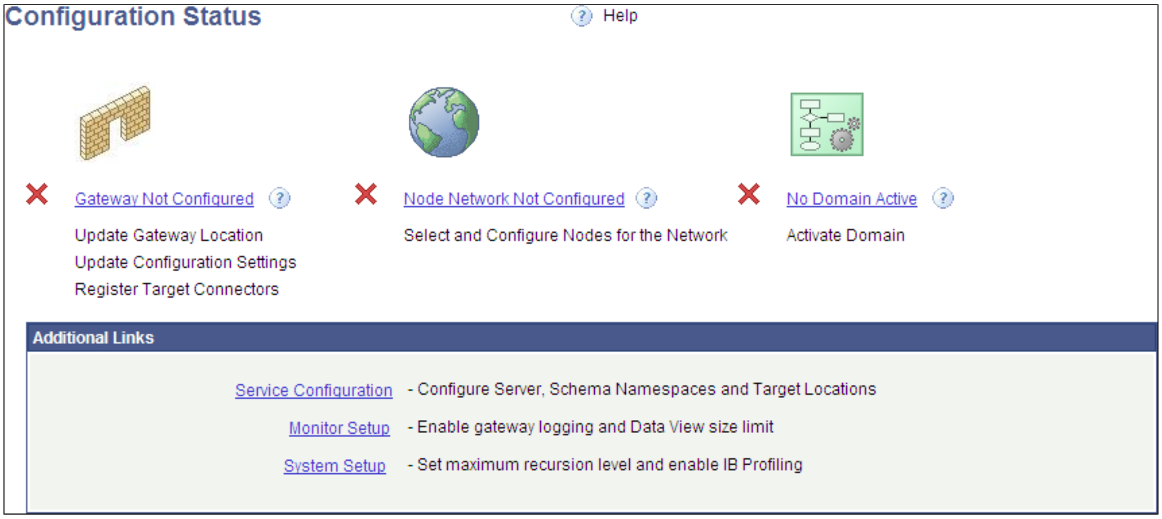
To configure the Integration Broker gateway and target nodes:

1. Select PeopleTools > Integration Broker > Integration Network > Configuration Status.

   The Configuration Status page appears.

   Image: Configuration Status page showing that the Integration Network is not configured

   This example illustrates the fields and controls on the Configuration Status page showing that the Integration Network is not configured.



Configuration Status

Help

✗ Gateway Not Configured

Update Gateway Location

Update Configuration Settings

Register Target Connectors

✗ Node Network Not Configured

Select and Configure Nodes for the Network

✗ No Domain Active

Activate Domain

Additional Links

Service Configuration - Configure Server, Schema Namespaces and Target Locations

Monitor Setup - Enable gateway logging and Data View size limit

System Setup - Set maximum recursion level and enable IB Profiling

   See "Using the Configuration Status Page" (PeopleTools 8.59: Integration Broker Administration).

2. Click the Gateway Not Configured link.

   The Gateways page appears.

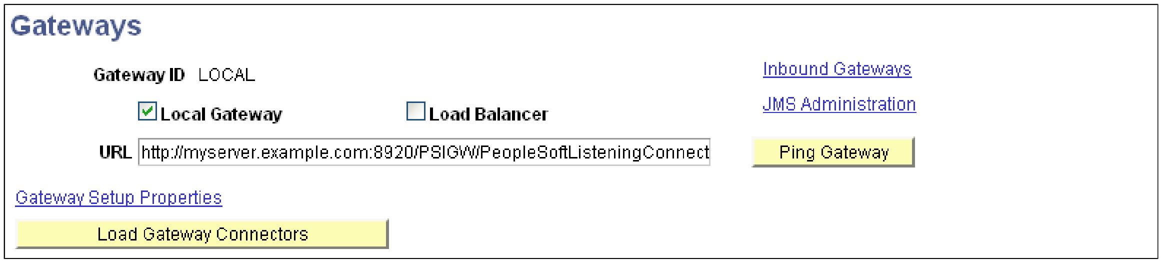
   See "Verifying and Managing Integration Gateway Configuration" (PeopleTools 8.59: Integration Broker Administration).

3. Configure the integration gateway:

a. Enter the machine name and complete URL to the PeopleSoftListeningConnector in the Gateway URL field:

Image: Gateways page with the gateway URL configured.

This example illustrates the fields and controls on the Gateways page with the gateway URL configured.



Gateways

Gateway ID LOCAL

☑ Local Gateway □ Load Balancer

URL http://myserver.example.com:8920/PSIGW/PeopleSoftListeningConnect

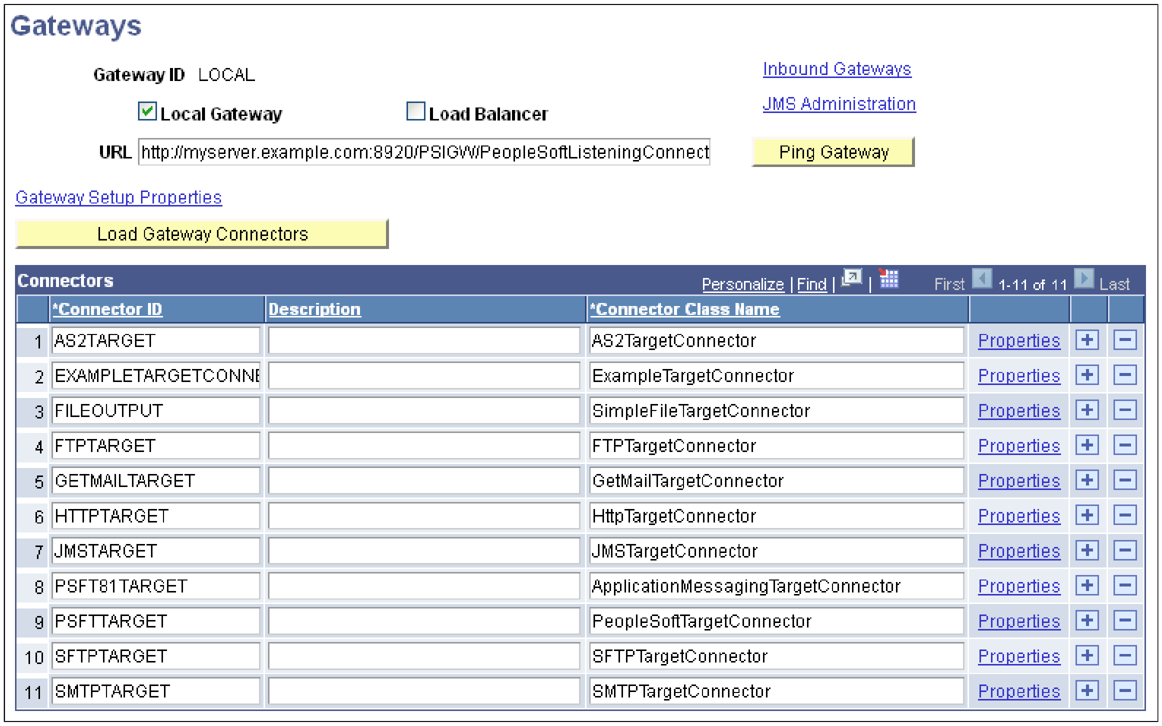
b. Click the Ping Gateway button.

The status should return as active.

c. Click the Load Gateway Connectors button to load the gateway connectors.

Image: Gateways page with connectors loaded

This example illustrates the fields and controls on the Gateways page with connectors loaded.



Gateways

Gateway ID LOCAL

☑ Local Gateway □ Load Balancer

URL http://myserver.example.com:8920/PSIGW/PeopleSoftListeningConnect

Connectors

	*Connector ID	Description	*Connector Class Name			
1	AS2TARGET		AS2TargetConnector	Properties	+	−
2	EXAMPLETARGETCONNE		ExampleTargetConnector	Properties	+	−
3	FILEOUTPUT		SimpleFileTargetConnector	Properties	+	−
4	FTPTARGET		FTPTargetConnector	Properties	+	−
5	GETMAILTARGET		GetMailTargetConnector	Properties	+	−
6	HTTPTARGET		HttpTargetConnector	Properties	+	−
7	JMSTARGET		JMSTargetConnector	Properties	+	−
8	PSFT81TARGET		ApplicationMessagingTargetConnector	Properties	+	−
9	PSFTTARGET		PeopleSoftTargetConnector	Properties	+	−
10	SFTPTARGET		SFTPTargetConnector	Properties	+	−
11	SMTPTARGET		SMTPTargetConnector	Properties	+	−

d. Click the Save button to save your changes.

4. Click the Gateway Setup Properties link.

5. Log in on the Gateway Properties page.

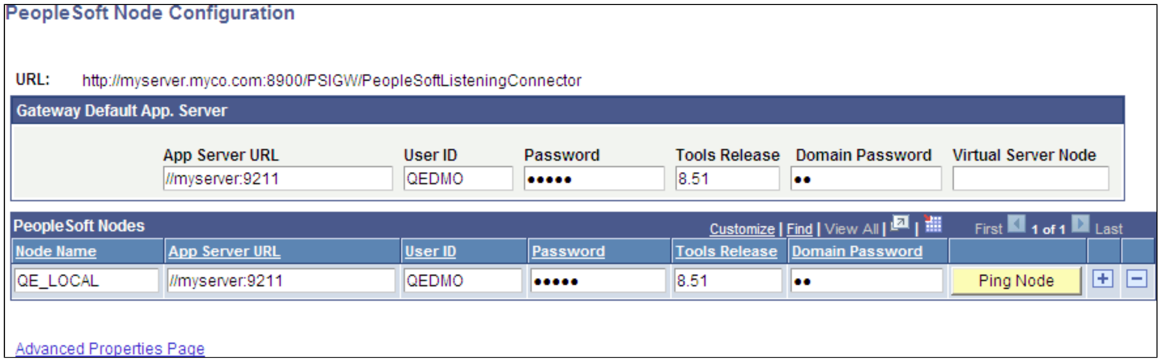
   The PeopleSoft Node Configuration page appears.

   See "Setting Oracle Jolt Connection Properties" (PeopleTools 8.59: Integration Broker Administration).

   a. Enter values for your environment. Enter a default target node in the Gateway Default App Server group box and the local target node in the PeopleSoft Nodes group box.

      Image: PeopleSoft Node Configuration page

      This example illustrates the fields and controls on the PeopleSoft Node Configuration page.



PeopleSoft Node Configuration

URL: http://myserver.myco.com:8900/PSIGW/PeopleSoftListeningConnector

Gateway Default App. Server

App Server URL	User ID	Password	Tools Release	Domain Password	Virtual Server Node
//myserver:9211	QEDMO	●●●●●	8.51	●●	

PeopleSoft Nodes

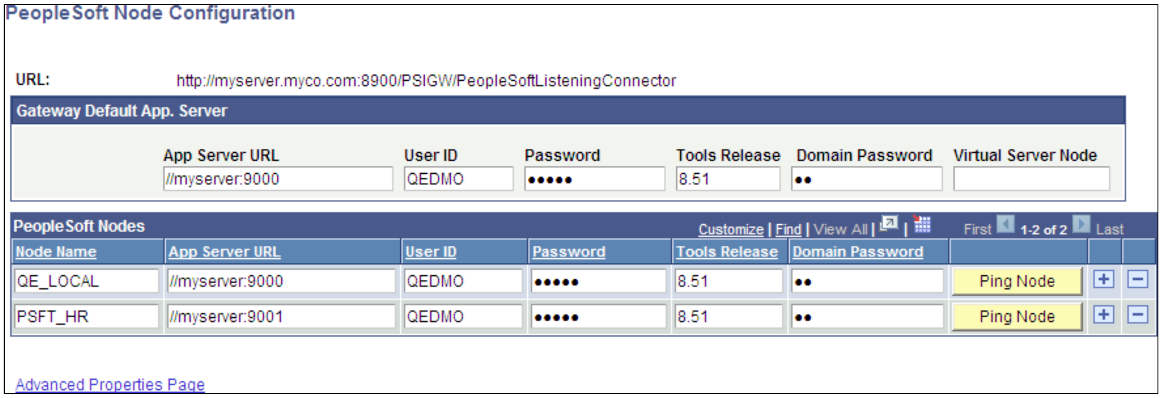
Node Name	App Server URL	User ID	Password	Tools Release	Domain Password			
QE_LOCAL	//myserver:9211	QEDMO	●●●●●	8.51	●●	Ping Node	+	−

Advanced Properties Page

   b. For a shared gateway, also enter remote target nodes (the default local node on the remote system) in the PeopleSoft Nodes group box.

      Image: PeopleSoft Node Configuration page (shared gateway configuration)

      This example illustrates the fields and controls on the PeopleSoft Node Configuration page with a shared gateway configuration.



PeopleSoft Node Configuration

URL: http://myserver.myco.com:8900/PSIGW/PeopleSoftListeningConnector

Gateway Default App. Server

App Server URL	User ID	Password	Tools Release	Domain Password	Virtual Server Node
//myserver:9000	QEDMO	●●●●●	8.51	●●	

PeopleSoft Nodes

Node Name	App Server URL	User ID	Password	Tools Release	Domain Password			
QE_LOCAL	//myserver:9000	QEDMO	●●●●●	8.51	●●	Ping Node	+	−
PSFT_HR	//myserver:9001	QEDMO	●●●●●	8.51	●●	Ping Node	+	−

Advanced Properties Page

   c. Click the Save button.

   d. Click the Ping Node button for each node.

      The status should return as success.

Note: If you receive an Integration Broker authentication error, the default local node requires that the authentication option be set to password or certificate. See the troubleshooting topic for more information.

e. On the Ping Node Results page, click the Return button.

6. On the PeopleSoft Node Configuration page, click the Advanced Properties Page link.

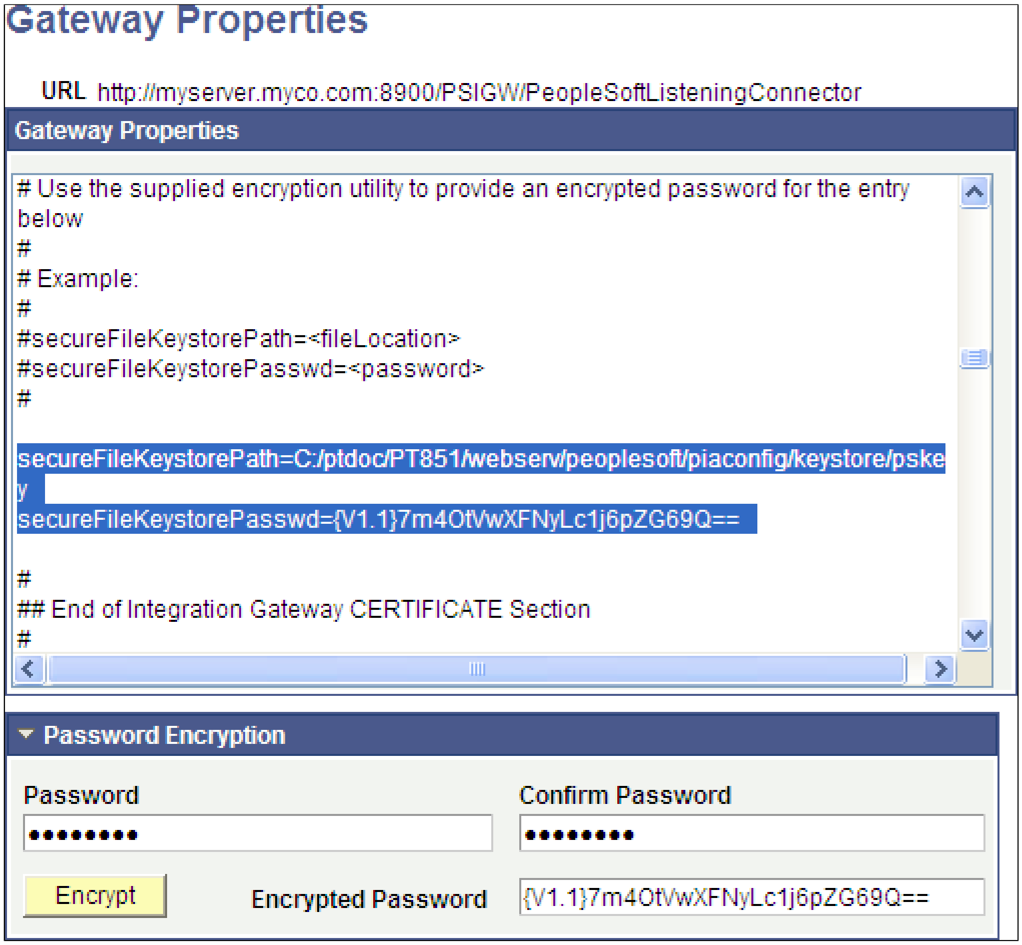
The Gateway Properties page appears.

See .

a. Enter the full path to the keystore file and enter the encrypted keystore password:

Image: Gateway Properties page

This example illustrates the fields and controls on the Gateway Properties page (setting the keystore location and encrypted password).



Gateway Properties

URL http://myserver.myco.com:8900/PSIGW/PeopleSoftListeningConnector

Gateway Properties

```
# Use the supplied encryption utility to provide an encrypted password for the entry
below
#
# Example:
#
#secureFileKeystorePath=<fileLocation>
#secureFileKeystorePasswd=<password>
#

secureFileKeystorePath=C:/ptdoc/PT851/webserv/peoplesoft/piaconfig/keystore/pske
y
secureFileKeystorePasswd={V1.1}7m4OtVwXFNyLc1j6pZG69Q==

#
## End of Integration Gateway CERTIFICATE Section
#
```

Password Encryption

Password: ••••••••

Confirm Password: ••••••••

Encrypted Password: {V1.1}7m4OtVwXFNyLc1j6pZG69Q==

See "Configuring Security and General Properties" (PeopleTools 8.59: Integration Broker Administration), "Encrypting Passwords" (PeopleTools 8.59: Integration Broker Administration).

b. Click the OK button.

7. On the PeopleSoft Node Configuration page, click the Save button again.
8. Click the OK button.
9. On the Gateways page, click the Return to Configuration Status link.

   The information on the Configuration Status page should indicate that the gateway is now configured.

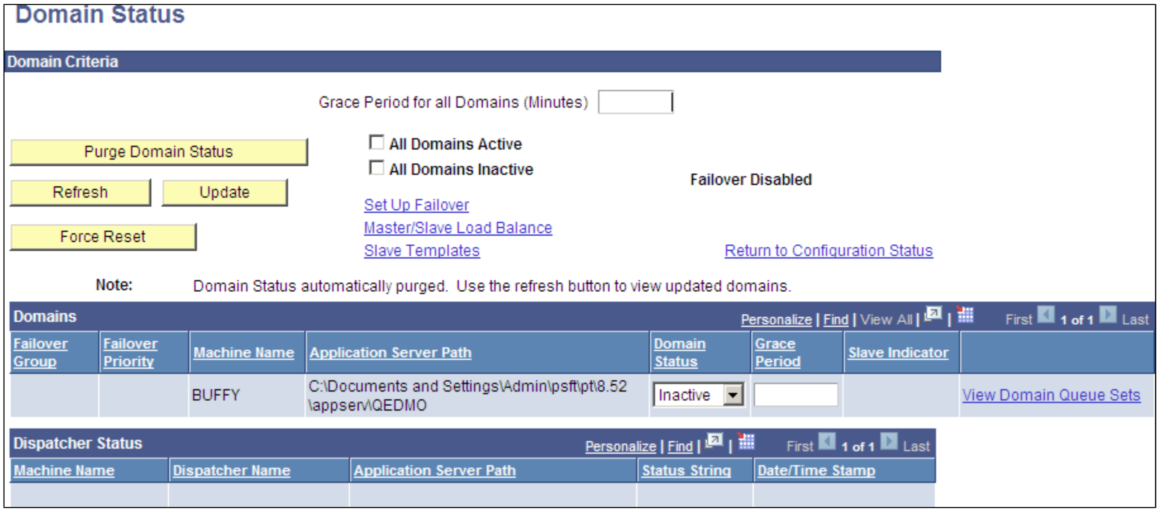
   The Configuration Status page might also indicate that the node network is configured. If the default local node is the only Integration Broker node that has been configured, then the Configuration Status page will indicate that the node network is complete. If additional Integration Broker nodes have been configured, then the Configuration Status page will indicate that the node network is not complete. However, for feed publishing and consumption, it is not necessary to complete the node network.

10. Click the No Domain Active link.

    The Domain Status page appears.

    Image: Domain Status page

    This example illustrates the fields and controls on the Domain Status page with no active domains.



Grace Period for all Domains (Minutes)

Failover Disabled

Note: Domain Status automatically purged. Use the refresh button to view updated domains.

Domains

Failover Group	Failover Priority	Machine Name	Application Server Path	Domain Status	Grace Period	Slave Indicator	
		BUFFY	C:\Documents and Settings\Admin\psft\pt\8.52 \appserv\QEDMO	Inactive			View Domain Queue Sets

Dispatcher Status

Machine Name	Dispatcher Name	Application Server Path	Status String	Date/Time Stamp

    a. In the Domains group box, set the status for this machine to *Active.*
    b. Also in the Domains group box, set the status for any pub/sub domains to *Active* to support scheduled feeds.

    Note: The pub/sub domain must be enabled first in the application server configuration through psadmin.

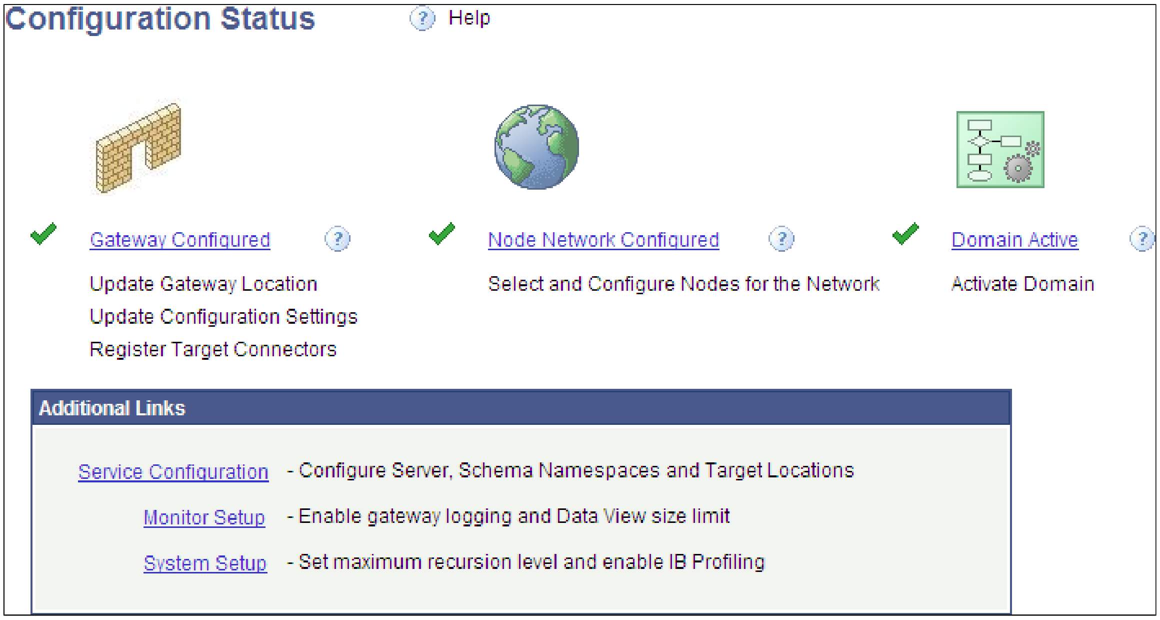
    c. Click the Update button to update the domain status.

11. Return to the Configuration Status page.

    The status should show that all steps have been completed and that your gateway and Integration Network are configured.

Image: Configuration Status page

This example illustrates the Configuration Status page showing a completed configuration.



Configuration Status

Help

✓ Gateway Configured

Update Gateway Location

Update Configuration Settings

Register Target Connectors

✓ Node Network Configured

Select and Configure Nodes for the Network

✓ Domain Active

Activate Domain

Additional Links

Service Configuration - Configure Server, Schema Namespaces and Target Locations

Monitor Setup - Enable gateway logging and Data View size limit

System Setup - Set maximum recursion level and enable IB Profiling

Configuring Integration Broker Service Target Locations

To configure service target locations:

1. Select PeopleTools >Integration Broker >Configuration >Service Configuration.

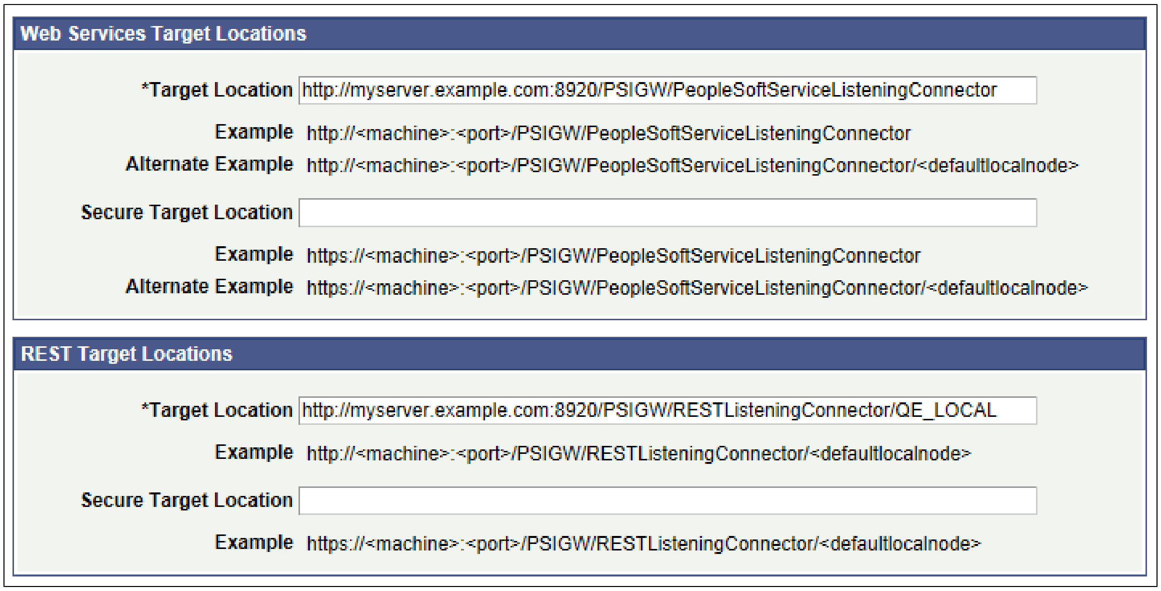
   See "Using the Service Configuration Page to Set Service Configuration Properties" (PeopleTools 8.59: Integration Broker Administration).

2. Click the Setup Target Locations link.

3. Enter a value for the target location or locations pointing to the appropriate Integration Broker gateway connector:

Image: Target Locations page

This example illustrates the fields and controls on the Target Locations page.



Web Services Target Locations

*Target Location http://myserver.example.com:8920/PSIGW/PeopleSoftServiceListeningConnector

Example http://<machine>:<port>/PSIGW/PeopleSoftServiceListeningConnector

Alternate Example http://<machine>:<port>/PSIGW/PeopleSoftServiceListeningConnector/<defaultlocalnode>

Secure Target Location

Example https://<machine>:<port>/PSIGW/PeopleSoftServiceListeningConnector

Alternate Example https://<machine>:<port>/PSIGW/PeopleSoftServiceListeningConnector/<defaultlocalnode>

REST Target Locations

*Target Location http://myserver.example.com:8920/PSIGW/RESTListeningConnector/QE_LOCAL

Example http://<machine>:<port>/PSIGW/RESTListeningConnector/<defaultlocalnode>

Secure Target Location

Example https://<machine>:<port>/PSIGW/RESTListeningConnector/<defaultlocalnode>

- If you plan to use basic authentication only and Secure Sockets Layer (SSL) has not been configured on your system, then enter a value in the Target Location field only.
- If you plan to use secure authentication and SSL has been configured on your system, then enter a value in the Target Location field and in the Secure Target Location field.

Important! You should use the secure authentication approach.
If you specify both target locations, then the secure target location is always used for access to feeds.

For more information on setting up SSL, see "Implementing WebLogic SSL Keys and Certificates" (PeopleTools 8.59: System and Server Administration).

4. Click the OK button.
5. On the Service Configuration page, click the Save button to save the configuration.

Configuring the Default User of the ANONYMOUS Node

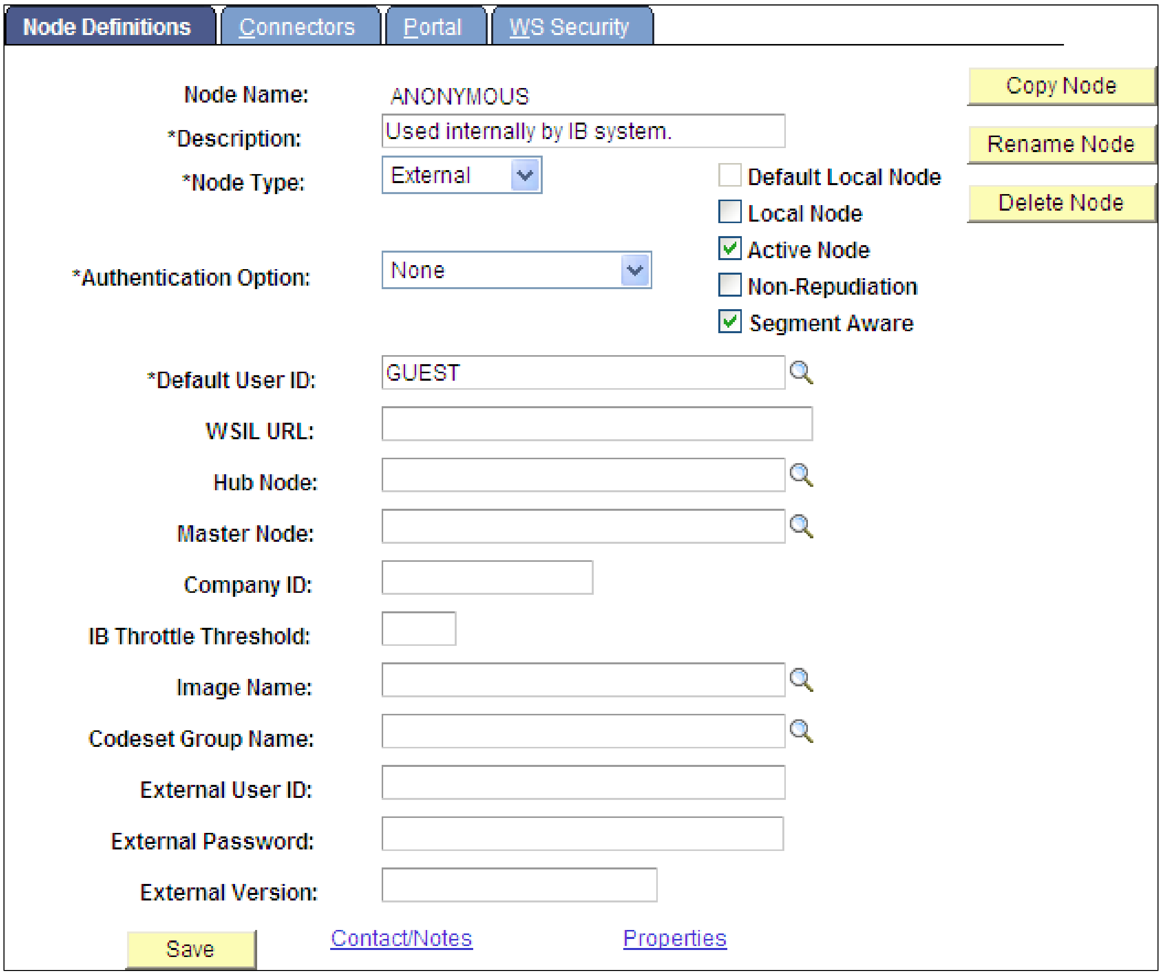
To configure the default user of the ANONYMOUS node:

1. Select PeopleTools >Integration Broker >Integration Setup >Node Definitions.
2. Select the ANONYMOUS node.
3. Select the Node Definitions page.
4. Verify that the Default User ID field contains a valid user with limited privileges, such as, the GUEST user.

Note: At a minimum, this user ID must be assigned the PTPT5500 (Feed Service Access) permissions.

Image: Node Definitions page - ANONYMOUS node

This example illustrates the fields and controls on the Node Definitions page.



Node Name: ANONYMOUS

*Description: Used internally by IB system.

*Node Type: External

□ Default Local Node

□ Local Node

☑ Active Node

□ Non-Repudiation

☑ Segment Aware

*Authentication Option: None

*Default User ID: GUEST

WSIL URL:

Hub Node:

Master Node:

Company ID:

IB Throttle Threshold:

Image Name:

Codeset Group Name:

External User ID:

External Password:

External Version:

See "Defining Node Parameters" (PeopleTools 8.59: Integration Broker Administration).

Configuring the Default Local Node

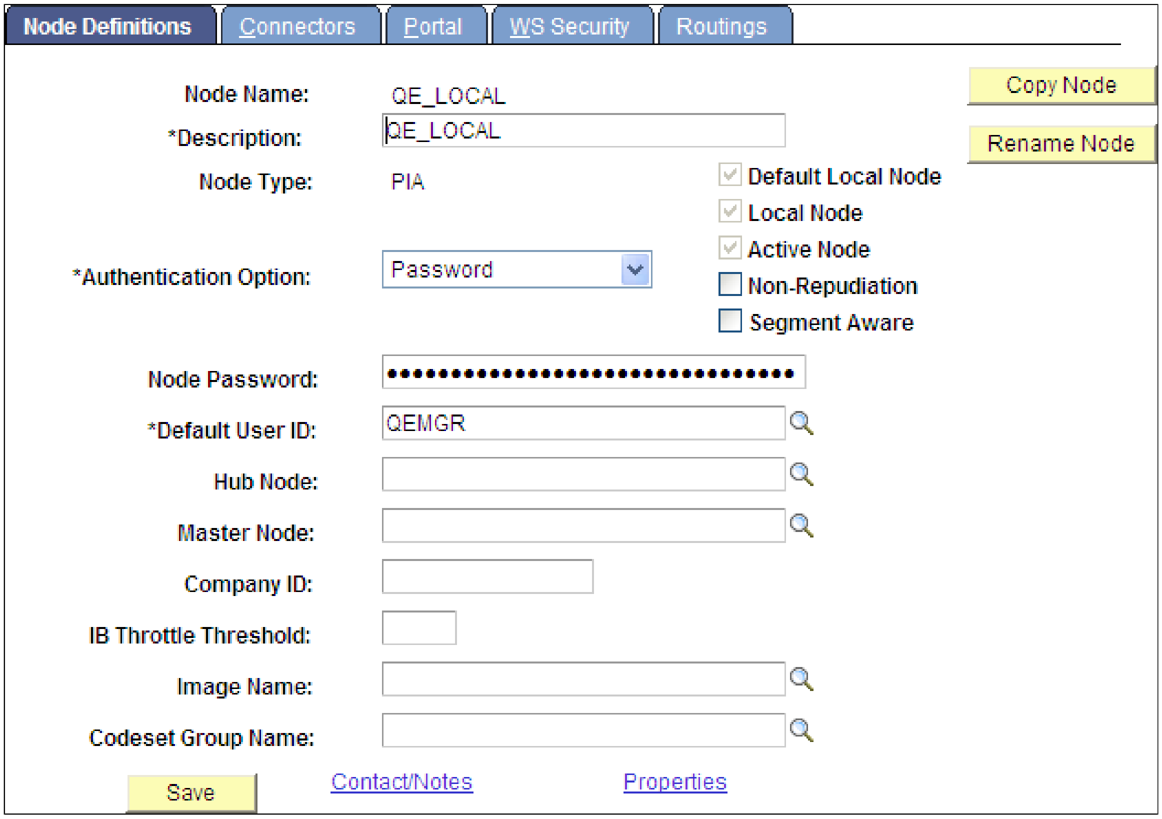
To configure the default local node:

1. Select PeopleTools >Integration Broker >Integration Setup >Node Definitions.
2. Select the default local node.
3. Select the Node Definitions page.
    a. Verify that the node type is PIA and that the Authentication Option field value is *Password* or *Certificate.*

b. Verify that the node password and default user ID are set:

Image: Node Definitions page - Default local node

This example illustrates the fields and controls on the Node Definitions page.



Node Name: QE_LOCAL

*Description: QE_LOCAL

Node Type: PIA

*Authentication Option: Password

☑ Default Local Node

☑ Local Node

☑ Active Node

□ Non-Repudiation

□ Segment Aware

Node Password: ********************************

*Default User ID: QEMGR

Hub Node:

Master Node:

Company ID:

IB Throttle Threshold:

Image Name:

Codeset Group Name:

See "Implementing Node Authentication" (PeopleTools 8.59: Integration Broker Administration), "Defining Node Parameters" (PeopleTools 8.59: Integration Broker Administration).

4. Select the Connectors page and then:

   a. Click the Ping Node button.

      It should show success.

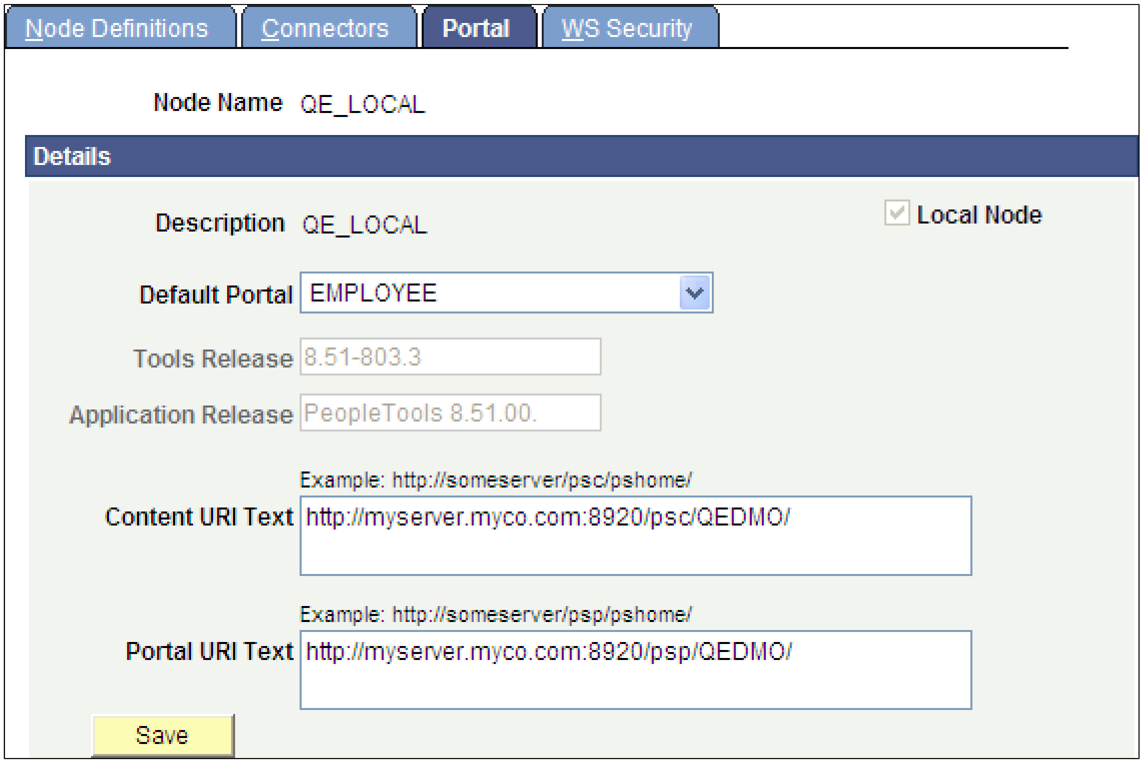
   b. Click the Return button.

5. Select the Portal page:

   a. Enter values for the Content URI Text field and the Portal URI Text field:

Image: Portal page - Default local node

This example illustrates the fields and controls on the Portal page.



Node Name QE_LOCAL

Description QE_LOCAL

☑ Local Node

Default Portal EMPLOYEE

Tools Release 8.51-803.3

Application Release PeopleTools 8.51.00.

Example: http://someserver/psc/pshome/

Content URI Text http://myserver.myco.com:8920/psc/QEDMO/

Example: http://someserver/psp/pshome/

Portal URI Text http://myserver.myco.com:8920/psp/QEDMO/

See "Setting Portal Node Characteristics" (PeopleTools 8.59: Portal Technology).

b. Save the configuration of the default local node.

Important! If you set or changed the authentication option on the Node Definitions page, log out and log back in again. Otherwise, you might get an authentication token error when you attempt to access a feed.

Setting URI Text for Local Host Nodes

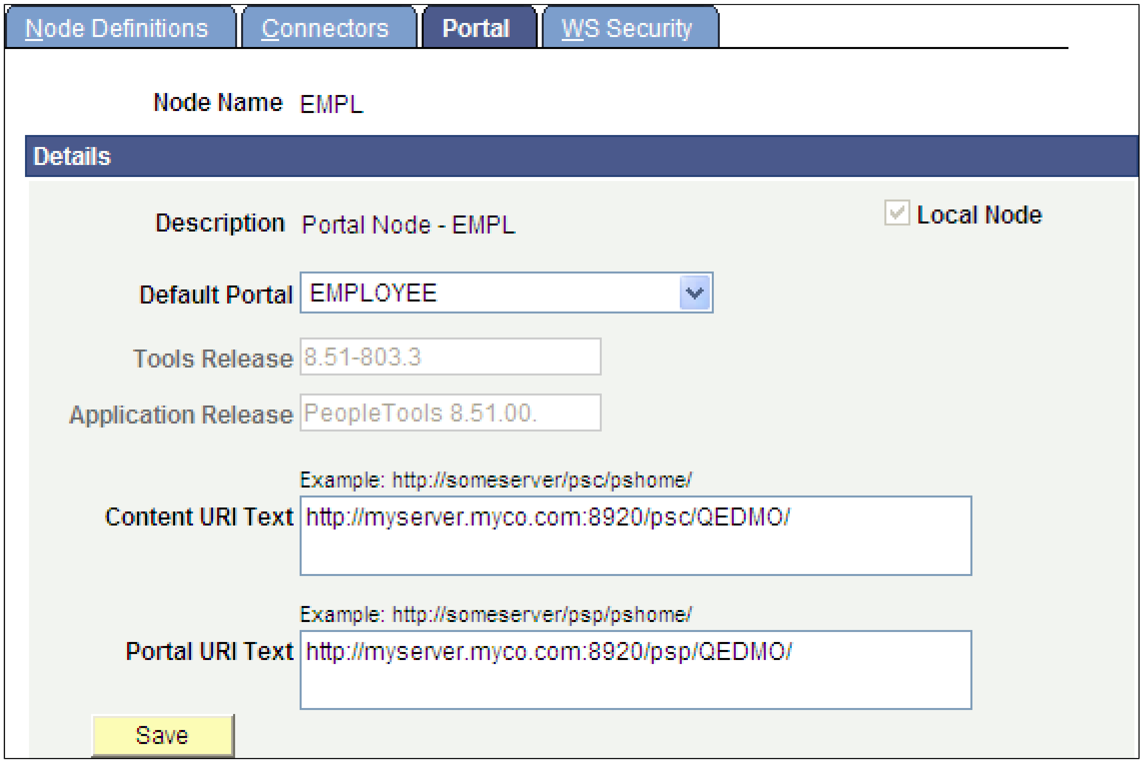
For each local host node, you must set the URI text for the node definition.

Repeat the following procedure for each local host node:

1. Select PeopleTools >Integration Broker >Integration Setup >Node Definitions.
2. Select a local host node.
3. Select the Portal page:
   a. Enter values for the Content URI Text field and the Portal URI Text field:

Image: Portal page - Local host node

This example illustrates the fields and controls on the Portal page.



Node Name EMPL

Description Portal Node - EMPL

☑ Local Node

Default Portal EMPLOYEE

Tools Release 8.51-803.3

Application Release PeopleTools 8.51.00.

Example: http://someserver/psc/pshome/

Content URI Text http://myserver.myco.com:8920/psc/QEDMO/

Example: http://someserver/psp/pshome/

Portal URI Text http://myserver.myco.com:8920/psp/QEDMO/

See "Setting Portal Node Characteristics" (PeopleTools 8.59: Portal Technology).

b. Save the configuration of the local host node.

Chapter 4

Creating and Using Feeds and Feed Templates

Understanding Creation and Use of Feeds and Feed Templates

This section discusses how to create and use feeds and feed templates.

Process Flow for Feed Publishing and Consumption

The business process flow for developing, administering, and viewing feeds includes the following steps:

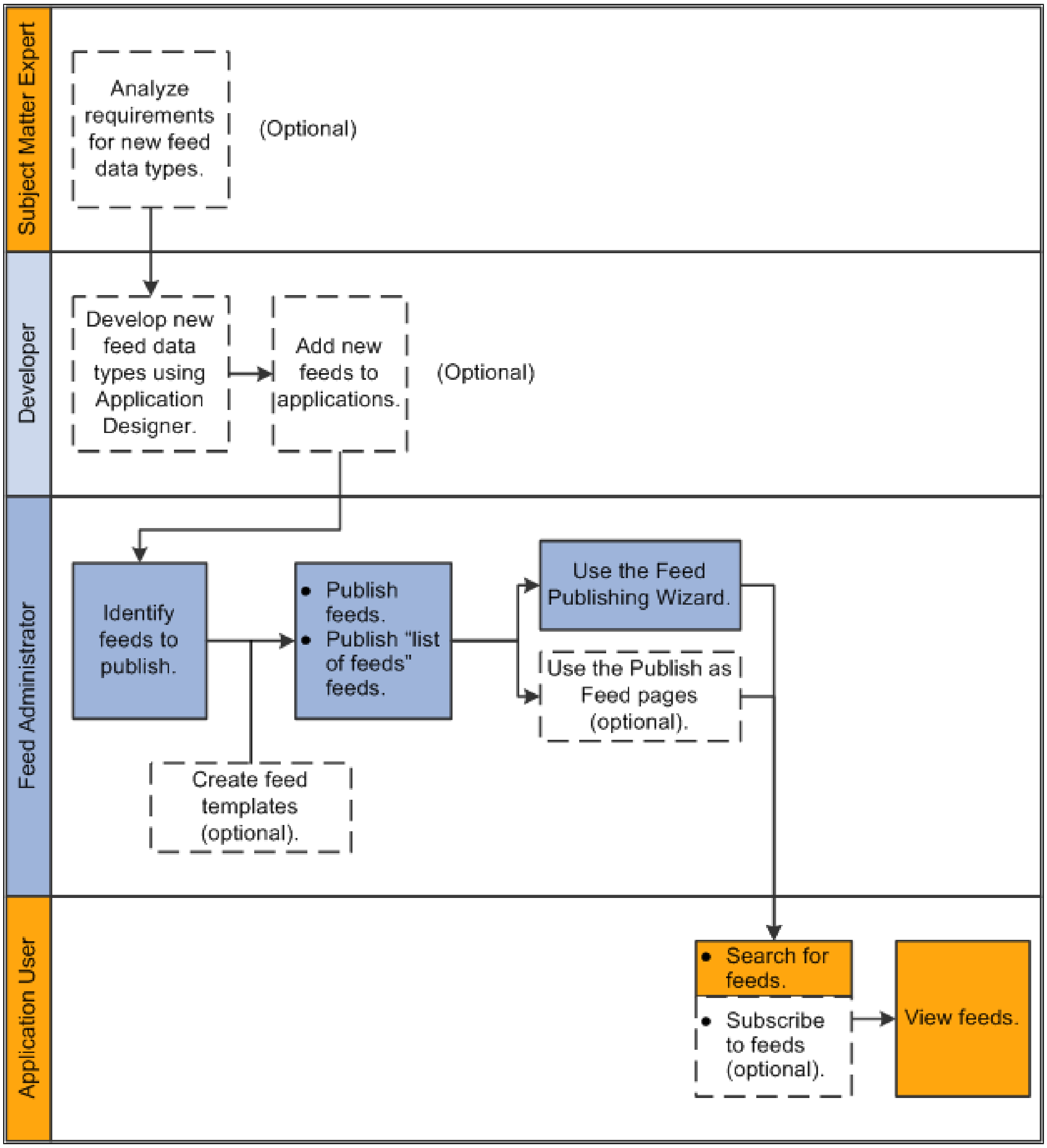
1. (Optional) The subject matter expert analyzes the requirements for new feed data types.
2. (Optional) The application developer develops new feeds.
   a. The application developer develops the new feed data types using Application Designer.
   b. The application developer adds the new feed types to the application.
3. The feed administrator identifies the feeds to publish.
4. (Optional) The feed administrator creates feed templates to aid with feed publication.
5. The feed administrator publishes:
   a. Individual feeds.
   b. "List of feeds" feeds.
6. The feed administrator creates and publishes feed definitions using one of the following methods:
   - The four steps of the Feed Publishing Wizard.

     (Optional) Feed templates are used to aid in creating feed definitions.
   - The Publish as Feed pages.

     (Optional) Feed templates are used to aid in creating feed definitions.
7. The application user searches for feeds and then:
   a. Views the feed.
   b. (Optional) Subscribes to the feed.

Image: Business process flow for developing, administering, and viewing feeds

The following diagram illustrates the business process flow for developing, administering, and viewing feeds.


Subject Matter Expert
Analyze requirements for new feed data types.
(Optional)
Developer
Develop new feed data types using Application Designer.
Add new feeds to applications.
(Optional)
Feed Administrator
Identify feeds to publish.
• Publish feeds.
• Publish "list of feeds" feeds.
Use the Feed Publishing Wizard.
Use the Publish as Feed pages (optional).
Create feed templates (optional).
Application User
• Search for feeds.
• Subscribe to feeds (optional).
View feeds.

Methods for Publishing Feeds

The Feed Publishing Framework provides two methods for publishing feeds:

- Feed Publishing Wizard.
- Publish as Feed links and pages.

In a PeopleSoft system, administrative users and content owners with appropriate authorization can use either method to create, view, edit, and delete feed definitions related to a particular data source.

Feed Publishing Wizard

The Feed Publishing Wizard provides an easy to follow, four-step guide to publish feeds from any data source for the feed data types defined in the system. The Feed Publishing Wizard simplifies the creation and editing of feed definitions. In addition, the Feed Publishing Wizard:

- Provides the ability to create and edit feed templates.
- Allows you to specify data source parameters as *personalized* — that is, as user specified or administrator specified.

See Publishing Feeds Using the Feed Publishing Wizard.

Publish as Feed Links and Pages

The Publish as Feed links and Publish as Feed pages provide another access point and mechanism for publishing feeds. The Publish as Feed links can be found on the content maintenance pages for specific types of content only. In PeopleTools, these Publish as Feed links are found on the maintenance pages for content associated with the four delivered PeopleTools feed data types: FEED, GENERICFEED, PSQUERY, and WORKLIST.

When using these publish pages, the feed administrator is not guided, but must follow a defined process for creating a feed definition.

See Publishing Feeds Using the Publish as Feed Pages.

This table lists the base feed data types delivered in PeopleTools and the navigation path to the publish pages for each data type:

Feed Data Type	***Navigation***
FEED	PeopleTools > Feeds > Define Feed Data Types See Publishing a List of Feeds Feed.
GENERICFEED	PeopleTools > Feeds > Define IB Generic Message Feed See Understanding Integration Broker Generic Message Feeds.
PSQUERY	Reporting Tools > Query > Query Manager See Understanding Query Feeds.
WORKLIST	Worklist > My Worklist - Summary View Worklist > My Worklist - Detail View See Understanding Worklist Feeds.

Feed Templates

The Feed Publishing Framework provides the ability to create feed templates, which aid in predefining and reusing feed settings and parameters. Feed templates can be created from scratch or an existing feed definition can be saved as a feed template. Feed templates can then be used when creating new feed definitions or when editing existing feed definitions.

Publishing Feeds Using the Feed Publishing Wizard

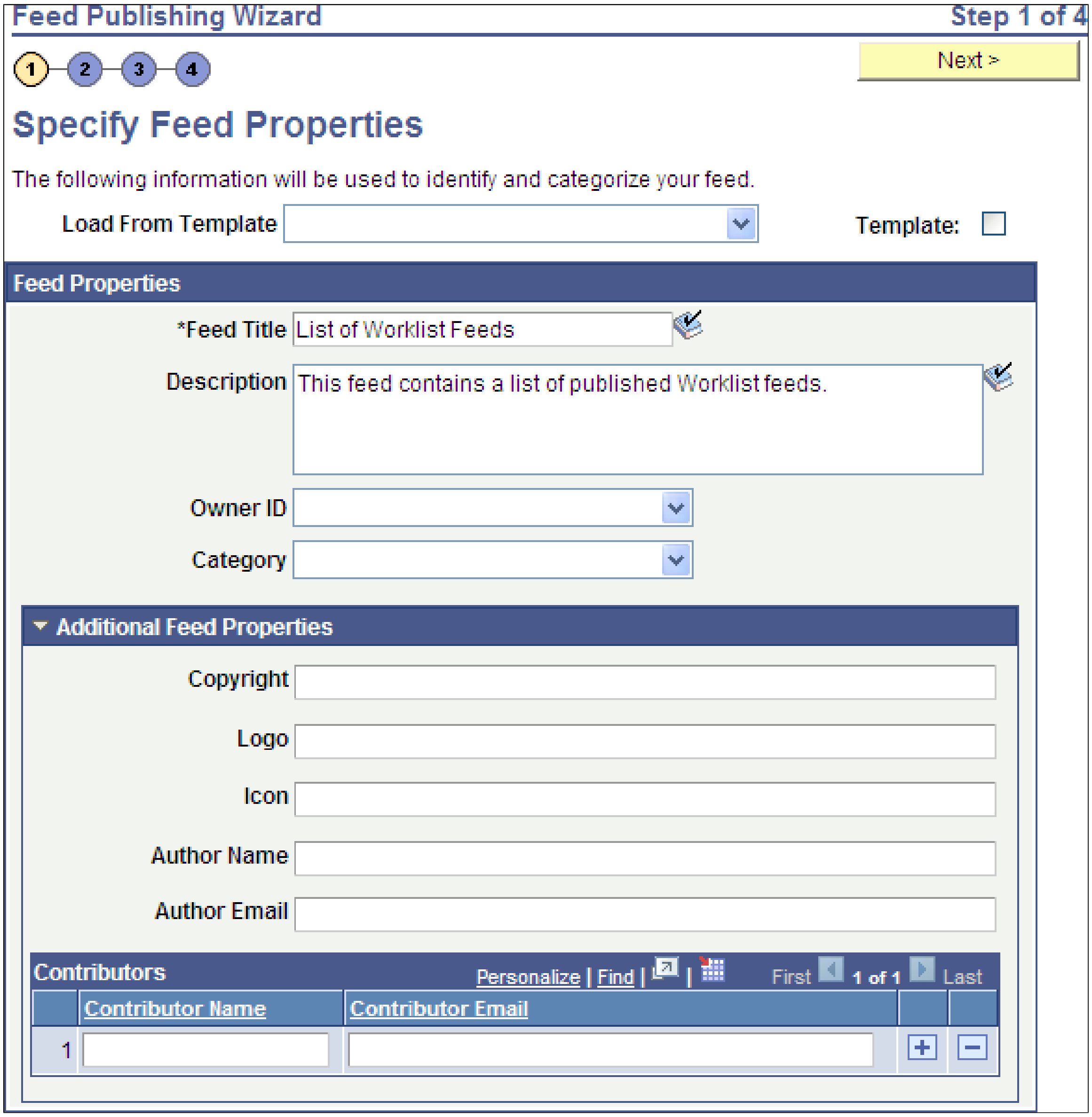
This section discusses how to use the Feed Publishing Wizard to publish a feed in four steps.

Step 1: Specifying Feed Properties

Access the Feed Publishing Wizard - Specify Feed Properties page (PeopleTools >Feeds >Feed Publishing Wizard; or click the Go to Feed Publishing Wizard link on any Publish Feed Definition page).

Image: Feed Publishing Wizard - Specify Feed Properties page

This example illustrates the fields and controls on the Feed Publishing Wizard - Specify Feed Properties page. You can find definitions for the fields and controls later on this page.



Specify Feed Properties

The following information will be used to identify and categorize your feed.

Load From Template

Template: □

Feed Properties

*Feed Title: List of Worklist Feeds

Description: This feed contains a list of published Worklist feeds.

Owner ID

Category

Additional Feed Properties

Copyright

Logo

Icon

Author Name

Author Email

Contributors

	Contributor Name	Contributor Email		
1			+	−

Use the Specify Feed Properties page to define feed properties, such as the feed title, as well as advanced feed properties.

Load From Template

The list displays *all* the available feed templates in the system. Selecting a feed template loads the current feed definition with values from the template.

Template

Select this option to make the current feed definition a feed template.

Important! Unlike the Save As Template link, this option does not create a new object. If you are editing an existing feed definition, the feed definition will be converted into a feed template, which means that the feed itself will no longer be available. Conversely, if you are editing a feed template and deselect this option, the feed template will be converted to a feed definition.

Feed Properties

Feed Title

Enter a short description for this feed at a maximum of 30 characters.

Description

Enter no more than a 254-character description of the feed.

Owner ID

Select the owner ID for the feed definition.

The owner ID is a way to identify which definitions are owned by which PeopleSoft applications, such as PeopleSoft General Ledger, Accounts Receivables, PeopleTools, and so on. The values in the drop-down list box are Translate table values associated with the OBJECTOWNERID field.

Note: The owner ID does not appear in the published feed document.

Category

Select a category for the feed definition from the list of active categories.

If a feed definition is assigned to a category and then that category is deleted, the feed definition no longer displays an assigned category.

Additional Feed Properties

Not all feed readers display all feed properties. This list describes the additional feed properties that some feed readers process and display.

Note: Values entered in these fields replace the default feed property values defined at the feed data type level.

See Feed Document Properties.

See Creating the Feed Data Type Definition.

Copyright

Enter a copyright date to be included in the feed document.

Logo

Enter a URL to the logo to be included in the feed document, for example: *http://myserver.example.com:80/images/logo.gif*

Icon

Enter a URL to an icon to be included in the feed document, for example: *http://myserver.example.com:80/images/icon.gif*

Author Name

Enter the author's name to be included in the feed document.

Author Email

Enter the author's email address to be included in the feed document.

Contributor Name

Enter a contributor's name to be included in the feed document.

Contributor Email

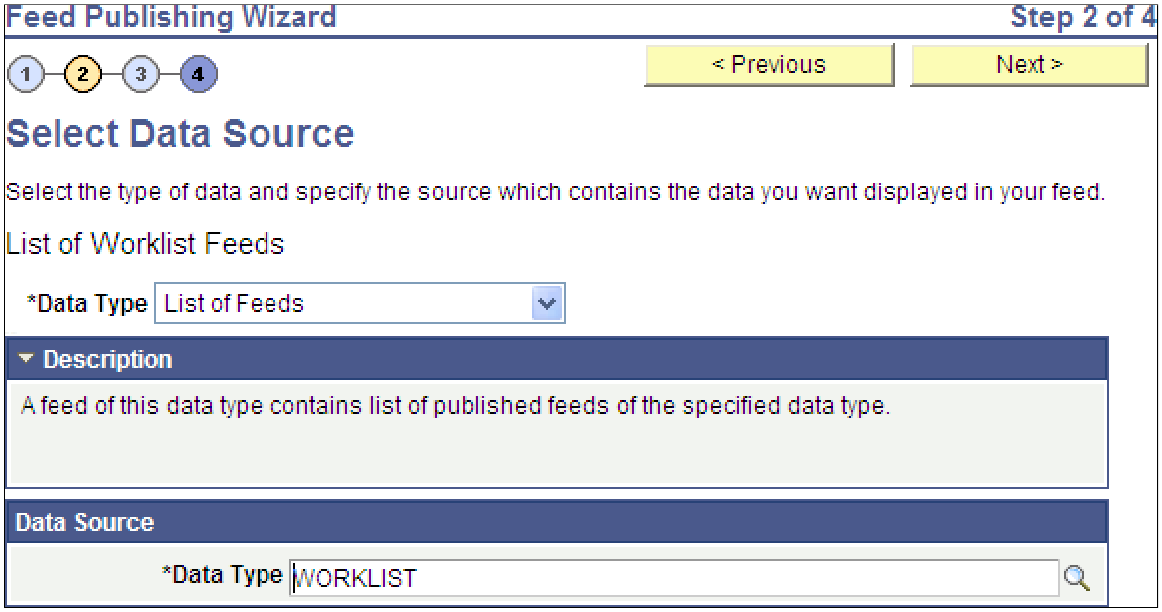
Enter the contributor's email address to be included in the feed document.

Step 2: Selecting a Data Source

Access the Feed Publishing Wizard - Select Data Source page (click the Next button on the Feed Publishing Wizard - Specify Feed Properties page).

Image: Feed Publishing Wizard - Select Data Source page

This example illustrates the fields and controls on the Feed Publishing Wizard - Select Data Source page. You can find definitions for the fields and controls later on this page.



Select Data Source

Select the type of data and specify the source which contains the data you want displayed in your feed.

List of Worklist Feeds

*Data Type List of Feeds

Description

A feed of this data type contains list of published feeds of the specified data type.

Data Source

*Data Type WORKLIST

Use the Select Data Source page to select the feed data type and then the data source settings for that feed data type.

Data Type

Select the feed data type from the drop-down list of active feed data types.

Data Source

Specify the data source settings. Typically, you will need to select a specific data source; however, depending on the feed data type, there could be zero to three data source settings.

Note: After selecting a data source, you can save the feed definition. However, saving the feed definition is not the same as publishing the feed. You must complete step 4 in the wizard to publish a feed.

Specifying Additional Query Feed Options

If the selected feed data type is PS Query Feeds and you have selected the query data source, then additional grids are displayed for advanced options, query fields, and feed entry content mapping.

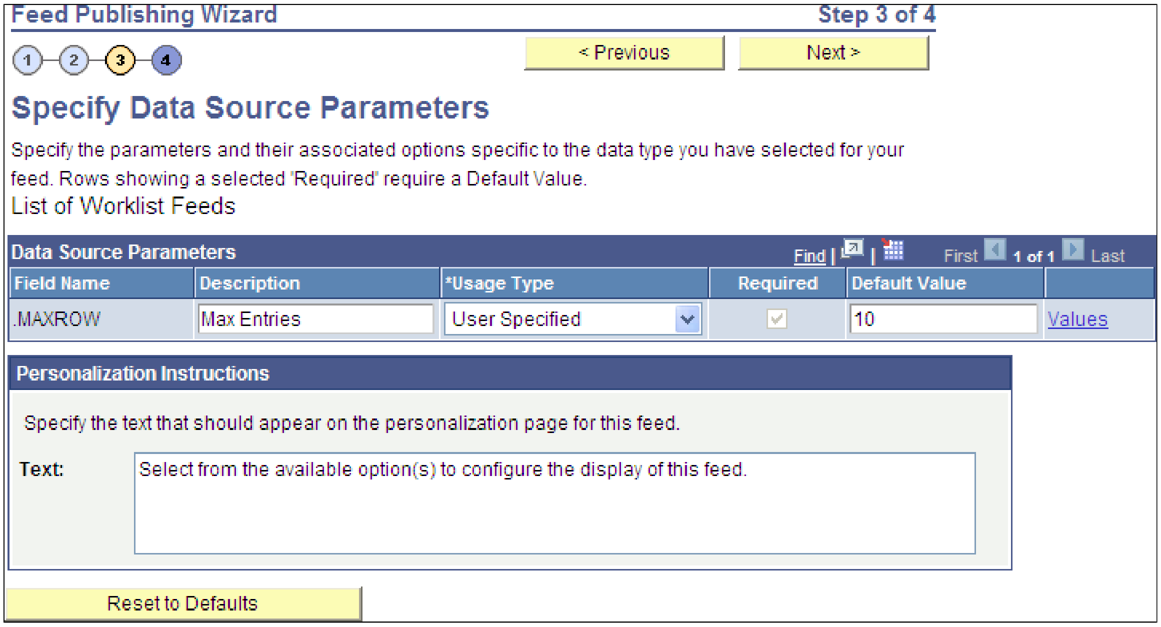
See Publishing Query Feeds.

Step 3: Specifying Data Source Parameters

Access the Feed Publishing Wizard - Select Data Source Parameters page (click the Next button on the Feed Publishing Wizard - Select Data Source page).

Image: Feed Publishing Wizard - Select Data Source Parameters page

This example illustrates the fields and controls on the Feed Publishing Wizard - Select Data Source Parameters page. You can find definitions for the fields and controls later on this page.



Specify Data Source Parameters

Specify the parameters and their associated options specific to the data type you have selected for your feed. Rows showing a selected 'Required' require a Default Value.

List of Worklist Feeds

Data Source Parameters

Field Name	Description	*Usage Type	Required	Default Value	
.MAXROW	Max Entries	User Specified	☑	10	Values

Personalization Instructions

Specify the text that should appear on the personalization page for this feed.

Text: Select from the available option(s) to configure the display of this feed.

Use the Select Data Source Parameters page to configure each data source parameter, such as the parameter's description, usage type, and default value. Each data source parameter can have one value only. Required parameters must have a default value assigned.

Note: This page is accessible only when you are building a feed definition with a data source that allows modification of one or more data source parameters.

Field Name Displays the name of the data source parameter.

Description Displays a description of the data source parameter.

Usage Type The usage type defines how the parameter gets a value at runtime:

- *Admin Specified:* Select to allow administrative users to specify values to configure the feed definition for their user community or communities. The value of an administrator-specified parameter can be derived as follows: as a system default value, as an administrator-configured value, or as a pure runtime value. The following list describes each method for deriving the value of an administrator-specified parameter listed in their order of precedence when the feed is requested:

  1. *Runtime value:* Runtime values are values passed in using what the industry refers to as "query parameters" of the feed request. Runtime values are not stored.

     Specifically, this means that a standard feed URL can also include a data source parameter and its value in the form of a query parameter (that is, &PARAM=value). A standard feed URL will already include some query parameters that are consumed by the feed runtime engine to generate the feed document. Additional query parameters, such as data source parameters, can be passed on to the data source itself—for example, to a PeopleSoft query, to a worklist, and so on.

     For example, if a PeopleSoft query included a field named DAYS and this field was also defined as a query prompt, then the feed URL could be formed with a runtime value to provide the value for this prompt: http://myserver.example.com:8920/PSIGW/HttpListeningConnector/feeds/RealtimeQueryFeed?FEED_ID=MY_FEED&PORTAL_NAME=EMPLOYEE&NODE_NAME=PSFT_TRN&To=PSFT_TRN&DAYS=60

  2. *Administrator-configured value:* Administrator-configured values can be specified by configuring the feed through the My Feeds page. Administrator-configured values are stored as feed personalization data.

  3. *System default values:* System default values can be specified on this page at design time, and are stored with the feed definition.

  4. *Empty values:* Provides the correct empty value according to the parameter's data type. For example if the parameter is a number, then 0 would be provided; for a string, "" would be provided.

- *Fixed:* Select to enter a fixed value for the data source parameter. This value is stored with the feed definition.

- *System Variable:* Select to assign a system variable as the data source parameter value. The actual value of the parameter is determined at runtime.

  For example, suppose that you specify *%DateTime* as the system variable for a parameter named Timestamp. When the feed is requested, the Timestamp parameter is populated by the current value of %DateTime.

- *User Specified:* Select to allow end users to specify values to personalize the feed definition for their own use. The value of an user-specified parameter can be derived as follows: as a system default value, as a user-configured value, or as a pure runtime value.

  Note: User personalization does not apply to feeds published as public and a warning message is issued when the feed definition is saved.

  The following list describes each method for deriving the value of a user-specified parameter listed in their order of precedence when the feed is requested:

  1. *Runtime value:* Runtime values are values passed in using what the industry refers to as "query parameters" of the feed request. Runtime values are not stored.

     Specifically, this means that a standard feed URL can also include a data source parameter and its value in the form of a query parameter (that is, &PARAM=value). A standard feed URL will already include some query parameters that are consumed by the feed runtime engine to generate the feed document. Additional query parameters, such as data source parameters, can be passed on to the data source itself—for example, to a PeopleSoft query, to a worklist, and so on.

     For example, if a PeopleSoft query included a field named DAYS and this field was also defined as a query prompt, then the feed URL could be formed with a runtime value to provide the value for this prompt: http://myserver.example.com:8920/PSIGW/HttpListeningConnector/feeds/RealtimeQueryFeed?FEED_ID=MY_FEED&PORTAL_NAME=EMPLOYEE&NODE_NAME=PSFT_TRN&To=PSFT_TRN&DAYS=60

  2. *User-configured value:* User-configured values can be specified by personalizing the feed through the My Feeds page. User-configured values are stored as feed personalization data.

Users can select a value from a prompt, or they can manually enter their own value if no prompt values are available. To define the list of valid prompt values, click the Values link. This list of prompt values and the default value are stored with the feed definition.

3. *System default values:* System default values can be specified on this page at design time, and are stored with the feed definition.
4. *Empty values:* Provides the correct empty value according to the parameter's data type. For example if the parameter is a number, then 0 would be provided; for a string, "" would be provided.

Required

A data source parameter is defined as required in the application class for the feed data type. This check box is selected but disabled except for user-specified parameters. All required parameters should be assigned a default value before proceeding to step 4 in the wizard.

Default Value

Enter the default value for the data source parameter. The data type of the value you enter should be the same as the parameter's data type. However, the entered value is not type checked by the Feed Publishing Wizard; the type should be checked by and any errors reported by the application class for the feed data type.

Consider these guidelines when entering a default value:

- If you select *User Specified* as the usage type and the data source parameter is a required value, you must enter the default value that will appear before a user enters a value. If the data source parameter is not a required value, you do not have to enter a default value.

  If you select a default value when defining prompt values on the Feed Publishing Wizard - Specify Data Source Parameter Values page, that default value populates this field.

  If you select *User Specified* as the usage type, you can also enter a system variable as the default value. For example, to make the current date the default value for a user-specified date parameter, enter a default value of *%Date.*
- If you select *System Variable* as the usage type, you must enter a supported system variable to use as the data source parameter value. You can use the Look up Value button to access the following list of supported system variables:

Supported System Variables	***Supported System Variables***
%AuthenticationToken	%ClientDate
%ClientTimeZone	%ContentID
%ContentType	%Copyright
%Currency	%Date
%DateTime	%DbName
%DbServerName	%DbType
%EmailAddress	%EmployeeID
%HPTabName	%IsMultiLanguageEnabled
%Language*	%Language_Base
%MaxInterlinkSize	%LocalNode
%Market	%MaxMessageSize
%NavigatorHomePermissionList	%Node*
%PasswordExpired	%Portal*
%PrimaryPermissionList	%ProcessProfilePermissionList
%RowSecurityPermissionList	%RunningInPortal
%ServerTimeZone	%Time
%UserDescription	%UserID

* The language code, portal name, and node name passed in through the query parameters of the request are used as the values for %Language, %Portal, and %Node respectively. If the request does not contain these parameters, then the corresponding system variable's value will be used.

- If you select *Fixed* as the usage type, you must enter the fixed value if this is a required parameter.

Values

Note: The Values link is displayed for parameters with an associated translate table, Yes/No type of parameters, or if you select *User Specified* as the usage type.

Click Values link to access the Feed Publishing Wizard - Specify Data Source Parameter Values page.

Text

Note: If you select *User Specified* as the usage type, the Personalization Instructions group box is displayed.

Use the Text field to enter custom personalization instructions for the feed. These personalization instructions appear on the personalization page of the feed.

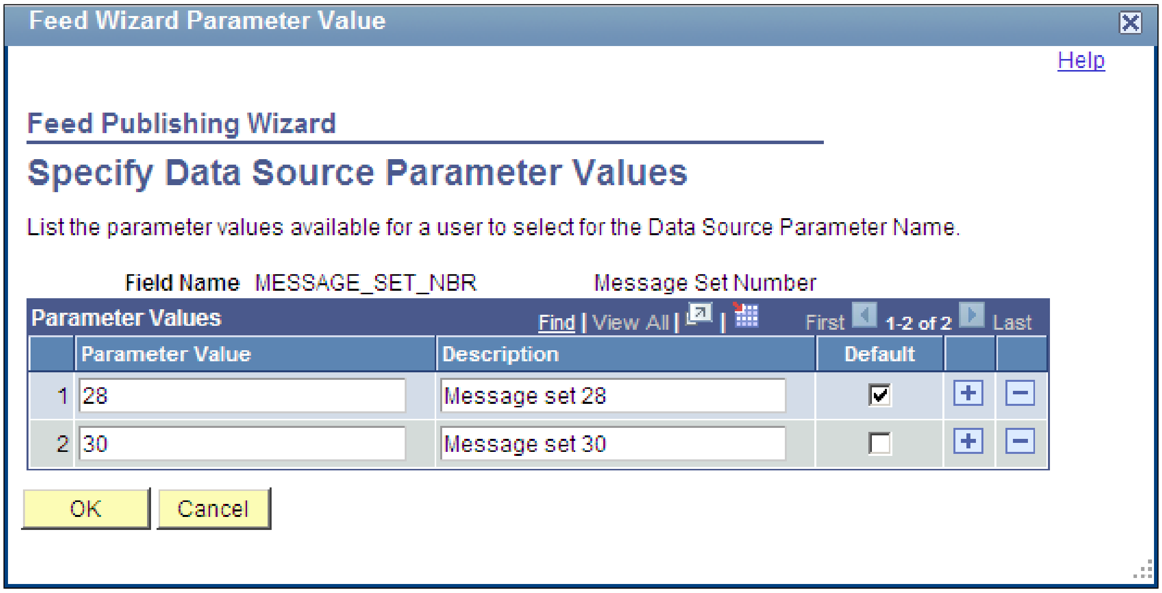
Note: Personalization instructions must be translatable.

Specifying Prompt Values for Data Source Parameters

Access the Feed Publishing Wizard - Specify Data Source Parameter Values page (click a Values link on the Feed Publishing Wizard - Select Data Source Parameters page).

Image: Feed Publishing Wizard - Specify Data Source Parameter Values page

This example illustrates the fields and controls on the Feed Publishing Wizard - Specify Data Source Parameter Values page. You can find definitions for the fields and controls later on this page.



Feed Publishing Wizard

Specify Data Source Parameter Values

List the parameter values available for a user to select for the Data Source Parameter Name.

Field Name MESSAGE_SET_NBR Message Set Number

Parameter Values

	Parameter Value	Description	Default		
1	28	Message set 28	☑	+	−
2	30	Message set 30	□	+	−

Use the Specify Data Source Parameter Values page to specify the prompt values that are displayed to users when they personalize the feed.

For Yes/No type of parameters and for parameters associated with a translate table, you cannot modify or add to the values listed on this page. However, for user-specified parameters, you can add, edit, or delete value-description pairs. In addition, you can specify the default value for the data source parameter.

Parameter Value

Specify the parameter value.

Description

Provide an optional description of the parameter value.

Default

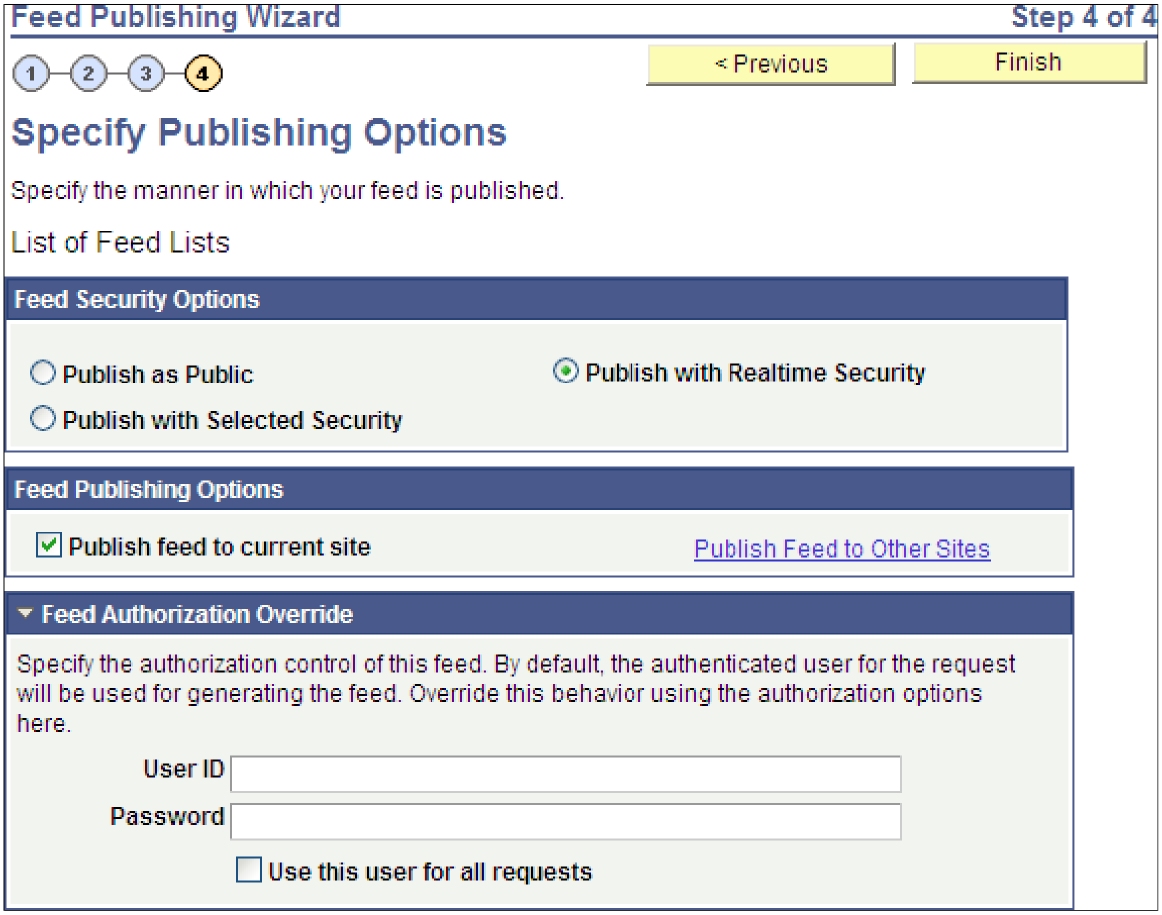
Select one value as the default value. If the parameter is required, then a default value is required; otherwise, it is optional.

Step 4: Specifying Publishing Options

Access the Feed Publishing Wizard - Specify Publishing Options page (click the Next button on the Feed Publishing Wizard - Select Data Source Parameters page).

Image: Feed Publishing Wizard - Specify Publishing Options page

This example illustrates the fields and controls on the Feed Publishing Wizard - Specify Publishing Options page. You can find definitions for the fields and controls later on this page.



Feed Publishing Wizard

Step 4 of 4

Specify Publishing Options

Specify the manner in which your feed is published.

List of Feed Lists

Feed Security Options

○ Publish as Public

● Publish with Realtime Security

○ Publish with Selected Security

Feed Publishing Options

☑ Publish feed to current site

Publish Feed to Other Sites

Feed Authorization Override

Specify the authorization control of this feed. By default, the authenticated user for the request will be used for generating the feed. Override this behavior using the authorization options here.

User ID

Password

□ Use this user for all requests

Use the Specify Publishing Options page to specify the publishing options for a feed definition.

A feed definition can be published to multiple sites, or to no site at all. Publishing a feed definition to a site makes that feed available to all users of that site. Once published, a feed can be removed from a site by deselecting the publish option. Users will no longer be able to access and view an unpublished feed; however, the feed definition for an unpublished feed can still be edited using the Feed Publishing Wizard.

Feed Security Options

For query feeds and worklist feeds, the Feed Security Options group box is hidden. For worklist feeds, real-time security is always used. For query feeds, security can be set in Step 2 of the Feed Publishing Wizard.

Publish as Public Select to make the feed available for public access.

Publish with Realtime Security

Select to have the system check the data source object to determine whether the viewer has access to the feed in real time. This option is the default for new feed definitions.

Publish with Selected Security

Select to make the feed available based on the viewer's role-based permission lists.

When you select this type of security, the Selected Security grid appears, enabling you to set role and permission list security.

Note: User-based permission lists, such as the Primary Permission List, are not applied with this type of security.

Feed Publishing Options

If the Template option was selected in Step 1, the Feed Publishing Options group box is hidden.

Publish feed to current site

Select this option to publish the feed to the current site. Deselect this option to remove the published feed from the current site.

Publish Feed to Other Sites

Click to save the feed definition and publish it to additional sites.

See Publishing Feed Definitions to Additional Sites.

Feed Authorization Override

By default, the authenticated user who requests a feed is used for generating the feed. You can override the default behavior by specifying a user ID and password to be used for requests for this feed. Doing so ensures every user who requests the feed sees the same result. This can be useful for public feeds.

Warning! Use care when selecting a user ID for this override as this can potentially give unauthorized users access to feed data that they would not normally be able to see.

User ID

Specify the user ID to use as the override for requests for this feed.

Password

Specify the password for this user.

Use this user for all requests

Select this option to use the specified user ID and password as the authentication override. If this option is not selected, then the user ID and password data is merely saved, but not used.

Actions

Finish

Click the Finish button to publish the feed definition.

Note: The Finish button appears for new feed definitions only. Use the Save button for already published feed definitions.

Save

Click the Save button to publish the feed definition.

Note: The Save button appears for already published feed definitions only. Use the Finish button for new feed definitions.

Preview Feed

Click to preview the published feed.

Note: This link appears for already published feed definitions only.

Save As Template

Click to save the existing feed definition as a feed template. After a feed definition is saved as a feed template, both will exist as separate entities. The original feed definition can be used normally as any other feed and the feed template can be used while creating new feeds or editing existing feed definitions. The feed template will be saved with the same security as that of the feed definition.

Note: This link appears for already published feed definitions only.

Publishing Feeds Using the Publish as Feed Pages

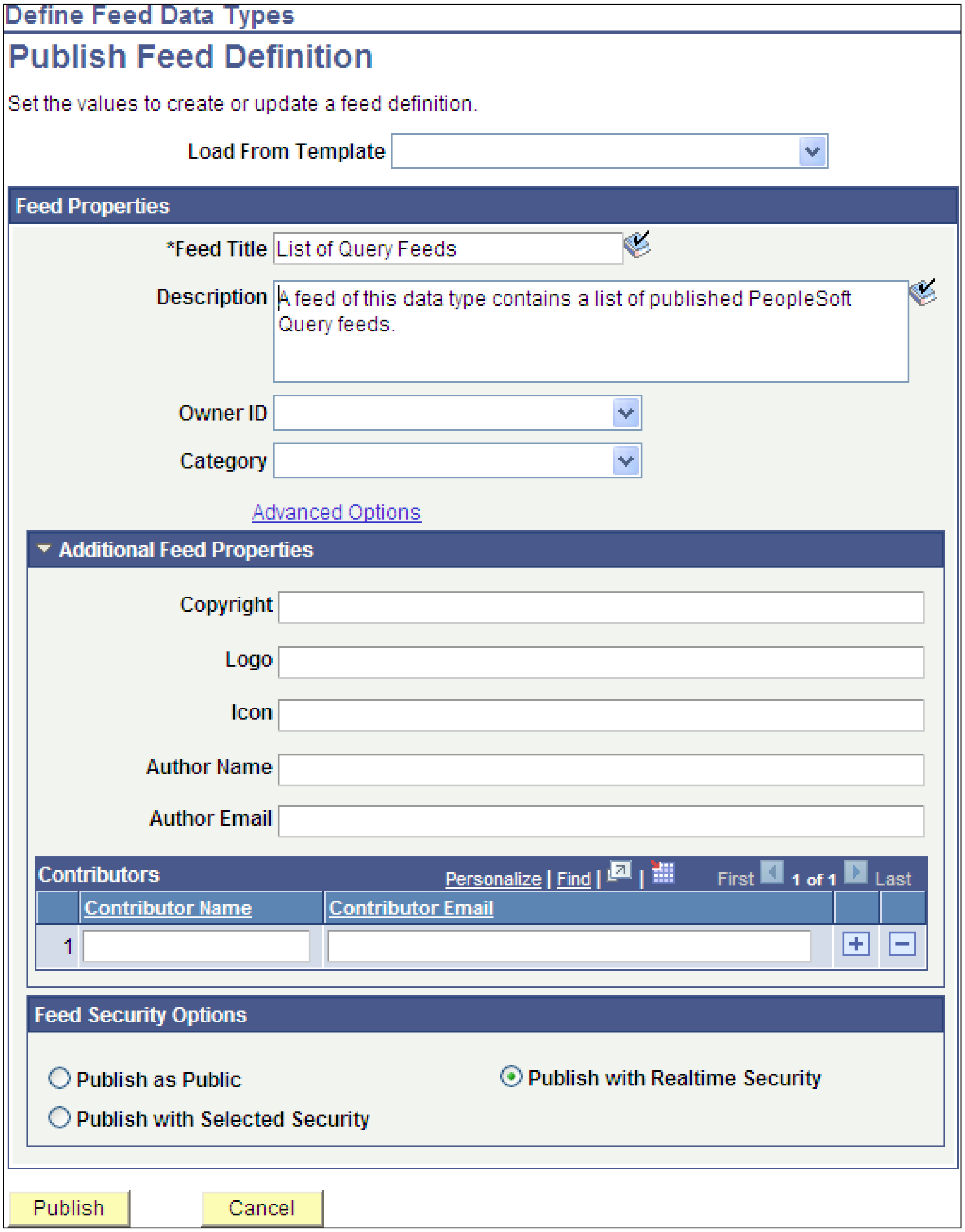
This section discusses how to publish feeds using the Publish as Feed Pages.

Defining Feed Properties

Access the Publish Feed Definition page (click the Publish as Feed link).

Image: Publish Feed Definition page for a list of feeds type feed

This example illustrates the fields and controls on the Publish Feed Definition page. You can find definitions for the fields and controls later on this page.



Define Feed Data Types

Publish Feed Definition

Set the values to create or update a feed definition.

Load From Template:

Feed Properties

*Feed Title: List of Query Feeds

Description: A feed of this data type contains a list of published PeopleSoft Query feeds.

Owner ID:

Category:

Advanced Options

Additional Feed Properties

Copyright:

Logo:

Icon:

Author Name:

Author Email:

Contributors — Personalize | Find | First 1 of 1 Last

	Contributor Name	Contributor Email		
1			+	−

Feed Security Options

○ Publish as Public
○ Publish with Selected Security
● Publish with Realtime Security

When you click the Publish as Feed link, you then must define feed properties, such as the feed title, security, and other options.

Load From Template	Feed templates that match the given data type and data source settings are available in the drop-down list. Selecting a feed template from the drop-down populates the existing feed definition with the values from the feed template.

Feed Properties

Feed Title

The system populates the field based on the selected content item information when you create a new feed definition. You can edit the field. Enter a maximum of 30 characters.

Description

Enter no more than a 254-character description of the feed.

Owner ID

Select the owner ID for the feed definition.

The owner ID is a way to identify which definitions are owned by which PeopleSoft applications, such as PeopleSoft General Ledger, Accounts Receivables, and so on. The values in the drop-down list box are Translate table values associated with the OBJECTOWNERID field.

Note: The owner ID does not appear in the published feed document.

Category

Select a category for the feed definition from the list of active categories.

If a feed definition is assigned to a category and then that category is deleted, the feed definition no longer displays an assigned category.

Advanced Options

Click the Advanced Options link to access the Advanced Options page for the feed data type. Advanced options vary by feed data type.

Additional Feed Properties

Not all feed readers display all feed properties. This list describes the additional feed properties that some feed readers process and display.

Note: Values entered in these fields replace the default feed property values defined at the feed data type level.

See Feed Document Properties.

See Creating the Feed Data Type Definition.

Copyright

Enter a copyright date to be included in the feed document.

Logo

Enter a URL to the logo to be included in the feed document, for example: *http://myserver.example.com:80/images/logo.gif*

Icon

Enter a URL to an icon to be included in the feed document, for example: *http://myserver.example.com:80/images/icon.gif*

Author Name

Enter the author's name to be included in the feed document.

Author Email

Enter the author's email address to be included in the feed document.

Contributor Name

Enter a contributor's name to be included in the feed document.

Contributor Email

Enter the contributor's email address to be included in the feed document.

Feed Security Options

The list that follows describes feed security options.

Note: Query feeds apply security on the Advanced Options page.
Worklist feeds support real-time security only, and feed security options do not appear for Worklist feeds.

See Understanding Worklist Feeds.

See Defining Advanced Options for Query Feeds.

Publish as Public

Select to make the feed available for public access.

Publish with Selected Security

Select to make the feed available based on the viewer's role-based permission lists.

When you select this type of security, the Selected Security grid appears, enabling you to set role and permission list security.

Note: User-based permission lists, such as the Primary Permission List, are not applied with this type of security.

Publish with Realtime Security

Select to have the system check the data source object to determine whether the viewer has access to the feed in real time. This option is the default for new feed definitions.

Sync with Data

Click this button to explicitly sync the feed permission with the selected feed data permissions. This button appears for already published feed definitions only when you select the Publish with Selected Security option.

Type

Select either *Role* or *Permission List* security.

Name

Enter the name of the permission list or role that has access to the feed.

Actions

Publish

Click the Publish button to publish the feed definition.

Preview Feed

Click to preview the published feed.

Note: This link only appears for already published feed definitions.

Publish Feed to Other Sites

Click to publish the already published feed definition to additional sites.

Note: This link only appears for already published feed definitions.

See Publishing Feed Definitions to Additional Sites.

Save As Template

Click to save the existing feed definition as a feed template. After a feed definition is saved as a feed template, both will exist as separate entities. The original feed definition can be used normally as any other feed and the feed template can be used while creating new feeds or editing existing feed definitions. The feed template will be saved with the same security as that of the feed definition.

Go to Feed Publishing Wizard

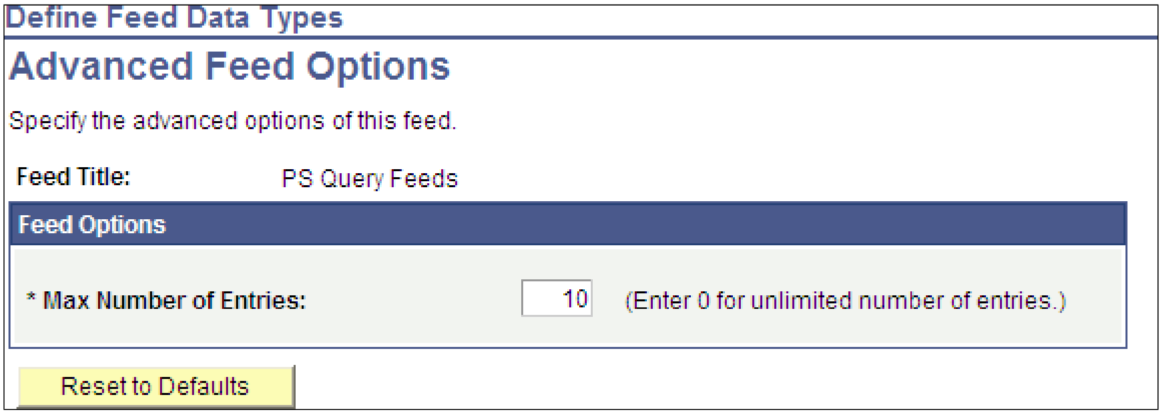
Click to save any changes and open the feed definition in the Feed Publishing Wizard

Defining Advanced Feed Options

Access the Advanced Feed Options page (click the Advanced Options link on the Publish Feed Definition page).

Image: Advanced Feed Options page for a list of feeds type feed

This example illustrates the fields and controls on the Advanced Feed Options page. You can find definitions for the fields and controls later on this page.



Define Feed Data Types

Advanced Feed Options

Specify the advanced options of this feed.

Feed Title: PS Query Feeds

Feed Options

* Max Number of Entries: 10 (Enter 0 for unlimited number of entries.)

Advanced options differ by feed data type.

Max Number of Entries

Enter the maximum number of entries that the feed should return to the user. Enter *0* for unlimited entries up to the maximum row limit specified on the Feed Options page, which is 300 by default.

Note: You can configure the upper limit on the Feed Options page. If the maximum number of entries specified is either 0 or greater than the PTFP_MAX_ROW_LIMIT, then the output is limited to the value in the PTFP_MAX_ROW_LIMIT field. If PTFP_MAX_ROW_LIMIT is 0 or is undefined, then the output is limited by the maximum number of entries.

Reset to Defaults

Click to reset any advanced options to their default values.

Related Links

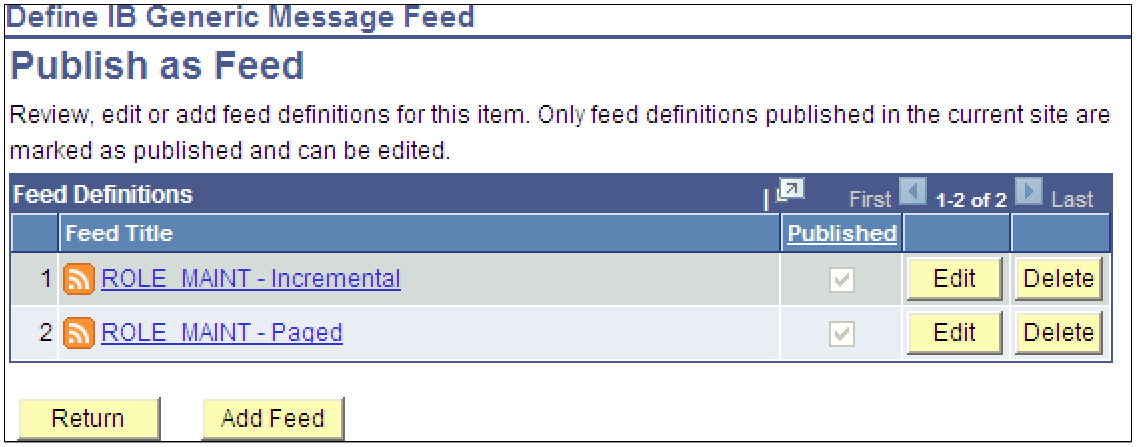
Defining Advanced Options for Generic Message Feeds
Defining Advanced Options for Query Feeds
Defining Advanced Options for Worklist Feeds

Managing Published Feeds

Access the Publish as Feed page (click the Publish as Feed link or click the Publish button on the Publish Feed Definition page).

Image: Publish as Feed page for an Integration Broker generic message type feed

This example illustrates the fields and controls on the Publish as Feed page for an Integration Broker generic message type feed. You can find definitions for the fields and controls later on this page.



Define IB Generic Message Feed

Publish as Feed

Review, edit or add feed definitions for this item. Only feed definitions published in the current site are marked as published and can be edited.

Feed Definitions | First 1-2 of 2 Last

	Feed Title	Published		
1	ROLE_MAINT - Incremental	✓	Edit	Delete
2	ROLE_MAINT - Paged	✓	Edit	Delete

Use the Publish as Feed page to review, edit, add, or delete feed definitions for this item.

Feed Title

Click a link to open the feed document for this feed definition in a separate browser window.

Edit

Click this button to access the Publish Feed Definition page on which you can edit the published feed definition.

Delete

Click this button to delete the feed definition.

Return

Click this button to return to the administration page for the item.

Add Feed	Click this button to define and publish a new feed definition for the item.

Creating and Editing Feed Templates

You can create and edit feed templates much in the same way that you create and edit feed definitions:

- At Step 1 (Feed Publishing Wizard - Specify Feed Properties page), you can select the Template option:

    - For a new definition, the object will be saved as a feed template.
    - For an existing feed definition, the object will be *converted* to a feed template, which means that the feed itself will no longer be available.

    See Step 1: Specifying Feed Properties.

- At Step 4 (Feed Publishing Wizard - Specify Publishing Options page), you can click the Save As Template link to save an existing feed definition as a feed template.

    After a feed definition is saved as a feed template, both will exist as separate entities.

    See Step 4: Specifying Publishing Options.

- On the Publish Feed Definition page, you can click the Save As Template link to save an existing feed definition as a feed template.

    After a feed definition is saved as a feed template, both will exist as separate entities.

    See Defining Feed Properties.

Publishing a List of Feeds Feed

A *list of feeds* feed is a feed that contains a list of all available feeds of a specific feed data type. When you access a list of feeds feed, you see a listing of only the feeds that you are authorized to view. You can publish a list of feeds feed through the Feed Publishing Wizard or from the Feed Data Types page.

You publish a list of feeds feed by using the FEED data type.

To publish a list of feeds feed using the Feed Publishing Wizard:

1. Select PeopleTools > Feeds > Feed Publishing Wizard.
2. Select to add a new value.
3. Enter a feed ID and click Add.
4. At Step 1, enter a feed title at a minimum.

Note: To distinguish this as a list of feeds, you can change the feed title to include “List of” — for example, “List of Query Feeds”.

See Step 1: Specifying Feed Properties.

5. Click Next.
6. At step 2, select *List of Feeds* as the data type.
7. Select the data source for which to produce the list of feeds:

   - *FEED*

     Select this option to produce a list of all published list of feeds feeds.

   - *GENERICFEED*

     Select this option to produce a list of all published generic Integration Broker message feeds.

   - *PSQUERY*

     Select this option to produce a list of all published query feeds.

   - *WORKLIST*

     Select this option to produce a list of all published worklist feeds.

   See Step 2: Selecting a Data Source.
8. Click Next.
9. Click Next to accept the defaults for the data source parameters.

   Otherwise, modify the data source parameters to suit your implementation.

   See Step 3: Specifying Data Source Parameters.
10. At Step 4, specify the feed publishing options.

    See Step 4: Specifying Publishing Options.
11. Click Finish to publish the feed definition.

Publishing Feed Definitions to Additional Sites

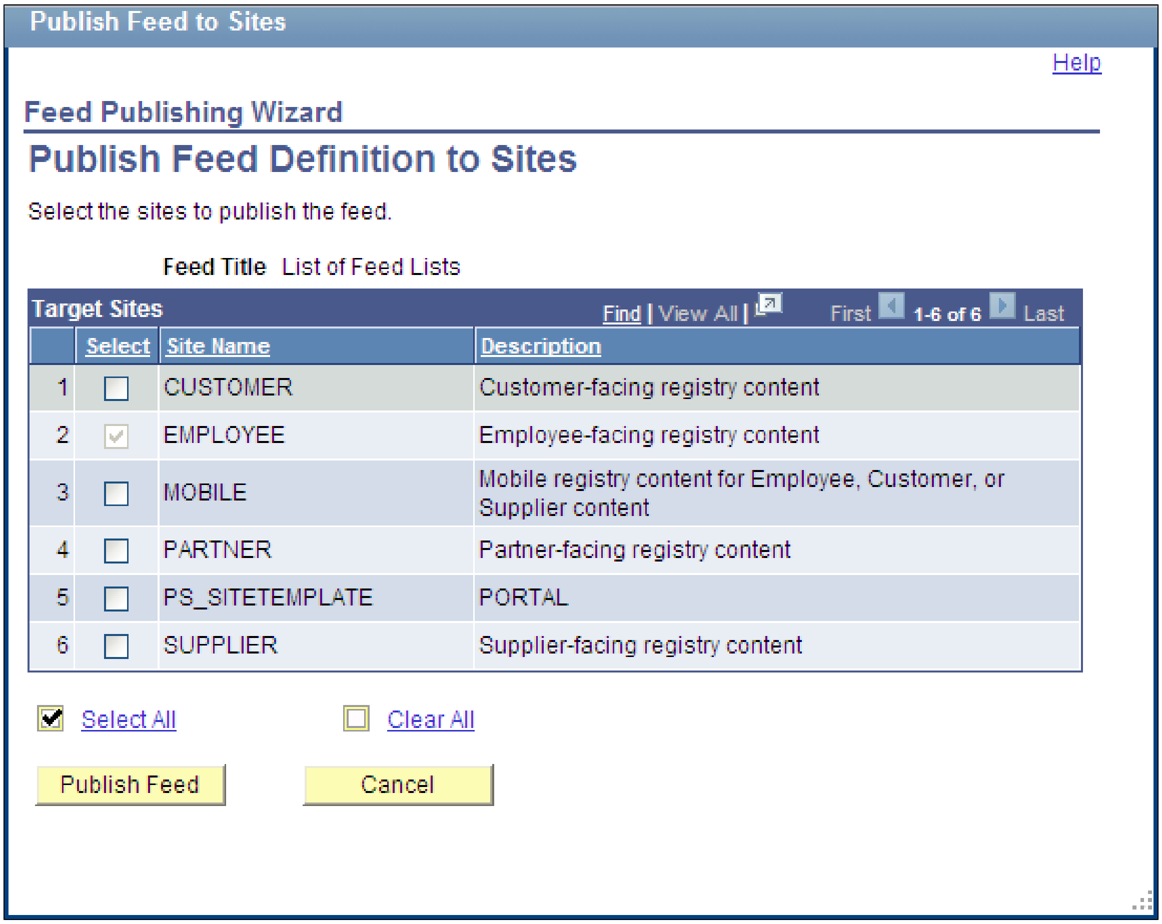
A feed definition can be published to additional sites from the Feed Publishing Wizard or from the Publish as Feed pages.

Publishing Feed Definitions to Additional Sites

Access the Publish Feed Definition to Sites page (click the Publish Feed to Other Sites link at Step 4 of the Feed Publishing Wizard or on the Publish Feed Definition page).

Image: Publish Feed Definition to Sites page

This example illustrates the fields and controls on the Publish Feed Definition to Sites page. You can find definitions for the fields and controls later on this page.



Publish Feed to Sites

Help

Feed Publishing Wizard

Publish Feed Definition to Sites

Select the sites to publish the feed.

Feed Title List of Feed Lists

Target Sites — Find | View All | First 1-6 of 6 Last

	Select	Site Name	Description
1	□	CUSTOMER	Customer-facing registry content
2	☑	EMPLOYEE	Employee-facing registry content
3	□	MOBILE	Mobile registry content for Employee, Customer, or Supplier content
4	□	PARTNER	Partner-facing registry content
5	□	PS_SITETEMPLATE	PORTAL
6	□	SUPPLIER	Supplier-facing registry content

☑ Select All □ Clear All

You can publish the feed to any site listed in the additional sites list.

Select Select this check box to publish the feed definition to this site.

Site Name Displays the valid sites to which you can publish the feed definition. This field is display-only.

☑ Select All Click to select all feed definitions in the list.

□ Clear All Click to deselect all feed definitions.

Publish Feed Click to publish the feed definition to additional sites.

Accessing Feeds

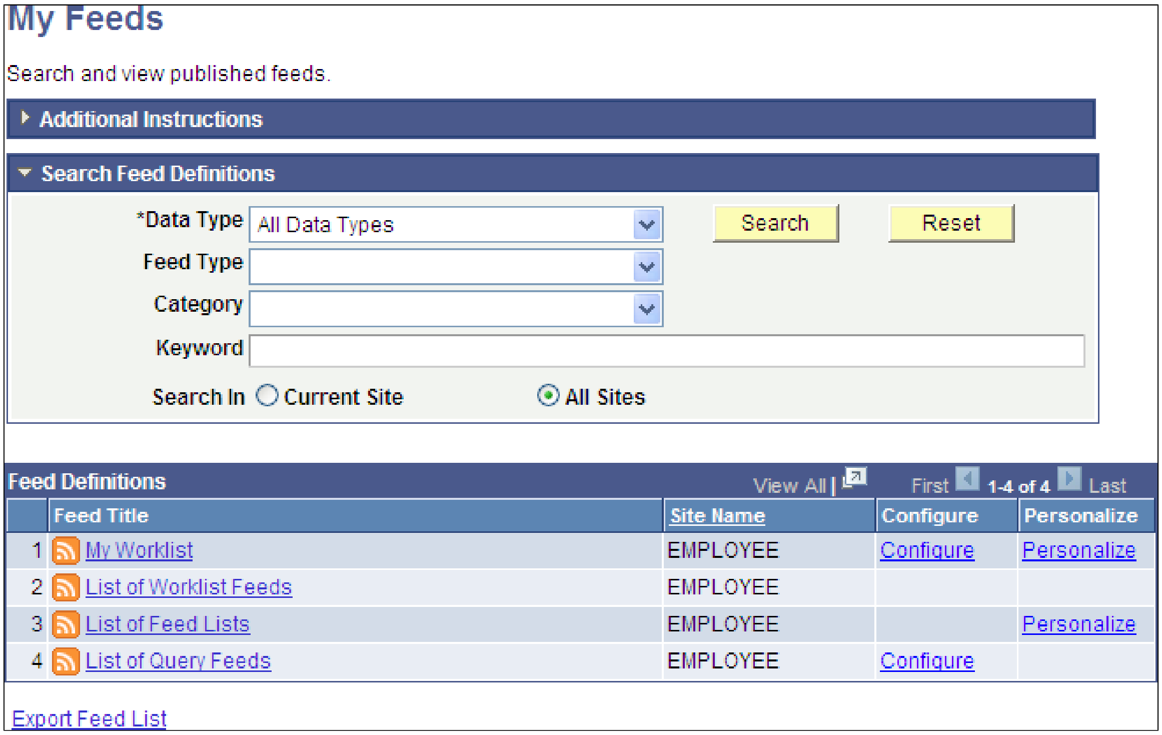
This section discusses how to access feeds.

Using the My Feeds Page

Access the My Feeds page (click My Feeds in the Main Menu).

Image: My Feeds page

This example illustrates the fields and controls on the My Feeds page. You can find definitions for the fields and controls later on this page.



My Feeds

Search and view published feeds.

▸ Additional Instructions
▾ Search Feed Definitions
Search
Reset

*Data Type: All Data Types

Feed Type:

Category:

Keyword:

Search In ○ Current Site ● All Sites

Feed Definitions

View All | First 1-4 of 4 Last

	Feed Title	Site Name	Configure	Personalize
1	My Worklist	EMPLOYEE	Configure	Personalize
2	List of Worklist Feeds	EMPLOYEE		
3	List of Feed Lists	EMPLOYEE		Personalize
4	List of Query Feeds	EMPLOYEE	Configure	

Export Feed List

Use the My Feeds page to search for and view the list of published feeds to which you have access.

Search Feed Definitions

Data Type

Select from the following feed data types:

- *All Data Types* returns a list of all feeds to which you have access.

  Note: This value is the default.

- *IB Generic Message Feeds* returns a list of Integration Broker generic message feeds.
- *List of Feeds* returns a list of all feed list feeds.
- *PS Query Feeds* returns a list of PeopleSoft Query feeds.
- *Worklist Feeds* returns a list of worklist feeds.

Note: The previous list includes only the base feed data types delivered with PeopleTools. The results can differ on your system depending which other PeopleSoft applications you have installed and whether any custom feed data types have been developed.

Feed Type

Select from the following feed types:

- *Real Time* returns a list of real-time feeds.
- *Scheduled* returns a list of scheduled feeds.

Category

Select from the feed categories defined on your system.

Keyword

Enter a keyword to narrow the search criteria. Keywords search the *Feed ID, Feed Title,* and *Description* fields, and are *not* case sensitive.

Search In

Select from the following options to search for feeds:

- Current Site Select this option to search in the current site only.

  Note: This value is the default.

- All Sites Select this option to search every site for available feeds.

Search

Click the Search button to return the list of available feeds that satisfy the search criteria that you entered.

Reset

Click the Reset button to clear the search results and reset all search criteria to their default values.

Export Feed List

Click this link to export the current search results in the Feed Definitions grid to a file in OPML 2.0 format. You can import OPML files into third-party feed readers, and you will be able to subscribe to each feed in the list of feeds in the exported file.

This link is visible only when a search returns one or more values.

Feed Definitions

This grid lists the feeds that match the search criteria and that you have permission to access.

Feed Title

Click the link for a feed title to view the feed document in a separate browser window.

Site Name

Displays the site to which the feed has been published.

Note: The Site Name column appears when the All Sites option is selected.

Configure

Click a Configure link to configure the feed as an administrator for this site.

Note: The Configure link appears only for those feeds that have been defined in the Feed Publishing Wizard with *Admin Specified* data source parameters.

See Step 3: Specifying Data Source Parameters.

Personalize

Click a Personalize link to personalize the feed by specifying user personalization values.

Note: The Personalize link appears only for those feeds that have been defined in the Feed Publishing Wizard with *User Specified* data source parameters.

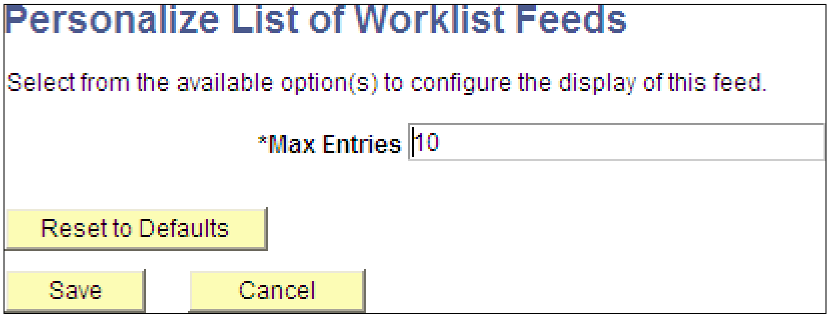
See Step 3: Specifying Data Source Parameters.

Personalizing a Feed

Access the Personalize feed page (click a Personalize link on the My Feeds page).

Image: Personalize feed page

This example illustrates the fields and controls on the Personalize List of Worklist Feeds page. You can find definitions for the fields and controls later on this page.



Personalize List of Worklist Feeds

Select from the available option(s) to configure the display of this feed.

*Max Entries: 10

Use the Personalize feed page to set user personalization values for the feed. Only data source parameters defined as *User Specified* in the feed definition will appear on this page.

Reset to Defaults

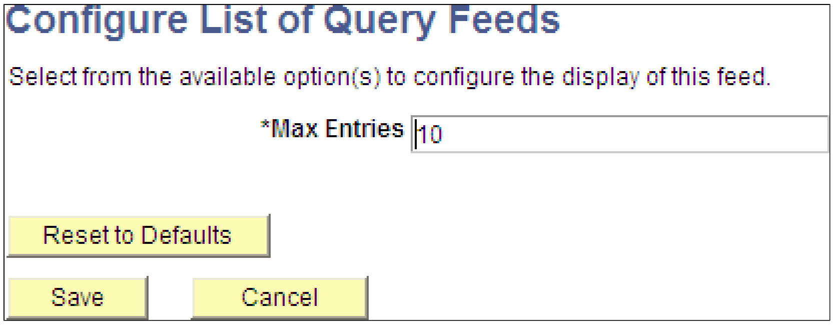
Click the Reset to Defaults button to reset all parameter values to their system default values stored with the feed definition.

Configuring a Feed

Access the Configure feed page (click a Configure link on the My Feeds page).

Image: Configure feed page

This example illustrates the fields and controls on the Configure List of Query Feeds page. You can find definitions for the fields and controls later on this page.



Configure List of Query Feeds

Select from the available option(s) to configure the display of this feed.

*Max Entries: 10

Use the Configure feed page to set administrator-specified values for the feed. Only data source parameters defined as *Admin Specified* in the feed definition will appear on this page.

Reset to Defaults

Click the Reset to Defaults button to reset all parameter values to their system default values stored with the feed definition.

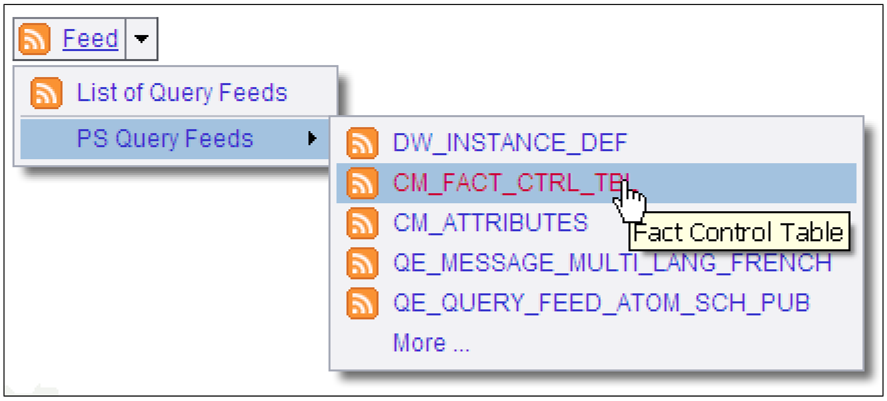
Using Related Feeds Hover Menus

You can view the feeds of a particular data type to which you have access through the related feeds hover menu on a page.

Notice the hover menu extending to show the query feeds that are available to this user:

Image: An extended related feeds hover menu

This example illustrates the extended related feeds hover menu.



Note: The hover menu is inactive when no feeds have been published for that particular feed data type. To hide inactive hover menus, set a value on the Feed Options page.

Related Links

Setting Feed Publishing Framework Options

Using the GetFeedList Service Operation

The GetFeedList service operation (PTFP_GETFEEDLIST) returns a list of feeds managed by the Feed Publishing Framework using the OPML format. Users can save this file to their desktop and share the file with other users who might want to access the same list. You can also use this file to import feeds into a feed reader.

In the file, feeds are organized in folders based on their feed data type. Only feeds visible to authenticated user of the request are returned. The GetFeedList operation provides the capability to query feeds through the use of operational query parameters, the same way as the My Feeds component. This table provides Some examples of the parameters:

Parameter	***Description***	***Example***
PTFP_DATA_TYPE	The data type being queried.	To search for the PS Query feed type, use this parameter: PTFP_DATA_TYPE=PSQUERY.
PTFP_FEED_KEYWORD	The keyword being used in the query. The keyword searches the feed title, description, and ID.	To search feeds whose name contains a *User* string, use this parameter: PTFP_FEED_KEYWORD=User.
PTFP_DSS_COUNT	The data source setting name/value pair parameter count.	To specify the query string when it contains one pair of the data source setting name and value, use this parameter: PTFP_DSS_COUNT=1.
PTFP_DSS_NAMEn	The data source setting name, where *n* is an integer beginning with 1.	To specify the first data source setting name, use this parameter: PTFP_DSS_NAME1=QRYNAME.
PTFP_DSS_VALUEn	The data source setting value, where *n* is an integer beginning with 1.	To specify the first data source setting value, use this parameter:PTFP_DSS_VALUE1=MESSAGES_FOR_MSGSET.
PORTAL_NAME	The portal being searched if you have multiple portals. The system searches all sites if this parameter is not specified.	To search only the EMPLOYEE portal, use this parameter: PORTAL_NAME=EMPLOYEE.
LANGUAGE_CD	The language of the feed being queried.	To search the feed in Canadian French, use this parameter: LANGUAGE_CD=CFR.

Note: PeopleTools provides one GetFeedList service. However, you can develop your own services for specific feed data types.

Example of .opml File Exported From the My Feeds Page

OPML refers to Outline Processor Markup Language. It is the protocol used for exchanging feed lists between feed readers and aggregators. This is an example of an opml file:

```
<?xml version="1.0"?>
<opml version="2.0">
  <head>
    <dateCreated>Mon, 04 May 2009 16:32:27 GMT</dateCreated>
    <dateModified>Mon, 04 May 2009 16:32:27 GMT</dateModified>
    <ownerName>QE User</ownerName>
    <title>Feed Search Results</title>
  </head>
  <body>
    <outline category="IB Generic Message Feeds" description="Generic Operation" te⇒
xt="IB_GENERIC" title="ADMN_IB_GENERIC" type="rss" xmlUrl="http://xxxxx.xx.oracle.c⇒
om:8920/PSIGW/HttpListeningConnector/feeds/GetScheduledFeed?FeedID=IB_GENERIC.V1&am⇒
p;ChildFeedID=ADMN_IB_GENERIC&amp;S=P&amp;PORTAL_NAME=EMPLOYEE&amp;NODE_NAME=QE_LOC⇒
AL"/>
    <outline category="PS Query Feeds" description="Dimension Control Table" text="⇒
CM_DIM_CTRL_TBL" title="ADMN_CM_DIM_CTRL_TBL" type="rss" xmlUrl="http://xxxxx.xx.or⇒
acle.com:8920/PSIGW/HttpListeningConnector/feeds/RealtimeQueryFeed?FEED_ID=ADMN_CM_⇒
DIM_CTRL_TBL&amp;PORTAL_NAME=EMPLOYEE&amp;NODE_NAME=QE_LOCAL"/>
    <outline category="PS Query Feeds" description="Message Set" text="CD_MSGSET" t⇒
itle="ADMN_CD_MSGSET1" type="rss" xmlUrl="http://xxxxx.xx.oracle.com:8920/PSIGW/Htt⇒
pListeningConnector/feeds/RealtimeQueryFeed?FEED_ID=ADMN_CD_MSGSET1&amp;PORTAL_NAME⇒
=EMPLOYEE&amp;NODE_NAME=QE_LOCAL"/>
    <outline category="IB Generic Message Feeds" description="User Profile" text="U⇒
ser Profile Feed" title="ADMN_USER_PROFILE" type="rss" xmlUrl="http://xxxxx.xx.orac⇒
le.com:8920/PSIGW/HttpListeningConnector/feeds/GetScheduledFeed?FeedID=USER_PROFILE⇒
.VERSION_84&amp;ChildFeedID=ADMN_USER_PROFILE&amp;PORTAL_NAME=EMPLOYEE&amp;NODE_NAM⇒
E=QE_LOCAL"/>
    <outline category="List of Feeds" description="A feed of this data type contain⇒
s data of the specified PS Query." text="PS Query Feeds" title="ADMN_PS_QUERY_FEEDS⇒
" type="rss" xmlUrl="http://xxxxx.xx.oracle.com:8920/PSIGW/HttpListeningConnector/f⇒
eeds/GetRealTimeFeed?FEED_ID=ADMN_PS_QUERY_FEEDS&amp;S=P&amp;PORTAL_NAME=EMPLOYEE&a⇒
mp;NODE_NAME=QE_LOCAL"/>
    <outline category="List of Feeds" description="A feed of this data type contain⇒
s worklist items of the current user." text="Worklist Feeds" title="ADMN_WORKLIST_F⇒
EEDS" type="rss" xmlUrl="http://xxxxx.xx.oracle.com:8920/PSIGW/HttpListeningConnect⇒
or/feeds/GetRealTimeFeed?FEED_ID=ADMN_WORKLIST_FEEDS&amp;S=P&amp;PORTAL_NAME=EMPLOY⇒
EE&amp;NODE_NAME=QE_LOCAL"/>
    <outline category="List of Feeds" description="A feed of this data type contain⇒
s list of published feeds of the specified data type." text="List of Feeds" title="⇒
ADMN_LIST_OF_FEEDS" type="rss" xmlUrl="http://xxxxx.xx.oracle.com:8920/PSIGW/HttpLi⇒
steningConnector/feeds/GetRealTimeFeed?FEED_ID=ADMN_LIST_OF_FEEDS&amp;S=P&amp;PORTA⇒
L_NAME=EMPLOYEE&amp;NODE_NAME=QE_LOCAL"/>
    <outline category="PS Query Feeds" description="User ID's Access Activity" text⇒
="User Access Activity" title="ADMN_PT_SEC_ACCESSLOG_USER" type="rss" xmlUrl="http:⇒
//xxxxx.xx.oracle.com:8920/PSIGW/HttpListeningConnector/feeds/RealtimeQueryFeed?FEE⇒
D_ID=ADMN_PT_SEC_ACCESSLOG_USER&amp;PORTAL_NAME=EMPLOYEE&amp;NODE_NAME=QE_LOCAL"/>
    <outline category="PS Query Feeds" description="Hierarchy Map Table" text="CM_H⇒
IER_MAP_TBL" title="ADMN_CM_HIER_MAP_TBL" type="rss" xmlUrl="http://xxxxx.xx.oracle⇒
.com:8920/PSIGW/HttpListeningConnector/feeds/RealtimeQueryFeed?FEED_ID=ADMN_CM_HIER⇒
_MAP_TBL&amp;PORTAL_NAME=EMPLOYEE&amp;NODE_NAME=QE_LOCAL"/>
    <outline category="PS Query Feeds" description="Attribute mappings" text="CM_AT⇒
TRIBUTES" title="ADMN_CM_ATTRIBUTES" type="rss" xmlUrl="http://xxxxx.xx.oracle.com:⇒
8920/PSIGW/HttpListeningConnector/feeds/RealtimeQueryFeed?FEED_ID=ADMN_CM_ATTRIBUTE⇒
S&amp;PORTAL_NAME=EMPLOYEE&amp;NODE_NAME=QE_LOCAL"/>
    <outline category="Worklist Feeds" description="Worklist notifications" text="W⇒
orklist Notification" title="ADMN_WORKLIST_ITEMS" type="rss" xmlUrl="http://xxxxx.x⇒
x.oracle.com:8920/PSIGW/HttpListeningConnector/feeds/GetRealTimeFeed?FEED_ID=ADMN_W⇒
ORKLIST_ITEMS&amp;PORTAL_NAME=EMPLOYEE&amp;NODE_NAME=QE_LOCAL"/>
  </body>
</opml>
```

Viewing Feeds

You can view feeds directly in the browser or by using a third-party feed reader program.

To view feeds:

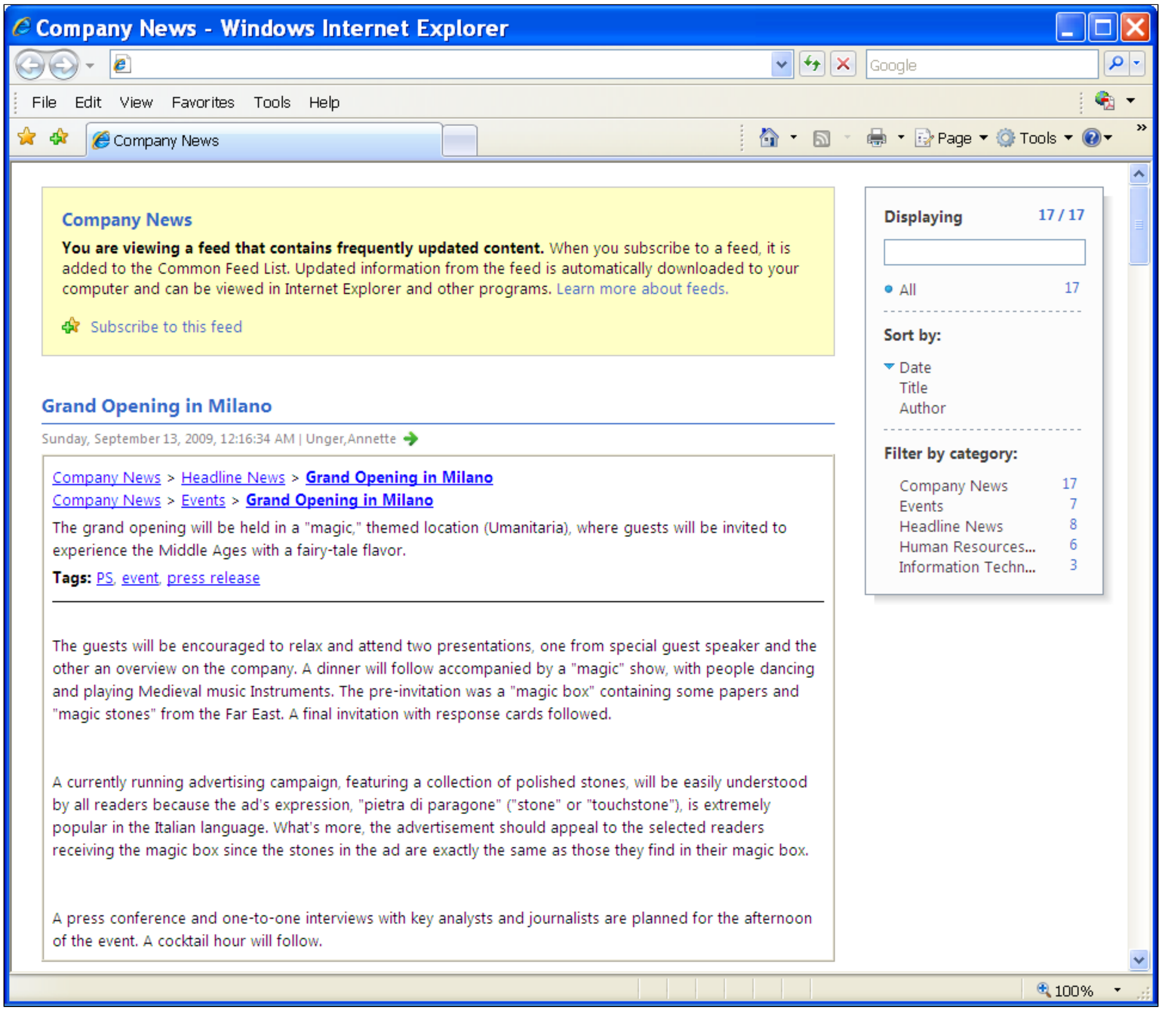
1. Click the feed link.

   The feed will open in the browser.

2. Alternatively, paste the URL into a feed reader.

Image: Company News feed in the browser

This example illustrates a company news feed in the browser.



Company News

You are viewing a feed that contains frequently updated content. When you subscribe to a feed, it is added to the Common Feed List. Updated information from the feed is automatically downloaded to your computer and can be viewed in Internet Explorer and other programs. Learn more about feeds.

Subscribe to this feed

Grand Opening in Milano

Sunday, September 13, 2009, 12:16:34 AM | Unger,Annette

Company News > Headline News > **Grand Opening in Milano**
Company News > Events > **Grand Opening in Milano**

The grand opening will be held in a "magic," themed location (Umanitaria), where guests will be invited to experience the Middle Ages with a fairy-tale flavor.

Tags: PS, event, press release

The guests will be encouraged to relax and attend two presentations, one from special guest speaker and the other an overview on the company. A dinner will follow accompanied by a "magic" show, with people dancing and playing Medieval music Instruments. The pre-invitation was a "magic box" containing some papers and "magic stones" from the Far East. A final invitation with response cards followed.

A currently running advertising campaign, featuring a collection of polished stones, will be easily understood by all readers because the ad's expression, "pietra di paragone" ("stone" or "touchstone"), is extremely popular in the Italian language. What's more, the advertisement should appeal to the selected readers receiving the magic box since the stones in the ad are exactly the same as those they find in their magic box.

A press conference and one-to-one interviews with key analysts and journalists are planned for the afternoon of the event. A cocktail hour will follow.

Displaying 17 / 17

- All 17

Sort by:

Date
Title
Author

Filter by category:

Company News	17
Events	7
Headline News	8
Human Resources...	6
Information Techn...	3

Chapter 5

Administering Feeds

Administering the Feed Publishing Framework

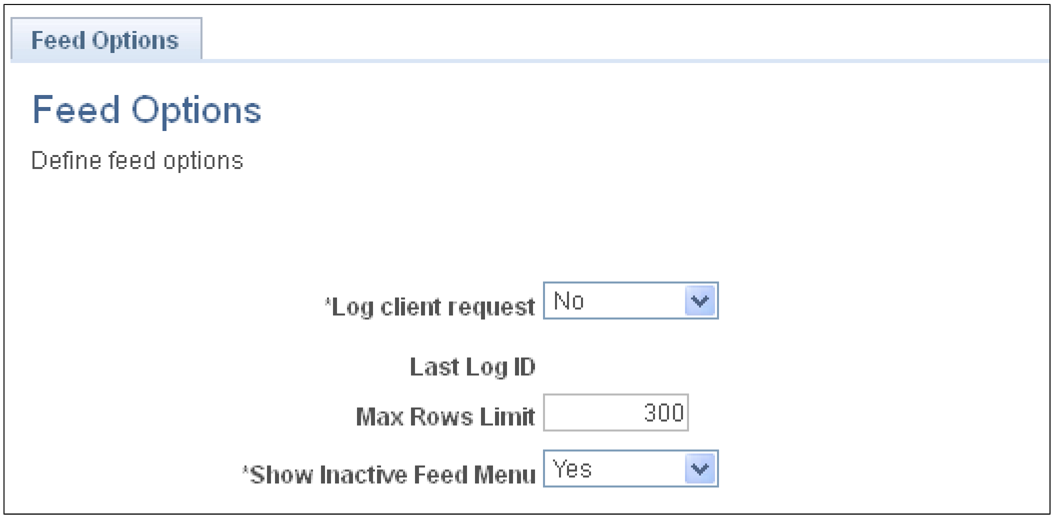
This section discusses how to administer the feed publishing framework.

Setting Feed Publishing Framework Options

Access the Feed Options page (PeopleTools, Feeds, Feed Options).

Image: Feed Options page

This example illustrates the fields and controls on the Feed Options page. You can find definitions for the fields and controls later on this page.



Feed Options

Define feed options

*Log client request: No

Last Log ID

Max Rows Limit: 300

*Show Inactive Feed Menu: Yes

Use the Feed Options page to define system-wide options for the Feed Publishing Framework.

Log client request

Select *Yes* to enable logging of feed requests in the PTFP_ ACCESS_LOG table. The default value is *No.*

Last Log ID

Displays the log ID of the last feed request.

Max Rows Limit

Specify the limit for the maximum number of feed entries in a feed. This limit applies to all feed types except paged scheduled feeds.

Show Inactive Feed Menu

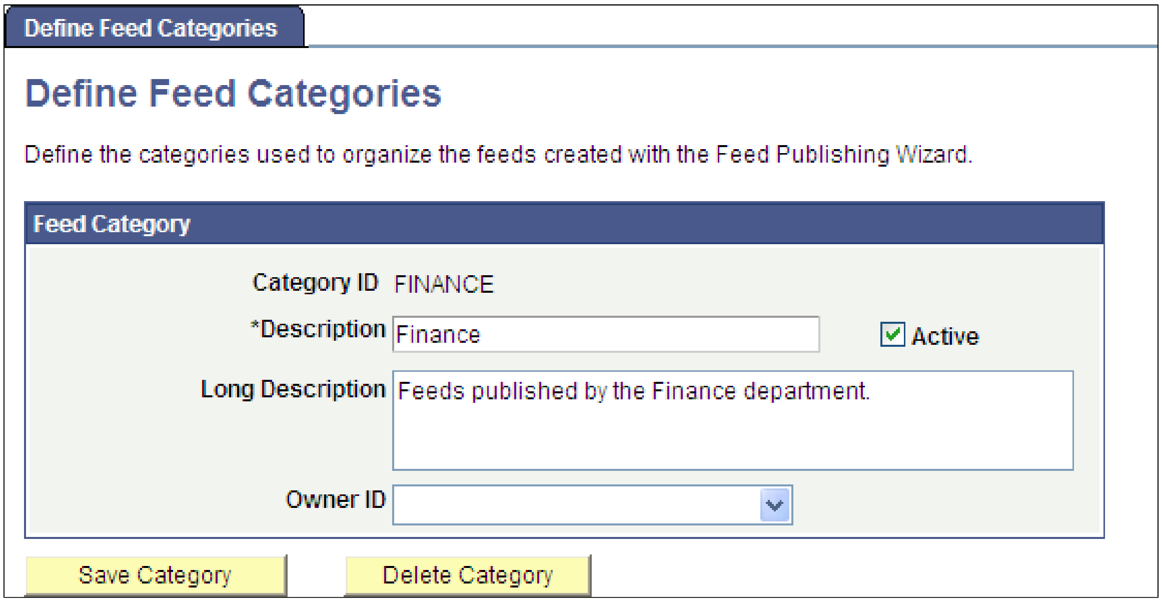
Select *No* to hide the related feeds hover menu when it is inactive—that is, when no feeds of that type have been defined. The default value is *Yes,* show the related feeds hover menu when it is inactive.

Defining Feed Categories

Access the Define Feed Categories page (PeopleTools, Feeds, Define Feed Categories).

Image: Define Feed Categories page

This example illustrates the fields and controls on the Define Feed Categories page. You can find definitions for the fields and controls later on this page.



Define Feed Categories

Define the categories used to organize the feeds created with the Feed Publishing Wizard.

Feed Category

Category ID	FINANCE
*Description	Finance ☑ Active
Long Description	Feeds published by the Finance department.
Owner ID	

Use the Define Feed Categories page to manage categories used to organize feeds.

Category ID

Specify a unique ID for the feed category.

Note: The category ID is a user-specified, not system-defined, value.

Description

Enter a short description for the feed category.

This description appears wherever a category can be assigned to a feed definition and in locations where you can search for feeds.

Active

Indicate whether the feed category is active or inactive.

An inactive category is not available to assign to feed definitions or in locations where you can search for feeds. However, if a feed definition is assigned to a category and then that category is made inactive, the feed definition no longer displays an assigned category.

Long Description

Enter a long description for the feed category.

Owner ID

Select the owner ID for the feed category.

The owner ID is a way to identify which definitions are owned by which PeopleSoft applications, such as PeopleSoft General Ledger, Accounts Receivables, PeopleTools, and so on. The

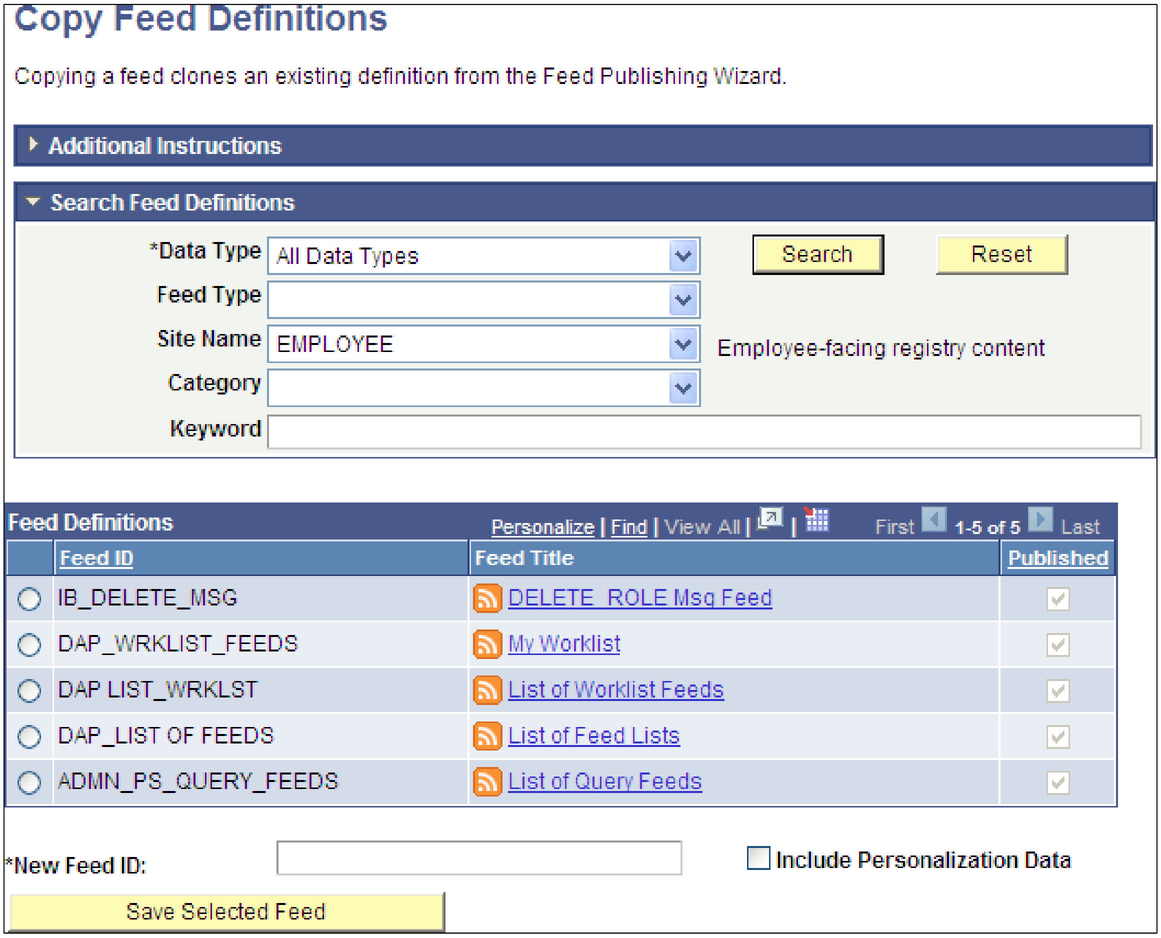
	values in the drop-down list box are Translate table values associated with the OBJECTOWNERID field.
Save Category	Click to save the feed category.
Delete Category	Click to delete the feed category. If a feed definition is assigned to a category and then that category is deleted, the feed definition no longer displays an assigned category.

Copying Feed Definitions

Access the Copy Feed Definitions page (PeopleTools, Feeds, Copy Feed Definitions).

Image: Copy Feed Definitions page

This example illustrates the fields and controls on the Copy Feed Definitions page. You can find definitions for the fields and controls later on this page.



Copy Feed Definitions

Copying a feed clones an existing definition from the Feed Publishing Wizard.

▸ Additional Instructions

▾ Search Feed Definitions

*Data Type	All Data Types	
Feed Type		
Site Name	EMPLOYEE	Employee-facing registry content
Category		
Keyword		

Feed Definitions

	Feed ID	Feed Title	Published
○	IB_DELETE_MSG	DELETE_ROLE Msg Feed	✓
○	DAP_WRKLIST_FEEDS	My Worklist	✓
○	DAP LIST_WRKLST	List of Worklist Feeds	✓
○	DAP_LIST OF FEEDS	List of Feed Lists	✓
○	ADMN_PS_QUERY_FEEDS	List of Query Feeds	✓

*New Feed ID:

□ Include Personalization Data

Use the Copy Feed Definitions page to clone an existing feed definition to one with a new feed ID. The new feed definition is published with the same feed options and to the same sites as the original.

Data Type	Select from the following feed data types:

- *All Data Types* returns a list of all feeds to which you have access.

  Note: This value is the default.

- *IB Generic Message Feeds* returns a list of Integration Broker generic message feeds.
- *List of Feeds* returns a list of all feed list feeds.
- *PS Query Feeds* returns a list of PeopleSoft Query feeds.
- *Worklist Feeds* returns a list of worklist feeds.

Note: The previous list includes only the base feed data types delivered with PeopleTools. The results can differ on your system depending which other PeopleSoft applications you have installed and whether any custom feed data types have been developed.

Feed Type

Select from the following feed types:

- *Real Time* returns a list of real-time feeds.
- *Scheduled* returns a list of scheduled feeds.

Site Name

Select from the sites available on your system.

This field does not filter the search results; the results indicate whether a feed definition is published to the selected site.

Category

Select from the active feed categories defined on your system.

Keyword

Enter a keyword to narrow the search criteria. Keywords search the Feed ID, Feed Title, and Description fields and are *not* case sensitive.

Search

Click the Search button to return the list of available feeds that satisfy the search criteria that you entered.

Reset

Click the Reset button to clear the search results and reset all search criteria to their default values.

Feed Definitions

Select the feed from this grid that you wish to copy.

Note: The Published field indicates that the feed definition is published to the site selected in the Site Name field.

New Feed ID

Enter an ID for the new feed definition using only alphabetic characters (A–Z), numeric characters (0–9), and the underscore character (_).

Note: The system converts all lowercase characters to uppercase. In addition, it will convert any other special characters, such as a space, #, @, and so on, to an underscore.

Include Personalization Data

Select this option to copy user-specified and administrator-specified data source parameter values along with the feed definition.

Save Selected Feed

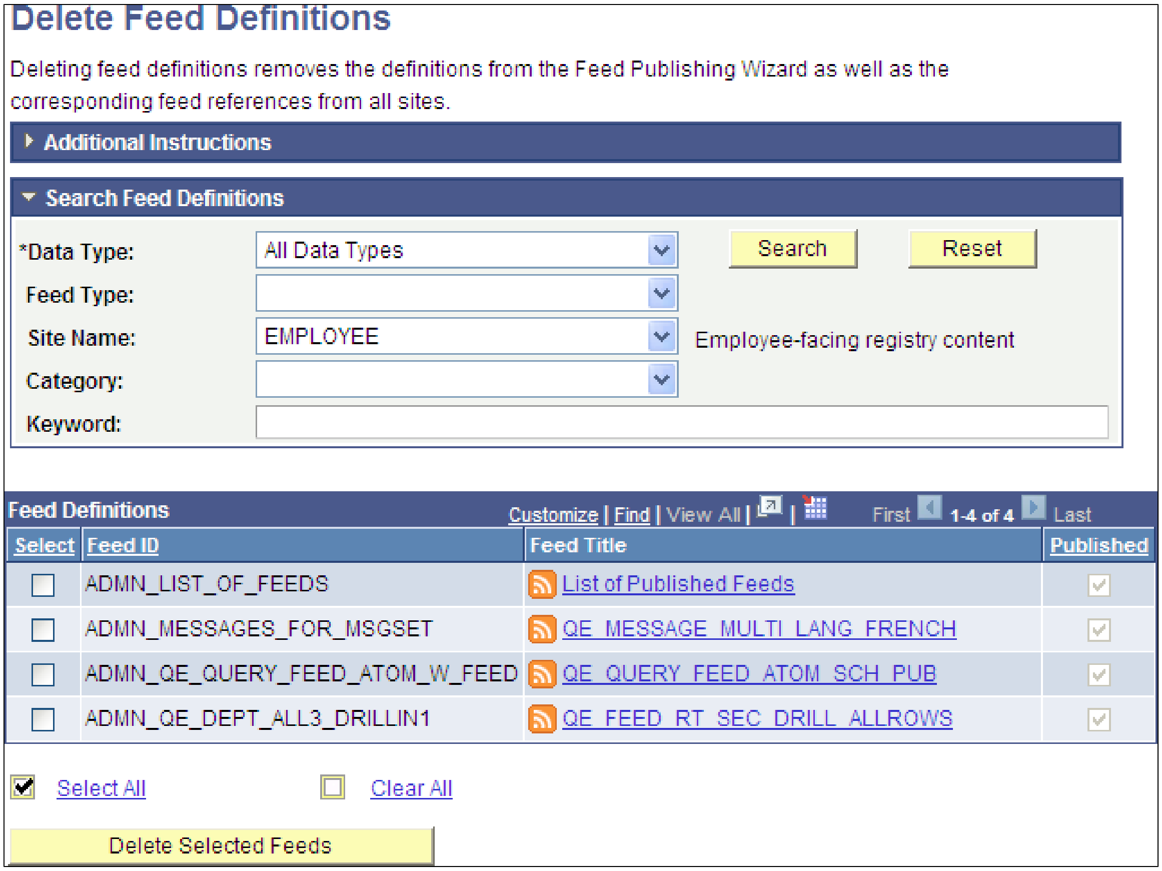
Click to copy the selected feed definition to a new feed definition.

Deleting Feed Definitions

Access the Delete Feed Definitions page (PeopleTools, Feeds, Delete Feed Definitions).

Image: Delete Feed Definitions page

This example illustrates the fields and controls on the Delete Feed Definitions page. You can find definitions for the fields and controls later on this page.



Delete Feed Definitions

Deleting feed definitions removes the definitions from the Feed Publishing Wizard as well as the corresponding feed references from all sites.

▸ Additional Instructions

▾ Search Feed Definitions

*Data Type:	All Data Types	Search Reset
Feed Type:		
Site Name:	EMPLOYEE	Employee-facing registry content
Category:		
Keyword:		

Feed Definitions — Customize | Find | View All | First 1-4 of 4 Last

Select	Feed ID	Feed Title	Published
□	ADMN_LIST_OF_FEEDS	List of Published Feeds	✓
□	ADMN_MESSAGES_FOR_MSGSET	QE_MESSAGE_MULTI_LANG_FRENCH	✓
□	ADMN_QE_QUERY_FEED_ATOM_W_FEED	QE_QUERY_FEED_ATOM_SCH_PUB	✓
□	ADMN_QE_DEPT_ALL3_DRILLIN1	QE_FEED_RT_SEC_DRILL_ALLROWS	✓

☑ Select All □ Clear All

Delete Selected Feeds

Use the Delete Feed Definitions page to delete selected feed definitions. The system deletes the feed definition from all sites to which it has been published.

Data Type

Select from the following feed data types:

- *All Data Types* returns a list of all feeds to which you have access.

Note: This value is the default.

- *IB Generic Message Feeds* returns a list of Integration Broker generic message feeds.
- *List of Feeds* returns a list of all feed list feeds.
- *PS Query Feeds* returns a list of PeopleSoft Query feeds.
- *Worklist Feeds* returns a list of worklist feeds.

Note: The previous list includes only the base feed data types delivered with PeopleTools. The results can differ on your system depending which other PeopleSoft applications you have installed and whether any custom feed data types have been developed.

Feed Type

Select from the following feed types:

- *Real Time* returns a list of real-time feeds.
- *Scheduled* returns a list of scheduled feeds.

Site Name

Select from the sites available on your system.

This field does not filter the search results; the results indicate whether a feed definition is published to the selected site.

Category

Select from the feed categories defined on your system.

Keyword

Enter a keyword to narrow the search criteria. Keywords search the Feed ID, Feed Title, and Description fields and are *not* case sensitive.

Search

Click the Search button to return the list of available feeds that satisfy the search criteria that you entered.

Reset

Click the Reset button to clear the search results and reset all search criteria to their default values.

Feed Definitions

Select the feed definition or definitions from this grid that you want to delete.

Note: The Published field indicates that the feed definition is published to the site selected in the Site Name field.

☑ Select All

Click to select all feed definitions in the list.

□ Clear All

Click to clear the selection of any feed definitions.

Delete Selected Feeds

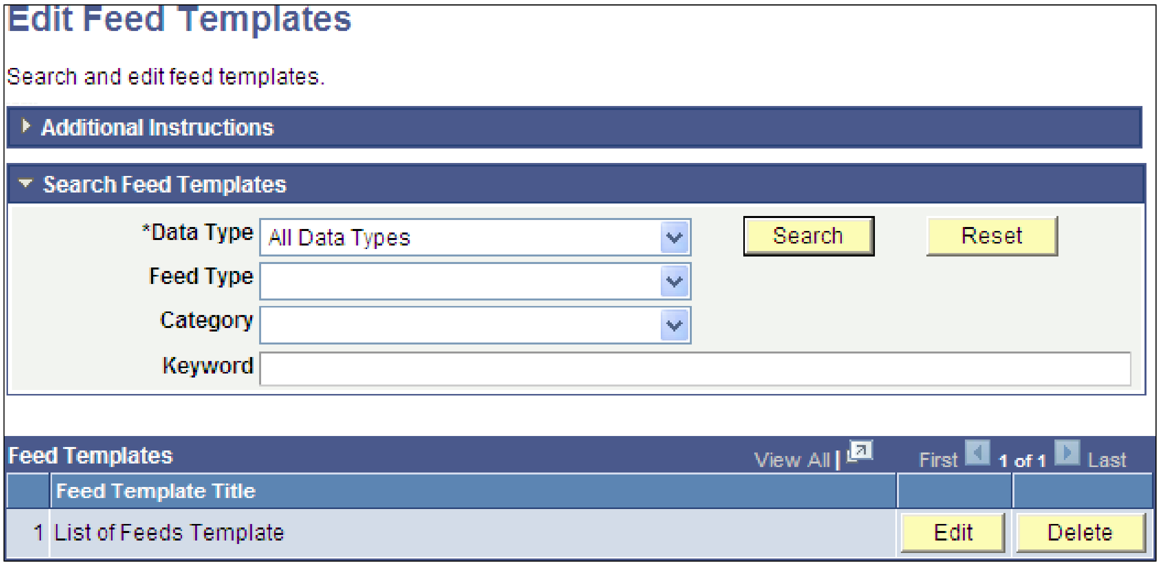
Click to delete the selected feed definition or definitions.

Editing Feed Templates

Access the Edit Feed Templates page (PeopleTools, Feeds, Edit Feed Templates).

Image: Edit Feed Templates page

This example illustrates the fields and controls on the Edit Feed Templates page. You can find definitions for the fields and controls later on this page.



Edit Feed Templates

Search and edit feed templates.

*Data Type: All Data Types

Feed Type:

Category:

Keyword:

Feed Templates

	Feed Template Title		
1	List of Feeds Template	Edit	Delete

Use the Edit Feed Templates page to edit or delete feed templates.

Data Type

Select from the following feed data types:

- *All Data Types* returns a list of all feed templates to which you have access.

  Note: This value is the default.

- *IB Generic Message Feeds* returns a list of Integration Broker generic message feed templates.
- *List of Feeds* returns a list of all feed list feed templates.
- *PS Query Feeds* returns a list of PeopleSoft Query feed templates.
- *Worklist Feeds* returns a list of worklist feed templates.

Note: The previous list includes only the base feed data types delivered with PeopleTools. The results can differ on your system depending which other PeopleSoft applications you have installed and whether any custom feed data types have been developed.

Feed Type

Select from the following feed types:

- *Real Time* returns a list of real-time feed templates.

- *Scheduled* returns a list of scheduled feed templates.

Category

Select from the active feed categories defined on your system.

Keyword

Enter a keyword to narrow the search criteria. Keywords search the Feed ID, Feed Title, and Description fields and are *not* case sensitive.

Search

Click the Search button to return the list of available feed templates that satisfy the search criteria that you entered.

Reset

Click the Reset button to clear the search results and reset all search criteria to their default values.

Edit

Click the Edit button to edit the feed template in the Feed Publishing Wizard.

See Step 1: Specifying Feed Properties.

Delete

Click to delete the feed template.

Converting Feed Metadata Between PeopleTools Releases

This section discusses how to:

- Convert feed metadata from PeopleTools 8.50 or 8.51 to the current release (ad hoc conversions).
- Convert feed metadata from the current release to PeopleTools 8.50 or 8.51.

Note: Starting with PeopleTools 8.52, feed definitions, feed templates, feed categories, and feed data types can be managed as PeopleTools managed objects. This enhancement required a change to the data structures that store feed metadata. Starting with PeopleTools 8.52, new tables were introduced to store feed metadata—specifically, tables with a PSFP_ prefix. In addition, the pre-PeopleTools 8.52 tables (that is, tables with a PS_PTFP_ prefix) are now used as staging tables for importing and converting metadata to the PeopleTools 8.52 format. Feed Metadata Table Structures provides a list showing the mapping of pre-8.52 tables to their current equivalents.
See Feed Metadata Tables.

Converting Feed Metadata from PeopleTools 8.50 or 8.51 to the Current Release (Ad Hoc Conversions)

This section provides an overview of using the Metadata Conversion page and discusses how to:

- Convert feed definitions to the current release.
- Convert feed data types to the current release.
- Convert feed categories to the current release.

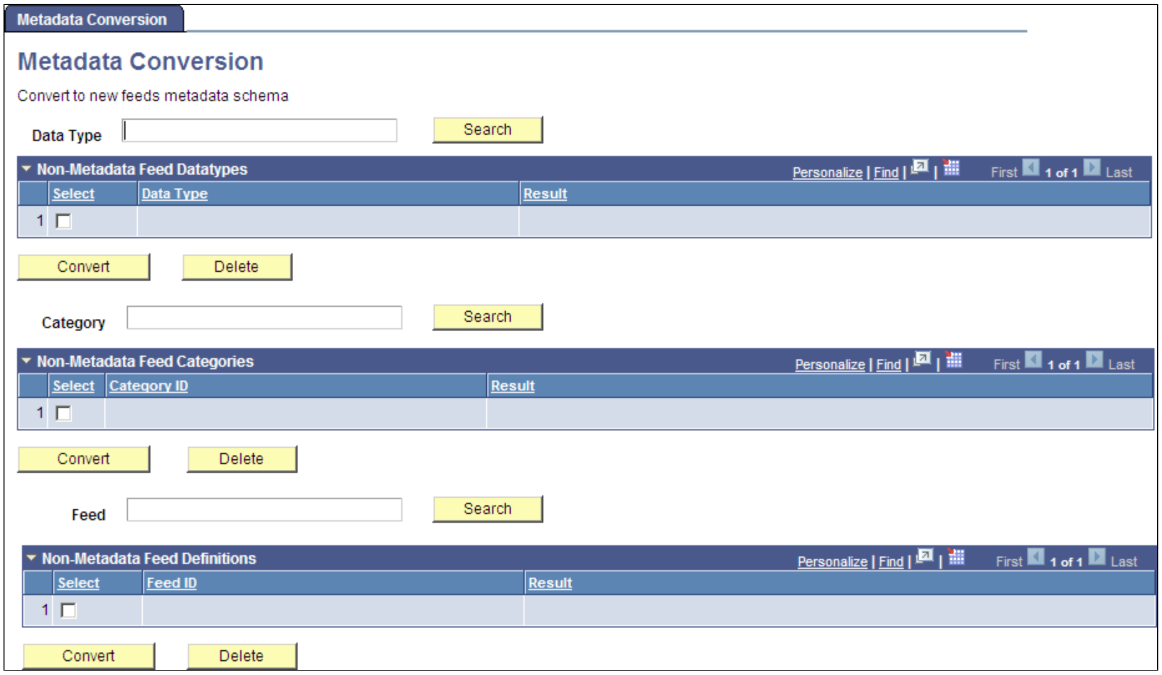
The PeopleTools upgrade process automatically converts existing feed metadata to the format introduced in PeopleTools 8.52. Therefore, if you have upgraded your PeopleSoft system from a previous release, then you do not need to complete any of these ad hoc conversions. However, if you have feed definitions, feed data types, or feed categories in a PeopleSoft system that has not been upgraded and want those feed objects to be migrated to an instance on the current release, then you can use the Metadata Conversion page and the procedures in this section to convert that metadata in an ad hoc manner to the PeopleTools 8.52 format.

Overview of Using the Metadata Conversion Page

Access the Metadata Conversion page (PeopleTools, Feeds, Feed Metadata Conversion).

Image: Metadata Conversion page

This example illustrates the fields and controls on the Metadata Conversion page. You can find definitions for the fields and controls later on this page.



	Select	Data Type	Result
1	☐		

	Select	Category ID	Result
1	☐		

	Select	Feed ID	Result
1	☐		

Use the Metadata Conversion page to convert feed metadata from a previous release (PeopleTools 8.50 or 8.51) to the current release. Each object type (feed definition, feed data type, and feed category) can be loaded and converted separately of the other object types.

Data Type

Enter any text to search for and retrieve all imported feed data types matching the search string in its initial characters. The following search strings each retrieve the data type named ADMN_TYPE:

```
AD
AD%
%MN
```

If no search string is specified, then all imported feed data types are retrieved.

Category

Enter any text to search for and retrieve all imported feed categories matching the search string in its initial characters.

The following search strings each retrieve the category named FINANCE:

```
FI
FI%
%NA
```

If no search string is specified, then all imported feed categories are retrieved.

Feed

Enter any text to search for and retrieve all imported feed definitions matching the search string in its initial characters. The following search strings each retrieve the feed definition named ADMN_WORKLIST1:

```
AD
AD%
%WORKLIST
```

If no search string is specified, then all imported feed definitions are retrieved.

Search

Click the Search button to search for imported objects matching the search criteria. If no search string is specified, then all imported objects of that type are retrieved.

Select

Select an object to convert or delete.

Convert

Click the Convert button to converted the selected objects to the PeopleTools 8.52 format.

Clicking the Convert button converts the data only and doesn't delete data from the staging tables (the PS_PTFP_ tables). Use the Delete button to manually delete data from the staging tables.

Important! If an object with the same ID already exists in the database, then a new version of that object is created, which becomes the current version of the object.

Delete

Click the Delete button to remove the selected objects from the pre-PeopleTools 8.52 staging tables—that is, from the tables with a PS_PTFP_ prefix.

See Feed Metadata Tables.

Converting Feed Definitions to the Current Release

Converting feed definitions from PeopleTools 8.50 or 8.51 to the current release consists of three major steps:

- Exporting the feed definitions from the source system.
- Importing the feed definitions into the PeopleTools 8.52 (or later) system.

- Converting the metadata to the PeopleTools 8.52 format.

To export the feed definitions from the source system, retrieve and run the appropriate Data Mover script:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Utilities > Administration > Message Catalog.
2. Enter *219* in the Message Set Number field and press the ENTER key.
3. Click the Find link and enter one of the following values in the Enter Search String field:
   - Enter *3300* to retrieve the script for exporting feed definitions *without* user personalization data.
   - Enter *3302* to retrieve the script for exporting feed definitions *with* user personalization data.
4. Click the OK button
5. Copy the script from the Description field.
6. Run PeopleSoft Data Mover on the source database.
7. Paste the script into the upper pane.
8. Modify the script as follows:
   - Change all occurrences of %3 to a value for this export of data—for example:

     ```
     PRODSYS01
     ```

   - Provide a complete path to the .DAT and .LOG files as needed.
   - Change all occurrences of %4 to identify the specific feed ID or IDs that you want to export—for example:

     ```
     'ADMN_WORKLIST_01'
     'ADMN_WORKLIST_01', 'ADMN_WORKLIST_02'
     'ADMN_WORKLIST%', 'ADMN_MY_%'
     ```

   - Optionally, replace %1 and %2 with values to provide a timestamp and the user ID who performed the conversion.
9. Run the script.

To import the feed definitions into the PeopleTools 8.52 (or later) system, retrieve and run the appropriate Data Mover script:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Utilities > Administration > Message Catalog.
2. Enter *219* in the Message Set Number field and press the ENTER key.
3. Click the Find link and enter one of the following values in the Enter Search String field:
   - Enter *3301* to retrieve the script for importing feed definitions *without* user personalization data. Use this script if you selected *3300* in the previous step 3.

- Enter *3303* to retrieve the script for importing feed definitions *with* user personalization data. Use this script if you selected *3302* in the previous step 3.

4. Click the OK button
5. Copy the script from the Description field.
6. Run PeopleSoft Data Mover on the PeopleTools 8.52 (or later) database.
7. Paste the script into the upper pane.
8. Modify the script as follows:

   - Change all occurrences of %3 to the value that you specified for %3 in the previous step 8—for example:

     ```
     PRODSYS01
     ```

   - Provide a complete path to the .DAT and .LOG files as needed.
   - Change all occurrences of %4 to the value that you specified for %4 in the previous step 8—for example:

     ```
     'ADMN_WORKLIST_01'
     'ADMN_WORKLIST_01', 'ADMN_WORKLIST_02'
     'ADMN_WORKLIST%', 'ADMN_MY_%'
     ```

   - Optionally, replace %1 and %2 with values to provide a timestamp and the user ID who performed the conversion.

9. Run the script.

To convert the metadata to the PeopleTools 8.52 format:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Feeds > Feed Metadata Conversion.
2. Click the Search button next to the text field labeled Feed to retrieve all imported feed definitions. Alternatively, enter the initial letters of the feed IDs for the feed definitions that you want to retrieve in the Feed text field and click Search.

   The Non-Metadata Feed Definitions grid is loaded with imported feed definitions.

3. Click in the Select column to select each of the feed definitions that you want to convert.
4. Click the Convert button.
5. To verify that the selected feed definitions were converted to the PeopleTools 8.52 format, go to the My Feeds page.
6. Click the Search button to list all feed definitions. Alternatively, enter search criteria to retrieve selected feeds only.

Converting Feed Data Types to the Current Release

Converting feed data types from PeopleTools 8.50 or 8.51 to the current release consists of three major steps:

- Exporting the feed data types from the source system.
- Importing the feed data types into the PeopleTools 8.52 (or later) system.
- Converting the metadata to the PeopleTools 8.52 format.

To export the feed data types from the source system, retrieve and run the appropriate Data Mover script:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Utilities > Administration > Message Catalog.
2. Enter *219* in the Message Set Number field and press the ENTER key.
3. Click the Find link and enter *3304* in the Enter Search String field
4. Click the OK button
5. Copy the script from the Description field.
6. Run PeopleSoft Data Mover on the source database.
7. Paste the script into the upper pane.
8. Modify the script as follows:
   - Change all occurrences of %3 to a value for this export of data—for example:

     ```
     PRODSYS01
     ```

   - Provide a complete path to the .DAT and .LOG files as needed.
   - Change all occurrences of %4 to identify the specific feed data type ID or IDs that you want to export—for example:

     ```
     'MY_DATATYPE_01'
     'MY_DATATYPE_01', 'MY_DATATYPE_02'
     '%FEED', 'MY_%'
     ```

   - Optionally, replace %1 and %2 with values to provide a timestamp and the user ID who performed the conversion.
9. Run the script.

To import the feed data types into the PeopleTools 8.52 (or later) system, retrieve and run the appropriate Data Mover script:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Utilities > Administration > Message Catalog.
2. Enter *219* in the Message Set Number field and press the ENTER key.
3. Click the Find link and enter *3305* in the Enter Search String field.

4. Click the OK button

5. Copy the script from the Description field.

6. Run PeopleSoft Data Mover on the PeopleTools 8.52 (or later) database.

7. Paste the script into the upper pane.

8. Modify the script as follows:

   - Change all occurrences of %3 to the value that you specified for %3 in the previous step 8—for example:

     ```
     PRODSYS01
     ```

   - Provide a complete path to the .DAT and .LOG files as needed.

   - Change all occurrences of %4 to the value that you specified for %4 in the previous step 8—for example:

     ```
     'MY_DATATYPE_01'
     'MY_DATATYPE_01', 'MY_DATATYPE_02'
     '%FEED', 'MY_%'
     ```

   - Optionally, replace %1 and %2 with values to provide a timestamp and the user ID who performed the conversion.

9. Run the script.

To convert the metadata to the PeopleTools 8.52 format:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Feeds >Feed Metadata Conversion.

2. Click the Search button next to the text field labeled Data Type to retrieve all imported feed data types. Alternatively, enter the initial letters of the feed data type IDs for the feed data types that you want to retrieve in the Data Type text field and click Search.

   The Non-Metadata Feed Datatypes grid is loaded with imported feed data types.

3. Click in the Select column to select each of the feed data types that you want to convert.

4. Click the Convert button.

5. To verify that the selected feed data types were converted to the PeopleTools 8.52 format, go to the Define Feed Data Types page (select PeopleTools, Feeds, Define Feed Data Types).

6. Click the Search button to list all feed data types.

Converting Feed Categories to the Current Release

Converting feed categories from PeopleTools 8.51 to the current release consists of three major steps:

- Exporting the feed categories from the source system.
- Importing the feed categories into the PeopleTools 8.52 (or later) system.

- Converting the metadata to the PeopleTools 8.52 format.

Note: Feed categories were introduced in PeopleTools 8.51. Therefore, these procedures apply to PeopleTools 8.51 only, and not to 8.50.

To export the feed categories from the source system, retrieve and run the appropriate Data Mover script:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Utilities > Administration > Message Catalog.
2. Enter *219* in the Message Set Number field and press the ENTER key.
3. Click the Find link and enter *3306* in the Enter Search String field
4. Click the OK button
5. Copy the script from the Description field.
6. Run PeopleSoft Data Mover on the source database.
7. Paste the script into the upper pane.
8. Modify the script as follows:
   - Change all occurrences of %3 to a value for this export of data—for example:

     ```
     PRODSYS01
     ```

   - Provide a complete path to the .DAT and .LOG files as needed.
   - Change all occurrences of %4 to identify the specific feed category ID or IDs that you want to export—for example:

     ```
     'MY_CATEGORY_01'
     'MY_CATEGORY_01', 'MY_CATEGORY_02'
     '%'
     ```

   - Optionally, replace %1 and %2 with values to provide a timestamp and the user ID who performed the conversion.
9. Run the script.

To import the feed categories into the PeopleTools 8.52 (or later) system, retrieve and run the appropriate Data Mover script:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Utilities > Administration > Message Catalog.
2. Enter *219* in the Message Set Number field and press the ENTER key.
3. Click the Find link and enter *3307* in the Enter Search String field.
4. Click the OK button
5. Copy the script from the Description field.
6. Run PeopleSoft Data Mover on the PeopleTools 8.52 (or later) database.

7. Paste the script into the upper pane.

8. Modify the script as follows:

    - Change all occurrences of %3 to the value that you specified for %3 in the previous step 8—for example:

      ```
      PRODSYS01
      ```

    - Provide a complete path to the .DAT and .LOG files as needed.

    - Change all occurrences of %4 to the value that you specified for %4 in the previous step 8—for example:

      ```
      'MY_CATEGORY_01'
      'MY_CATEGORY_01', 'MY_CATEGORY_02'
      '%'
      ```

    - Optionally, replace %1 and %2 with values to provide a timestamp and the user ID who performed the conversion.

9. Run the script.

To convert the metadata to the PeopleTools 8.52 format:

1. In the browser logged into the PeopleTools 8.52 (or later) system, select PeopleTools > Feeds > Feed Metadata Conversion.

2. Click the Search button next to the text field labeled Category to retrieve all imported feed categories. Alternatively, enter the initial letters of the feed category IDs for the feed categories that you want to retrieve in the Category text field and click Search.

   The Non-Metadata Feed Categories grid is loaded with imported feed categories.

3. Click in the Select column to select each of the feed categories that you want to convert.

4. Click the Convert button.

5. To verify that the selected feed categories were converted to the PeopleTools 8.52 format, go to the Define Feed Categories page (select PeopleTools, Feeds, Define Feed Categories).

6. Click the Search button to list all feed categories.

Converting Feed Metadata from the Current Release to PeopleTools 8.50 or 8.51

This section discusses how to:

- Convert from the current release to 8.51.

- Convert from the current release to 8.50.

Converting from the Current Release to 8.51

Converting feed definitions from the current release to PeopleTools 8.51 consists of two major steps:

- Creating a change package on the current release.
- Importing the change package into the PeopleTools 8.51 system using PeopleSoft Change Assistant.

To create a change package on the current release:

1. Sign into Application Designer on the current release in two-tier mode.
2. Create a new project.
3. Add the feed objects (feed definitions, feed data types, feed categories, or any combination of these objects) that you want to convert to the project.
4. From the menu, select File, Project Properties.
5. Select the Change Project option.
6. Go to the Update IDs tab.
7. Specify an update ID. If necessary, create a new update ID.
8. If there are prerequisite updates associated with this update, go to the Pre-Requisites tab and specify the update IDs there.
9. From the menu, select Tools, Create Change Package.
10. Specify the export directory—for example, C:\temp.
11. Select the option to generate a new template.
12. Click OK to begin the process of creating the change package.

    Several Application Engine processes will be launched in separate windows. A pop-up status message will indicate the success of the change package creation.

After the change package has been created, you can examine the data files, scripts, and miscellaneous files created in the update ID subdirectory of the export directory—C:\temp\upd*update_ID.*

To import the change package into PeopleTools 8.51 using PeopleSoft Change Assistant:

1. Run PeopleSoft Change Assistant on the PeopleTools 8.51 system.
2. Apply this update to the system.

   See "Applying Updates To A Target Environment" (PeopleTools 8.59: Change Assistant and Update Manager).

Converting from the Current Release to 8.50

Converting feed definitions from the current release to PeopleTools 8.50 consists of two major steps:

- Creating a change package on the current release.
- Importing the exported data into the PeopleTools 8.50 system using PeopleSoft Data Mover.

To create a change package on the current release:

1. Sign into Application Designer on the current release in two-tier mode.

2. Create a new project.
3. Add the feed objects (feed definitions, feed data types, or any combination of these objects) that you want to convert to the project.

   Note: Because feed categories were introduced in PeopleTools 8.50, feed category definitions cannot be converted to a PeopleTools 8.50 system. Do not include any feed category objects in your project.

4. From the menu, select File, Project Properties.
5. Select the Change Project option.
6. Go to the Update IDs tab.
7. Specify an update ID. If necessary, create a new update ID.
8. If there are prerequisite updates associated with this update, go to the Pre-Requisites tab and specify the update IDs there.
9. From the menu, select Tools, Create Change Package.
10. Specify the export directory—for example, C:\temp.
11. Select the option to generate a new template.
12. Click OK to begin the process of creating the change package.

    Several Application Engine processes will be launched in separate windows. A pop-up status message will indicate the success of the change package creation.

After the change package has been created, you can examine the data files, scripts, and miscellaneous files created in the update ID subdirectory of the export directory—for example, C:\temp\upd*update_ID.*

To import the exported data into PeopleTools 8.50 using PeopleSoft Data Mover:

1. Go to the batch scripts subdirectory of the export directory. For example, go to C:\temp \upd*update_ID*\upd*update_ID*_batch\scripts directory.
2. If you exported feed definitions, edit the upd*update_ID*_FeedDef_i.dms script.
3. Replace the `<%STAGINGDIRECTORY%>` placeholder with the path to the batch directory—for example, `C:\temp\upd`*`update_ID`*`\upd`*`update_ID`*`_batch`.
4. Replace the `IMPORT *;` command with the following commands:

   ```
   REPLACE_ALL PS_PTFP_FEED AS TEMP_PS_PTFP_FEED;


   INSERT INTO PS_PTFP_FEED
       SELECT
    T.PTFP_FEED_ID,
    T.DESCR,
    T.DESCR254,
    T.OBJECTOWNERID,
    T.PTFP_CATEGORY_ID,
    T.PTFP_DATATYPE_ID,
    T.PTFP_FEED_FORMAT,
    T.PTFP_CACHE_TYPE,
    T.PTFP_CACHE_TIME,
    T.PTFP_OPR_AUTH_TYPE,
   ```

```
 T.PTFP_OPR_ID,
 T.PTFP_OPR_PWD,
 T.PTFP_SEC_TYPE,
 T.IB_OPERATIONNAME,
 T.PORTAL_NAME,
 T.MSGNODENAME,
 T.CREATEDTTM,
 T.CREATEOPRID,
 T.LASTUPDDTTM,
 T.LASTUPDOPRID
  FROM TEMP_PS_PTFP_FEED T;

IMPORT PS_PTFP_ADMN_PREF;
IMPORT PS_PTFP_ATTRS;
IMPORT PS_PTFP_ATTRS_LANG;
IMPORT PS_PTFP_FEED_LANG;
IMPORT PS_PTFP_PARMS;
IMPORT PS_PTFP_PARMS_LANG;
IMPORT PS_PTFP_PUB_SITES;
IMPORT PS_PTFP_PVALS;
IMPORT PS_PTFP_PVALS_LANG;
IMPORT PS_PTFP_SECURITY;
IMPORT PS_PTFP_SETTINGS;


UPDATE PSSTATUS SET LASTREFRESHDTTM = %currentdatetimein;
```

5. Save your changes to the script.

6. Run PeopleSoft Data Mover on the PeopleTools 8.50 system.

7. Paste the script into the upper pane.

8. Run the script.

9. If you exported feed data types, edit the upd*update_ID*_FeedDat_i.dms script.

10. Replace the <%STAGINGDIRECTORY%> placeholder with the path to the batch directory — for example, C:\temp\upd*update_ID*\upd*update_ID*_batch.

11. Paste the script into the upper pane in PeopleSoft Data Mover.

12. Run the script.

Using SysAudit Information

SysAudit information is available for:

- Feed definition integrity.
- PS Query feed integrity.
- Worklist feed integrity.

Related Links

"Running SYSAUDIT" (PeopleTools 8.59: Data Management)

Archiving Feeds

This section provides an overview of archiving feed data and discusses how to archive scheduled feed data.

Understanding the Archiving of Feed Data

Scheduled feeds are published asynchronously and stored as messages in Integration Broker queues. The process of archiving feed data archives the messages stored in the Integration Broker queues used by scheduled feeds. You run a batch process to archive and delete the queue message data.

This section provides an overview of:

- Feed archiving parameters.
- Feed archiving options and logic.

Feed Archiving Parameters

The following parameters specified in the feed definition are used for archiving the Integration Broker queue messages:

- Integration Broker Service Operation name

  For generic scheduled feeds, the service operation name is the value of the IB_OPERATIONNAME data source setting.

  For other scheduled feeds, the system uses the service operations listed in the Feed Service Operations grid on the Define Data Types page.

- Integration Broker subqueue name, when applicable

  For generic scheduled feeds, no subqueue name exists.

  For other scheduled feeds, the Feed ID value is the sub queue name.

- PTFP_FEED:UTILITY:Utility DSPARAMETER_MAXROW

  For scheduled feeds, you must define this parameter in the data source parameter for the data type.

  This parameter indicates the number of feed items that will appear to the user.

  - If the parameter value is *PTFP_FEED:UTILITY:Utility SF_MAXROWOPTION_LATESTMSG*, then only the latest message in the Integration Broker queue appears to the user.
  - If the parameter value is *PTFP_FEED:UTILITY:Utility SF_MAXROWOPTION_ALLMSGS*, then all the messages in the Integration Broker queue appear to the user.
  - The parameter can be a number, for example, *4*, which indicates that only 4 items appear to the user.

- PTFP_FEED:UTILITY:Utility DSPARAMETER_SF_MAXMINUTES

  For scheduled feeds, you must define this parameter in the data source parameter for the data type.

This parameter tells the length of time for which the feed messages are valid in the Integration Broker queues.

- If the parameter value is *PTFP_FEED:UTILITY:Utility SF_MAXMINUTES_ALLMSGS*, then all the messages appear to the user.
- If the parameter value is a number, for example *100*, then all the feed messages existing in the Integration Broker queue for fewer than 100 minutes appear to the user.

Feed Archiving Options and Logic

This table describes the feeds archiving options and logic:

Option Value	***Description***	***Archiving Logic***
0	Archive all messages that are not within the specified date and time range.	When DSPARAMETER_SF _MAXMINUTES is not SF_ MAXMINUTES_ALLMSGS.
1	Archive all messages except latest 1 message per subqueue per language.	When DSPARAMETER_SF_ MAXMINUTES is SF_MAXMINUTES _ALLMSGS and DSPARAMETER_ MAXROW is SF_MAXROWOPTION_ LATESTMSG.
2	Archive all messages except latest *n* messages per subqueue per language.	When DSPARAMETER_SF_ MAXMINUTES is SF_MAXMINUTES _ALLMSGS and DSPARAMETER _MAXROW is some number. This number indicates the number of messages that should *not* be archived.
3	Archive nothing.	When DSPARAMETER_SF_ MAXMINUTES is SF_MAXMINUTES _ALLMSGS and DSPARAMETER_ MAXROW is SF_MAXROWOPTION_ ALLMSGS.

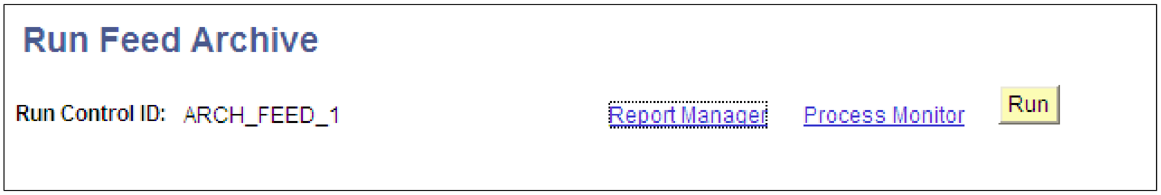
Note: The feed archiving options are implicitly arrived at based on the feed definition. These cannot be specified when the PTIBFEEDARCH process runs.

Archiving Scheduled Feed Data

Access the Run Feed Archive page (PeopleTools > Feeds > Archive Feed Data).

Image: Run Feed Archive page

This example illustrates the fields and controls on the Run Feed Archive page.



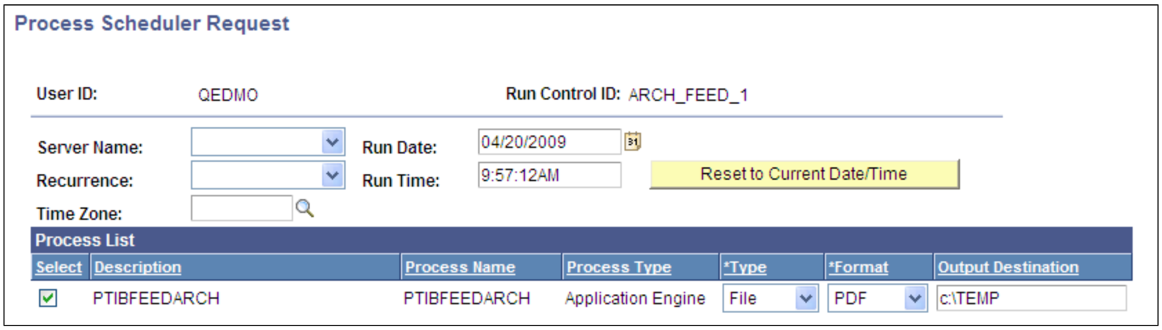
Run Feed Archive

Run Control ID: ARCH_FEED_1

Click the Run button to access the PTIBFEEDARCH process:

Image: Process Scheduler Request page showing the PTIBFEEDARCH process

This example illustrates the fields and controls on the Process Scheduler Request page showing the PTIBFEEDARCH process.



Process Scheduler Request

User ID: QEDMO

Run Control ID: ARCH_FEED_1

Server Name:

Run Date: 04/20/2009

Recurrence:

Run Time: 9:57:12AM

Time Zone:

Process List

Select	Description	Process Name	Process Type	*Type	*Format	Output Destination
☑	PTIBFEEDARCH	PTIBFEEDARCH	Application Engine	File	PDF	c:\TEMP

Use the Run Feed Archive page to archive Integration Broker messages that are used by scheduled feeds. Archiving takes no parameters from the user; parameters for archiving come from the feed definitions themselves.

All scheduled feeds are archived simultaneously; you cannot selectively archive feeds. The process archives and deletes the feed data based on the Archive flag in the Integration Broker queue used in the service operations that published messages to Integration Broker queues. If the Archive flag is enabled, then the messages are archived and then deleted; otherwise, the messages are just deleted.

You can view archived Integration Broker messages by using the Service Operations Monitor. Look for asynchronous services with the Archive flag enabled.

Note: A scheduled feed can be archived only if the service operations involved in the scheduled feeds are listed in the Feed Service Operations grid on the Define Feed Data Types page.

Note: Feeds archiving logic is comparable to Integration Broker archiving logic. For a feed message to be archived, you must select the Archive check box on the Queue Definition page for the service operation. If you do not select the Archive check box, then the feeds archiving process deletes the feed messages and the messages do not appear on the Service Operation Monitor - Asynchronous Services page when the Archive check box is selected.

See "Monitoring Asynchronous Service Operation Transactions" (PeopleTools 8.59: Integration Broker Service Operations Monitor).

Chapter 6

Creating and Using Integration Broker Generic Message Feeds

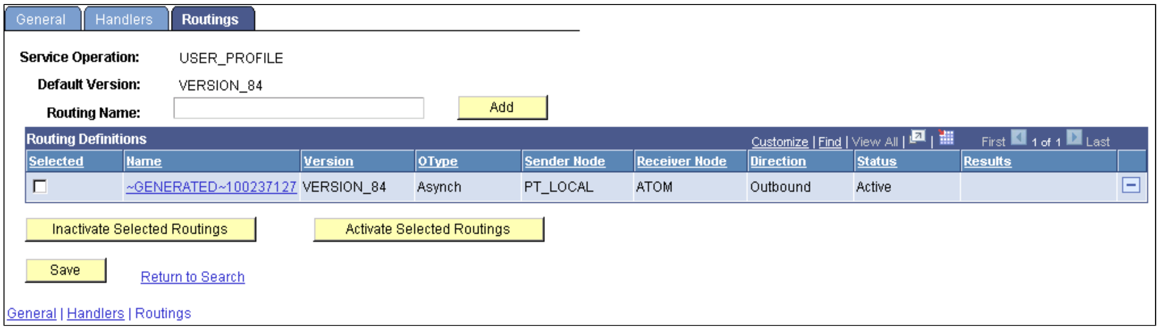
Understanding Integration Broker Generic Message Feeds

Integration Broker generic message feeds enable administrators to expose Integration Broker messages used in asynchronous, one-way service operations as feeds. These feeds are scheduled feeds. Unlike up-front feeds, for which the message contents in Integration Broker queues are feed messages, the messages published to Integration Broker queues in the case of generic feeds are either PeopleSoft rowset messages or non-rowset messages. These messages are called Integration Broker generic messages. Typically, these messages are generated based on actions performed by users or events occurring within the PeopleSoft system.

The Define Generic Feed page allows the feed administrator to select any existing asynchronous, one-way service operation in the system and expose it as a feed. When a feed is exposed for an asynchronous, one-way operation, the framework automatically creates a Local-to-Atom routing for the corresponding service operation.

Image: Example of service operation showing Local-to-Atom routing

This example illustrates the fields and controls on the Routings page, which shows the Local-to-Atom routing.


General
Handlers
Routings
Add
Routing Definitions
Customize | Find | View All |
First 1 of 1 Last
Inactivate Selected Routings
Activate Selected Routings
Save
Return to Search
General | Handlers | Routings

Service Operation: USER_PROFILE

Default Version: VERSION_84

Routing Name:

Selected	Name	Version	OType	Sender Node	Receiver Node	Direction	Status	Results
□	~GENERATED~100237127	VERSION_84	Asynch	PT_LOCAL	ATOM	Outbound	Active	

When a message is published for an asynchronous, one-way service operation, these messages will also be available in the feed, which is another form of application integration.

The default and recommended security option for generic Integration Broker message feeds is real-time security, although you can select other options as necessary. Real-time security allows only users who are in the permission list on the security page of the service operation to access that Integration Broker generic message as a feed. Users in the PeopleSoft Administrator and Portal Administrator roles also have access to Integration Broker generic message feeds.

Publishing Integration Broker Generic Message Feeds

This section provides an overview of the steps used to publish an Integration Broker generic message feed .

Understanding the Steps to Publish an Integration Broker Generic Message Feed

You can use either of these methods to publish an Integration Broker generic message as a feed:

- Using the Publish as Feed pages.

  When you use the Publish as Feed pages to publish an Integration Broker generic message as a feed, you will have access to the Advanced Options page, which facilitates setting advanced feed options for this type of feed. However, you will be unable to set any data source parameters as personalized — that is, as user specified or administrator specified.

- Using the Feed Publishing Wizard.

  When you use the Feed Publishing Wizard to publish an Integration Broker generic message as a feed, you will be able to set data source parameters as personalized — that is, as user specified or administrator specified. However, it is more difficult to set advanced feed options through the Feed Publishing Wizard because they are set by value, and not by selecting a native language option.

Because there advantages and limitations of both methods of publishing an Integration Broker generic message as a feed, you might want to publish the feed using one methodology, and then use the other methodology to edit the feed definition.

Publishing Using the Publish as Feed Pages

To publish an Integration Broker generic message as a feed using the Publish as Feed pages:

1. Select PeopleTools > Feeds > Define IB Generic Message Feed.
2. Select the service operation to publish as a feed.
3. On the Define IB Generic Message Feed page, click the Publish as Feed link.

   See Publishing an Integration Broker Generic Message as a Feed.
4. On the Publish Feed Definition page, define the feed properties and security.

   See Defining Feed Properties.
5. Click the Advanced Options link.
6. Define the advanced feed options.

   See Defining Advanced Options for Generic Message Feeds.
7. Click the OK button.
8. Click the Publish button.

Publishing Using the Feed Publishing Wizard

To publish an Integration Broker generic message as a feed using the Publish as Feed pages:

1. Select PeopleTools > Feeds > Feed Publishing Wizard.
2. Add a new value as the feed ID.
3. On the Specify Feed Properties page, define feed properties, such as the feed title, as well as additional feed properties.

   See Step 1: Specifying Feed Properties.
4. Click the Next button.
5. On the Select Data Source page, select *IB Generic Message Feeds* as the feed data type and then select the service operation as the data source.

   See Step 2: Selecting a Data Source.
6. Click the Next button.
7. On the Select Data Source Parameters page, set the usage type and default value for the data source parameters

   See Step 3: Specifying Data Source Parameters.

   If you set a parameter to be User Specified, click the Value link to specify the prompt values that are displayed to users when they personalize the feed; otherwise, continue with the next step.
8. Click the Next button.
9. On the Specify Publishing Options page, specify the publishing options for the feed, including feed security and sites.

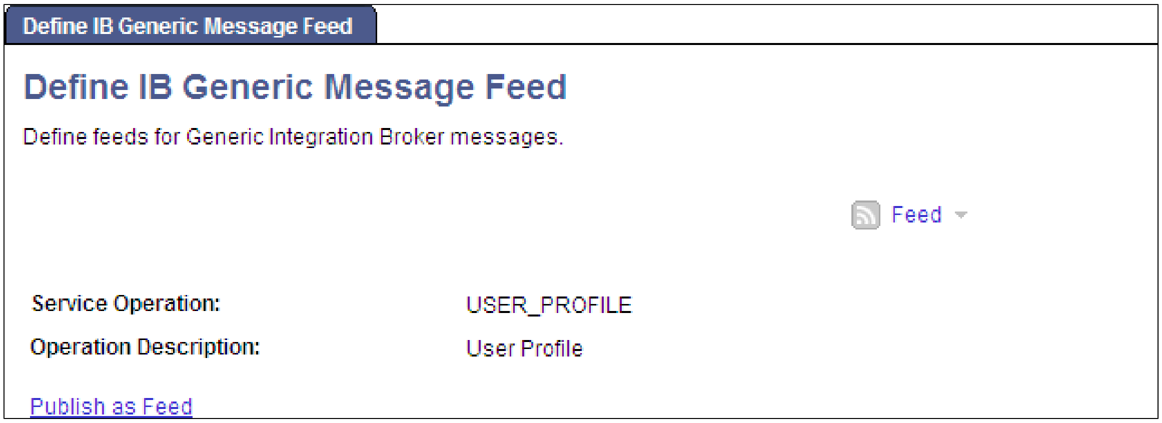
   See Step 4: Specifying Publishing Options.
10. Click the Save button to publish the feed.

Publishing an Integration Broker Generic Message as a Feed

Access the Define IB Generic Message Feed page (PeopleTools, Feeds, Define IB Generic Message Feed).

Image: Define IB Generic Message Feed page

This example illustrates the fields and controls on the Define IB Generic Message Feed page.



Define IB Generic Message Feed

Define feeds for Generic Integration Broker messages.

Service Operation: USER_PROFILE

Operation Description: User Profile

Use the Define IB Generic Message Feed page to publish an Integration Broker asynchronous, one-way service operation as a feed. Click the Publish as Feed link to begin creating the feed definition.

Create the feed definition on the Publish Feed Definition page.

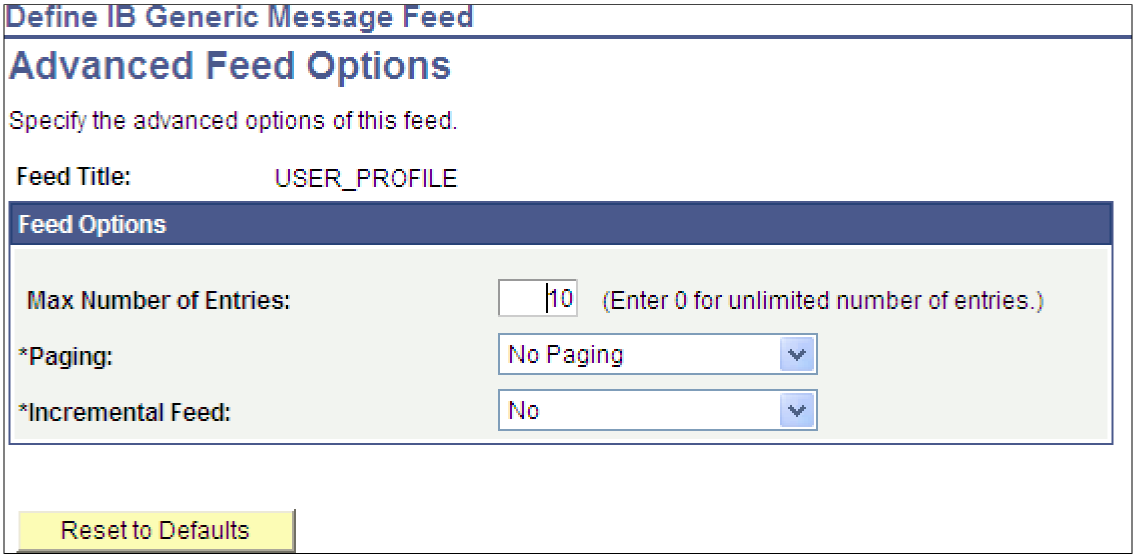
See Defining Feed Properties.

Defining Advanced Options for Generic Message Feeds

Access the Advanced Feed Options page (click the Advanced Options link on the Publish Feed Definition page).

Image: Define IB Generic Message Feed - Advanced Feed Options page

This example illustrates the fields and controls on the Define IB Generic Message Feed - Advanced Feed Options page. You can find definitions for the fields and controls later on this page.



Advanced Feed Options

Specify the advanced options of this feed.

Feed Title: USER_PROFILE

Feed Options

Max Number of Entries: 10 (Enter 0 for unlimited number of entries.)

*Paging: No Paging

*Incremental Feed: No

Use the Advanced Feed Options page to define advanced options for Integration Broker generic message feeds.

Max Number of Entries (maximum number of entries)

Enter the maximum number of entries that the feed should return to the user. Enter *0* for unlimited entries up to the maximum row limit specified on the Feed Options page, which is 300 by default.

Note: You configure the upper limit on the Feed Options page.

Paging

Select one of these options:

- *No Paging:* Indicates that the feed is not paged. This value is the default.
- *Segmented:* Indicates that the feed is paged.

  A *paged feed* is a feed that has been split into pages (also know as segments) to improve system performance in delivering the feed document and to improve performance for consuming a feed. A paged feed is presented with first, last, next, and previous links to provide access to additional pages in the feed document.

  Note: When *Segmented* is selected as the paging option, then the Max Number of Entries and Incremental Feed fields are disabled.

Incremental Feed

Select one of these options:

- *Yes:* Indicates that the feed is incremental.

  An *incremental feed* is a feed that has been published and updated with timestamps that allow the feed content to be delivered incrementally. An incremental feed allows the Feed Publishing Framework to deliver only the feed content that has changed since the feed was last requested by the user.
- *No:* Indicates that the feed is incremental. This value is the default.

Reset to Defaults

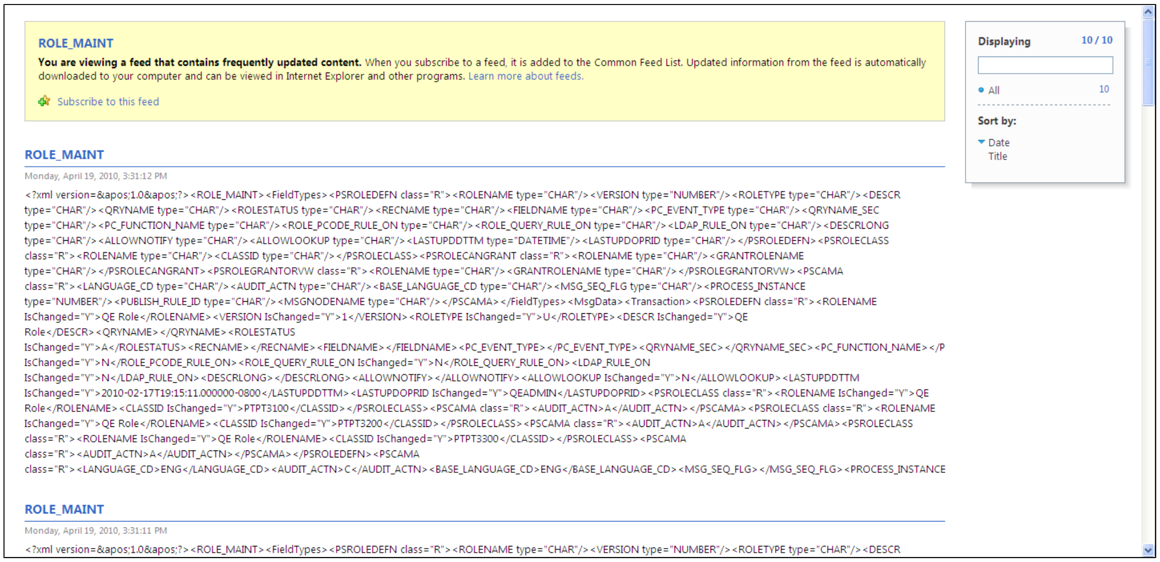
Click to reset the advanced options to their default values.

Using Integration Broker Generic Message Feeds

Integration Broker generic message feeds enable administrators to expose Integration Broker messages used in asynchronous, one-way service operations as feeds.

Image: Example generic message feed for the ROLE_MAINT service operation

This example illustrates a generic message feed for the ROLE_MAINT service operation.



You are viewing a feed that contains frequently updated content. When you subscribe to a feed, it is added to the Common Feed List. Updated information from the feed is automatically downloaded to your computer and can be viewed in Internet Explorer and other programs. Learn more about feeds.

ROLE_MAINT

Monday, April 19, 2010, 3:31:12 PM

```
<?xml version=&apos;1.0&apos;?><ROLE_MAINT><FieldTypes><PSROLEDEFN class="R"><ROLENAME type="CHAR"/><VERSION type="NUMBER"/><ROLETYPE type="CHAR"/><DESCR
type="CHAR"/><QRYNAME type="CHAR"/><ROLESTATUS type="CHAR"/><RECNAME type="CHAR"/><FIELDNAME type="CHAR"/><PC_EVENT_TYPE type="CHAR"/><QRYNAME_SEC
type="CHAR"/><PC_FUNCTION_NAME type="CHAR"/><ROLE_PCODE_RULE_ON type="CHAR"/><ROLE_QUERY_RULE_ON type="CHAR"/><LDAP_RULE_ON type="CHAR"/><DESCRLONG
type="CHAR"/><ALLOWNOTIFY type="CHAR"/><ALLOWLOOKUP type="CHAR"/><LASTUPDDTTM type="DATETIME"/><LASTUPDOPRID type="CHAR"/></PSROLEDEFN><PSROLECLASS
class="R"><ROLENAME type="CHAR"/><CLASSID type="CHAR"/></PSROLECLASS><PSROLECANGRANT class="R"><ROLENAME type="CHAR"/><GRANTROLENAME
type="CHAR"/></PSROLECANGRANT><PSROLEGRANTORVW class="R"><ROLENAME type="CHAR"/><GRANTROLENAME type="CHAR"/></PSROLEGRANTORVW><PSCAMA
class="R"><LANGUAGE_CD type="CHAR"/><AUDIT_ACTN type="CHAR"/><BASE_LANGUAGE_CD type="CHAR"/><MSG_SEQ_FLG type="CHAR"/><PROCESS_INSTANCE
type="NUMBER"/><PUBLISH_RULE_ID type="CHAR"/><MSGNODENAME type="CHAR"/></PSCAMA></FieldTypes><MsgData><Transaction><PSROLEDEFN class="R"><ROLENAME
IsChanged="Y">QE Role</ROLENAME><VERSION IsChanged="Y">1</VERSION><ROLETYPE IsChanged="Y">U</ROLETYPE><DESCR IsChanged="Y">QE
Role</DESCR><QRYNAME></QRYNAME><ROLESTATUS
IsChanged="Y">A</ROLESTATUS><RECNAME></RECNAME><FIELDNAME></FIELDNAME><PC_EVENT_TYPE></PC_EVENT_TYPE><QRYNAME_SEC></QRYNAME_SEC><PC_FUNCTION_NAME></P
IsChanged="Y">N</ROLE_PCODE_RULE_ON><ROLE_QUERY_RULE_ON IsChanged="Y">N</ROLE_QUERY_RULE_ON><LDAP_RULE_ON
IsChanged="Y">N</LDAP_RULE_ON><DESCRLONG></DESCRLONG><ALLOWNOTIFY></ALLOWNOTIFY><ALLOWLOOKUP IsChanged="Y">N</ALLOWLOOKUP><LASTUPDDTTM
IsChanged="Y">2010-02-17T19:15:11.000000-0800</LASTUPDDTTM><LASTUPDOPRID IsChanged="Y">QEADMIN</LASTUPDOPRID><PSROLECLASS class="R"><ROLENAME IsChanged="Y">QE
Role</ROLENAME><CLASSID IsChanged="Y">PTPT3100</CLASSID></PSROLECLASS><PSCAMA class="R"><AUDIT_ACTN>A</AUDIT_ACTN></PSCAMA><PSROLECLASS class="R"><ROLENAME
IsChanged="Y">QE Role</ROLENAME><CLASSID IsChanged="Y">PTPT3200</CLASSID></PSROLECLASS><PSCAMA class="R"><AUDIT_ACTN>A</AUDIT_ACTN></PSCAMA><PSROLECLASS
class="R"><ROLENAME IsChanged="Y">QE Role</ROLENAME><CLASSID IsChanged="Y">PTPT3300</CLASSID></PSROLECLASS><PSCAMA
class="R"><AUDIT_ACTN>A</AUDIT_ACTN></PSCAMA></PSROLEDEFN><PSCAMA
class="R"><LANGUAGE_CD>ENG</LANGUAGE_CD><AUDIT_ACTN>C</AUDIT_ACTN><BASE_LANGUAGE_CD>ENG</BASE_LANGUAGE_CD><MSG_SEQ_FLG></MSG_SEQ_FLG><PROCESS_INSTANCE
```

ROLE_MAINT

Monday, April 19, 2010, 3:31:11 PM

```
<?xml version=&apos;1.0&apos;?><ROLE_MAINT><FieldTypes><PSROLEDEFN class="R"><ROLENAME type="CHAR"/><VERSION type="NUMBER"/><ROLETYPE type="CHAR"/><DESCR
```

Chapter 7

Creating and Using Query Feeds

Understanding Query Feeds

This section discusses query feed security.

Query Feed Security

The Feed Publishing Framework has two levels of security: feed security and data security. Feed security controls who can see the feed when searching for feeds. Data security controls whether the underlying data can be seen in the feed entries when a user requests the feed document.

Query feed security, which feed administrators specify on the advanced options page, determines feed security, that is, who can see the feed in search results.

Publishing Query Feeds

This section provides an overview of the steps used to publish a query feed and discusses how to publish a query as a feed.

Understanding the Steps to Publish a Query Feed

You can use either of these methods to publish a query as a feed:

- Using the Publish as Feed pages.

  When you use the Publish as Feed pages to publish a query as a feed, you will have access to the Advanced Options page, which facilitates setting advanced feed options for this type of feed. However, you will be unable to personalize any data source parameters:

  - You must specify all query prompts as fixed values. For all users, each time the feed is requested, the exact same query is run.
  - You cannot use user-specific or administrator configurable values.

- Using the Feed Publishing Wizard.

  When you use the Feed Publishing Wizard to publish a query as a feed, you will be able to personalize data source parameters — that is, as user-specified or administrator-specified:

  - You can specify the usage type for each query prompt as: user-specified, administrator-specified, fixed, or system variable.

- For administrator-specified parameters, you specify a default value. For user-specified parameters, you also specify a default value and can specify a list of valid values.

Because there advantages and limitations of both methods of publishing a query as a feed, you might want to publish the feed using one methodology, and then use the other methodology to edit the feed definition.

Publishing Using the Publish as Feed Pages

To publish a query as a feed using the Publish as Feed pages:

Note: Any user with access to Query Manager can publish query feeds.

1. Select Reporting Tools > Query > Query Manager.
2. Select the query that you want to publish as a feed and click the Edit link.
3. Click the Publish as Feed link.

   Note: If a feed has already been published for this query, then the link appears as Manage Feeds rather than Publish as Feed.

   See Publishing a Query as a Feed.
4. On the Publish Feed Definition page, define the feed properties.

   See Defining Feed Properties.
5. Click the Advanced Options link.
6. Enter the query parameters, define advanced options such as query feed security, and map feed entry elements for the query feed.

   Enter the prompt values that will be used to run the query before you publish the feed.

   Important! If you do not enter necessary prompt values, then the published feed will contain no entries.

   See Defining Advanced Options for Query Feeds.
7. Click the OK button.
8. Click the Publish button.

Publishing Using the Feed Publishing Wizard

To publish a query as a feed using the Publish as Feed pages:

1. Select PeopleTools > Feeds > Feed Publishing Wizard.
2. Add a new value as the feed ID.
3. On the Specify Feed Properties page, define feed properties, such as the feed title, as well as additional feed properties.

See Step 1: Specifying Feed Properties.

4. Click the Next button.
5. On the Select Data Source page, select *PS Query Feeds* as the feed data type and then select the query as the data source.

   See Step 2: Selecting a Data Source.
6. Enter the query parameters, define advanced options such as query feed security, and map feed entry elements for the query feed.

   See Defining Advanced Options for Query Feeds.
7. Click the Next button.
8. On the Select Data Source Parameters page, set the usage type and default value for the data source parameters

   See Step 3: Specifying Data Source Parameters.

   If you set a parameter to be user specified, click the Value link to specify the prompt values that are displayed to users when they personalize the feed; otherwise, continue with the next step.
9. Click the Next button.
10. On the Specify Publishing Options page, specify the publishing options for the feed.

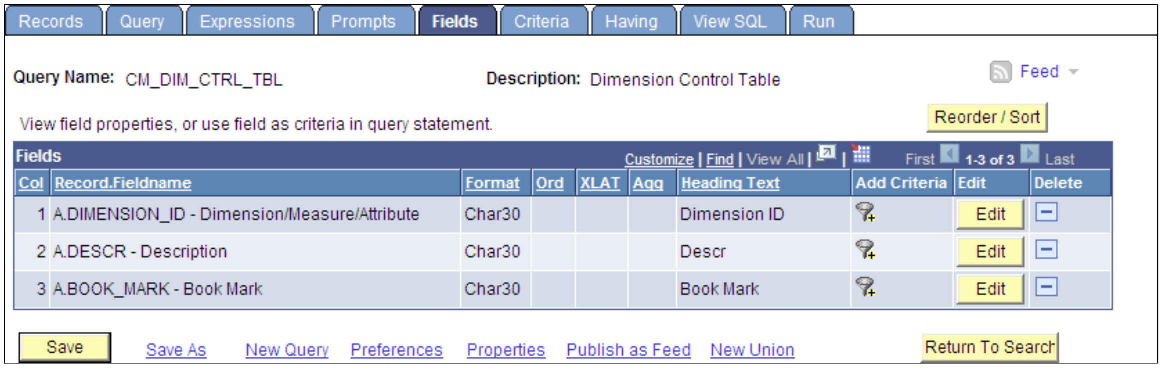
    See Step 4: Specifying Publishing Options.
11. Click the Save button to publish the feed.

Publishing a Query as a Feed

Access the Fields page (Reporting Tools, Query, Query Manager).

Image: Fields page

This example illustrates the fields and controls on the Fields page. You can find definitions for the fields and controls later on this page.



Query Name: CM_DIM_CTRL_TBL

Description: Dimension Control Table

View field properties, or use field as criteria in query statement.

Fields

Col	Record.Fieldname	Format	Ord	XLAT	Agg	Heading Text	Add Criteria	Edit	Delete
1	A.DIMENSION_ID - Dimension/Measure/Attribute	Char30				Dimension ID		Edit	⊟
2	A.DESCR - Description	Char30				Descr		Edit	⊟
3	A.BOOK_MARK - Book Mark	Char30				Book Mark		Edit	⊟

Use the Fields page to publish a query as a feed. Click the Publish as Feed link to begin creating the feed definition.

Note: If a feed has already been published for this query, then the link appears as Manage Feeds rather than Publish as Feed.

Create the feed definition on the Publish Feed Definition page.

See Defining Feed Properties.

Defining Advanced Options for Query Feeds

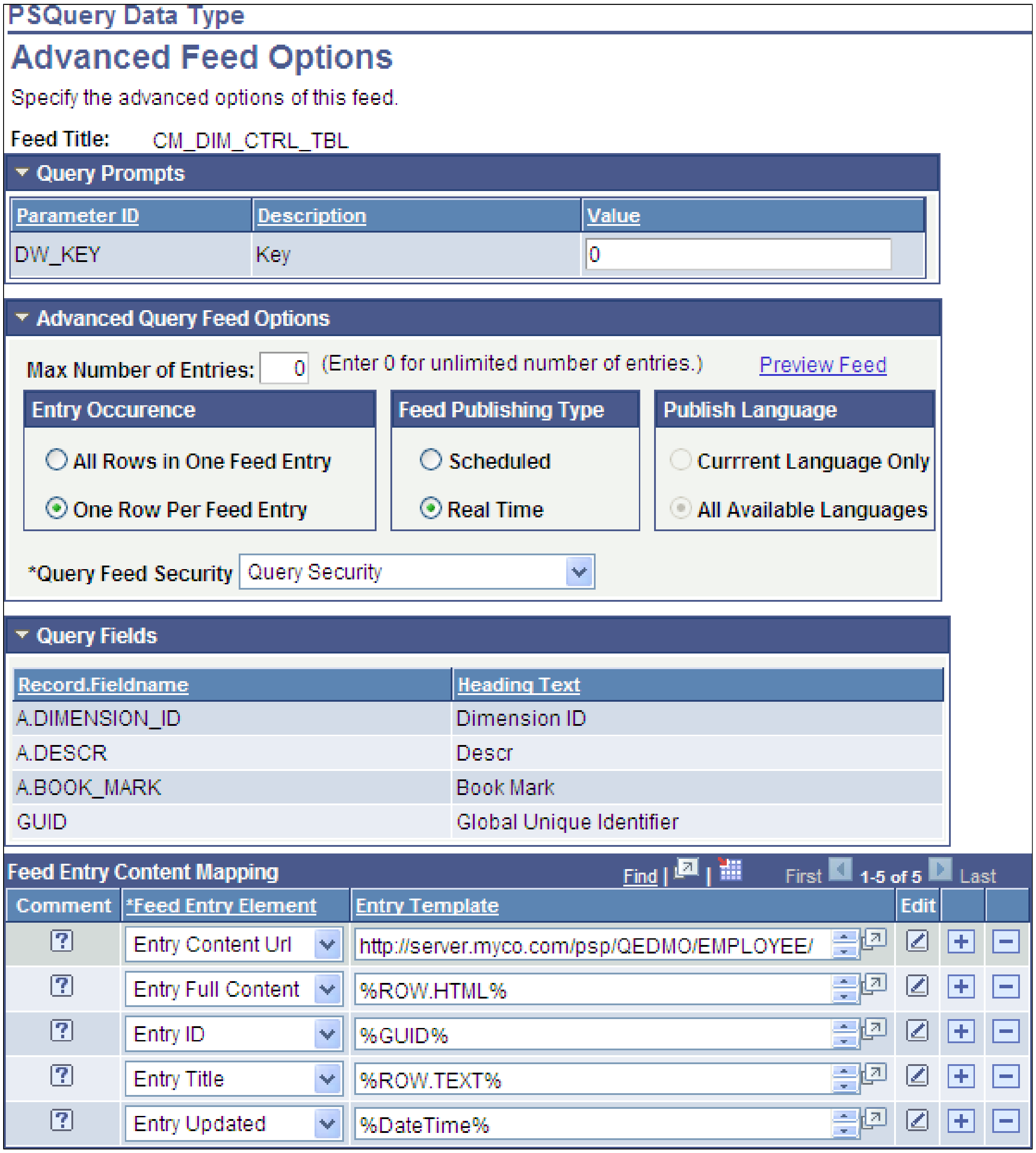
This section discusses how to define advanced options for query feeds.

Entering Advanced Feed Options for Query Feeds

Access the Advanced Feed Options page (click the Advanced Options link on the Publish Feed Definition page).

Image: PSQuery Data Type - Advanced Feed Options page

This example illustrates the fields and controls on the PSQuery Data Type - Advanced Feed Options page. You can find definitions for the fields and controls later on this page.



PSQuery Data Type

Advanced Feed Options

Specify the advanced options of this feed.

Feed Title: CM_DIM_CTRL_TBL

Query Prompts

Parameter ID	Description	Value
DW_KEY	Key	0

Advanced Query Feed Options

Max Number of Entries: 0 (Enter 0 for unlimited number of entries.) Preview Feed

Entry Occurence	Feed Publishing Type	Publish Language
○ All Rows in One Feed Entry	○ Scheduled	○ Currrent Language Only
● One Row Per Feed Entry	● Real Time	● All Available Languages

*Query Feed Security: Query Security

Query Fields

Record.Fieldname	Heading Text
A.DIMENSION_ID	Dimension ID
A.DESCR	Descr
A.BOOK_MARK	Book Mark
GUID	Global Unique Identifier

Feed Entry Content Mapping   Find | First 1-5 of 5 Last

Comment	*Feed Entry Element	Entry Template	Edit
?	Entry Content Url	http://server.myco.com/psp/QEDMO/EMPLOYEE/	
?	Entry Full Content	%ROW.HTML%	
?	Entry ID	%GUID%	
?	Entry Title	%ROW.TEXT%	
?	Entry Updated	%DateTime%	

Query Prompts

Any fields that are query prompts appear in this area. Enter the prompt values that will be used to run the query before you publish the feed.

Important! If you do not enter necessary prompt values, then the published feed contains no entries.

Advanced Query Feed Options

Max Number of Entries

Enter the maximum number of entries that the feed should return to the user. Enter *0* for unlimited entries up to the maximum row limit specified on the Feed Options page, which is 300 by default.

Note: You can configure the upper limit on the Feed Options page.

Preview Feed

Click to save and temporarily publish and view the feed.

Entry Occurrence

Select from these options:

All Rows in One Feed Entry: Select this option to have the feed reader display one entry for the entire query.

One Row Per Feed Entry: Select this option to have the feed reader display one entry for each row returned by the query.

Feed Publishing Type

Select from these options:

Scheduled: Select this option to use Query Scheduler to schedule the query to run once and be stored in the Integration Broker queues. This option provides superior performance, but it should be used when the data changes infrequently. If you select this option, you may select whether to publish the feed in the language of the current user or all available languages.

See "Scheduling Queries" (PeopleTools 8.59: Query).

Real Time: Select this option to run the query in real-time each time a user requests that this query feed be updated in the feed reader. If you select this option, you can only publish in the language of the user configuring the feed.

Publish Language

Active only when Feed Publishing Type is set to *Scheduled*, this field determines the languages in which the system publishes the query feed. Select from these options:

Current Language Only: Select to publish the query feed in the language of the content owner.

All Available Languages: Select to publish the query feed in all translated languages in your database.

Query Feed Security

Select from these options:

Public Feed: Select to make the query feed available to all users. This option is available with all Feed Publishing types.

Note: The user context for processing the query is the default user on the ANONYMOUS node.

Query Security: Select to use Query security as the basis for the feed-level security. This option is available only with the *Real Time* Feed Publishing type.

When you define this type of query feed security, only the users who have rights to modify the query in Query Manager can find and view the query feed.

DistributionList/QuerySecurity: Select to use the Query distribution list in Report Manager as the basis for feed-level security. This option is available only with the *Scheduled* Feed Publishing type.

When you define this type of query feed security, only the users that are on the distribution list when the query feed is scheduled to run from the Schedule Query Request dialog box can find and view the feed.

See Defining Feed Properties.

Query Fields

This grid displays the fields that are available to the query as defined in the query definition in Query Manager. The grid is display-only.

Feed Entry Content Mapping

The fields in the Feed Entry Content Mapping grid on the Advanced Feed Options page enable you to create a template for the query data that is used to generate each feed entry. You select feed entry elements and either use the given template or build one of your own using the Feed Element Mapping Builder. This information, when published, determines how each entry appears to the user.

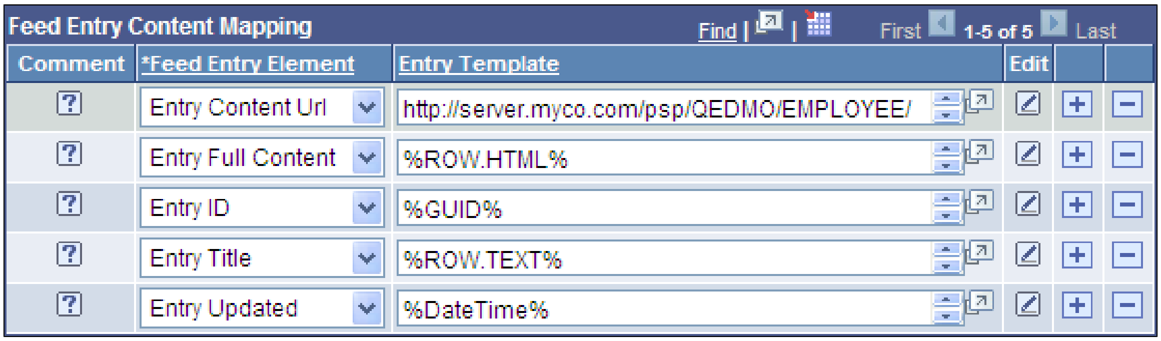
See Mapping Feed Entry Elements to Feed Entry Templates.

Mapping Feed Entry Elements to Feed Entry Templates

The Feed Entry Content Mapping grid appears at the bottom of the PSQuery Data Type - Advanced Feed Options page.

Image: Feed Entry Content Mapping grid

This example illustrates the fields and controls on the Feed Entry Content Mapping grid. You can find definitions for the fields and controls later on this page.



Feed Entry Content Mapping — Find | First 1-5 of 5 Last

Comment	*Feed Entry Element	Entry Template	Edit		
?	Entry Content Url	http://server.myco.com/psp/QEDMO/EMPLOYEE/		+	−
?	Entry Full Content	%ROW.HTML%		+	−
?	Entry ID	%GUID%		+	−
?	Entry Title	%ROW.TEXT%		+	−
?	Entry Updated	%DateTime%		+	−

You can use one of three methods to edit feed entry templates for feed entries:

- Directly in the Entry Template field in the Feed Entry Content Mapping grid.
- In a modal window when you click the Display Entry Element in Modal Window button.
- In Mapping Builder when you click the Edit button.

Important! Unless you are aware of the proper encodings to use, you should use the Mapping Builder to edit or create feed entry templates.

See Using Mapping Builder to Edit Feed Entry Templates.

The Feed Entry Content Mapping grid has these fields:

? Comment

Click to display more information about this feed entry element in a modal window.

Feed Entry Element

Each feed entry element is a property of an entry that can be interpreted and displayed by a feed reader.

Note: Not all feed readers display all elements. Consult the documentation for your feed reader.

Entry Template

A template is a string that encloses bind variables inside `%` symbols. When the feed is published, the string as XML passes to the feed reader, which displays the content from the PSQuery data source according to the template instructions.

Feed entry templates support these bind variable types:

- *System variable*, for example, `%Copyright%`.
- *Template variable*, for example, `%ROW.HTML%`.
- *Query columns*, for example, `%ALIASNAME.QUERYCOLUMNNAME%`.
- *Message Catalog text*, for example, `%MSGT:10.10%`.

- *Message Catalog explanation*, for example, `%MSGE:10.10%`.

Display Entry Element in Modal Window

- On the grid itself, click to display the entire grid in a separate modal window.
- To the right of an feed entry template, click to display and edit the text of the feed entry template in a separate modal window.

Edit

Click to access the Feed Element Mapping Builder page to use the Mapping Builder to modify the feed entry template.

See Using Mapping Builder to Edit Feed Entry Templates.

This table describes the values that you can select for the Feed Entry Element field:

Feed Entry Element Field	***Description***	***Action of Feed Reader***	***Feed Entry Template Values***	***Runtime Processing***
Entry Title	This element is a brief, single-line explanation of the feed entry. Only one entry title can exist per entry.	The reader displays the title as indicated by the template.	Any template containing any bind variable. Can have HTML tags but must be only one line.	The system escapes the HTML as well as XML characters.
Entry Content URL	This element is the URL to the PeopleSoft application page or other content. Only one entry content URL can exist per entry.	The reader displays an active link to enable the user to navigate to this URL.	Can map this to a field containing a URL, a computed field, or a static URL which has query columns as parameters	The system escapes no characters. Make sure that the URL is valid and confirm that you can use it as an attribute.
Entry Categories	This element defines the category of an entry. Multiple entries can belong to the same category, and one entry can belong to multiple categories.	The reader organizes and filters entries based on values.	Any template containing any bind variable, but user should consider short values for ease of organization.	The system escapes the HTML as well as XML characters.
Entry ID	This element is a permanent, universal identifier. Only one ID can exist per entry.	The reader detects changes to an existing entry and presents modifications of the existing entry instead of creating a new entry, while ignoring unchanged entries.	Any template containing any bind variable.	The system escapes the HTML as well as XML characters.

Feed Entry Element Field	***Description***	***Action of Feed Reader***	***Feed Entry Template Values***	***Runtime Processing***
Entry Updated	This element is the date and time that the entry was most recently modified. Only one update date can exist per entry.	The reader uses this entry to determine, in conjunction with the ID, whether to update the data in the entry.	Static value in PeopleTools date and time format. Dynamic value present in any date time query column. System variable `%DateTime`.	Values that are PeopleTools date time formatted will be converted to atom specific date time format.
Entry Published	This element is the date and time when the entry was originally created. Only one publish date can exist per entry.	The reader uses this entry to determine the maximum age of the data.	Static value in PeopleTools date and time format. Dynamic value present in any date time query column. System variable `%DateTime`.	Values that are PeopleTools date time formatted will be converted to atom specific date time format.
Entry Author	This element pertains to information about the creator, owner, and author of this entry. Only one author can exist per entry.	The reader organizes and filters entries based on the value of this field. The reader can use this field to send email to the author.	Static values in this format: `Name[EmailAddre⇒ ss]` Author name and email ID can be mapped to query columns or system variables.	The system extracts the email address extracts from [] and the name from the value before the brackets. **Note:** The brackets may be left empty, but they must be present for the system to correctly process this entry. The system escapes the HTML as well as XML characters.
Entry Contributors	This element pertains to information about various parties involved in this entry. Multiple contributors can exist per entry.	The reader organizes and filters entries based on the value of this field. The reader can use this field to send email to the author.	Static values in this format: `Name[EmailAddre⇒ ss]` Contributor name and email ID can be mapped to query columns or system variables.	The system extracts the email address extracts from [] and the name from the value before the brackets. **Note:** The brackets may be left empty, but they must be present for the system to correctly process this entry. The system escapes the HTML as well as XML characters.

Feed Entry Element Field	***Description***	***Action of Feed Reader***	***Feed Entry Template Values***	***Runtime Processing***
Entry Description	This element is a brief description about the entry, which can be more detailed information than a title but less information compared to full content.	The reader presents more information than the title but less than the full content, while providing a link to the full content.	Any template containing any bind variable. HTML tags.	The system escapes the HTML as well as XML characters.
Entry Enclosures	This element contains other content types accessible using a URL, for example images and mp3 files. Multiple enclosures signifying multiple attachments can exist per entry.	The reader can automatically download content and display it in place or enable the viewer to access it by using external tools based on the content type.	URL to the resource, its content, and its length in bytes. Use this format: `""href="URL" type="content/type" length="byte-size""`. The system extracts the values inside the quotes. You can leave empty quotes in case the value needs to be ignored. You can map the href, type, and length values that are within the quotes to any bind variable, but the template output should be a valid URL/Content-type/length.	The system escapes the HTML as well as XML characters.
Entry Full Content	This element is the body of the entry. Only one full content can exist per entry.	The reader can display the content of the entry by default or upon user request.	Any template containing any bind variable. HTML tags.	The system escapes the HTML as well as XML characters.
Expires	This element is the date time when this entry is invalid and should be considered outdated. Only one expiration date time can exist per entry.	The reader determines when to discard this entry.	Static value in PeopleTools date and time format. Dynamic value present in any date time query column.	Values that are PeopleTools date time formatted will be converted to atom specific date time format. The system escapes the HTML as well as XML characters.
Max Age	This element is the length of time in milliseconds until the entry will be valid after the Published or Updated date and time. Only one maximum age can exist per entry.	The reader determines when to discard this entry.	Static number or mapped to a query column, which results in number.	The system escapes the HTML as well as XML characters.

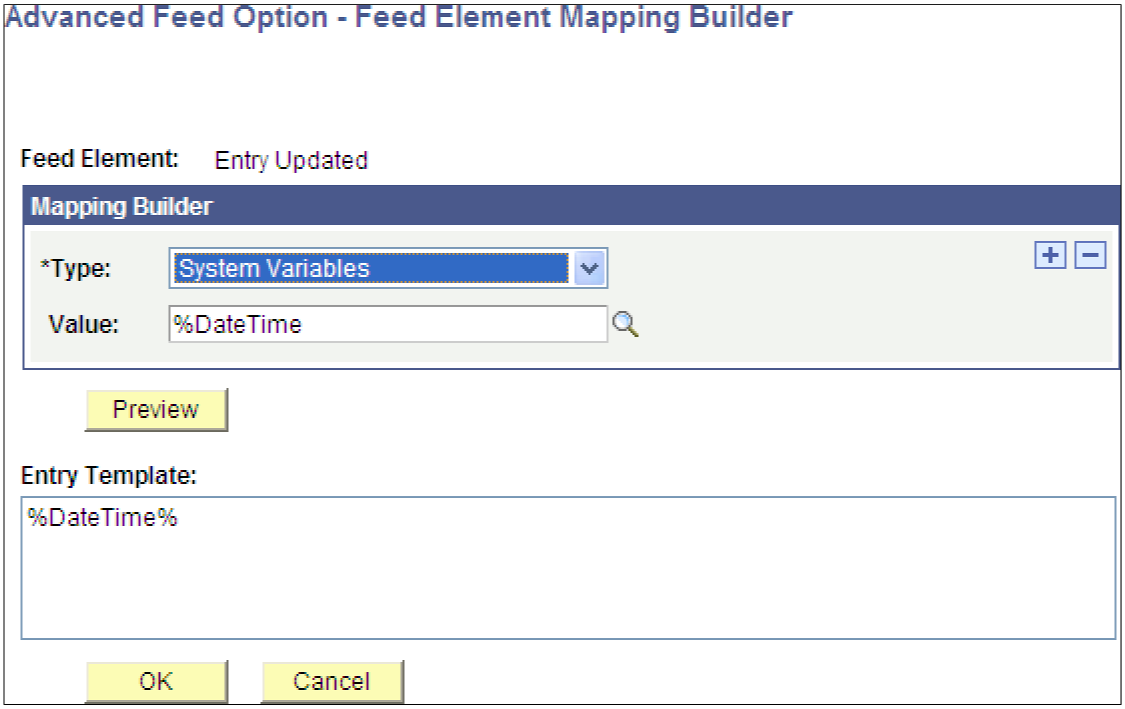
Feed Entry Element Field	***Description***	***Action of Feed Reader***	***Feed Entry Template Values***	***Runtime Processing***
Entry Copyright	This element contains the copyright information. Only one copyright can exist per entry.	The reader displays the copyright information upon request.	Any template containing any bind variable. The `%Copyright%` system variable can be used to map to the system-level copyright information.	The system escapes the HTML as well as XML characters.

Using Mapping Builder to Edit Feed Entry Templates

Access the Feed Element Mapping Builder page (click the Edit button next to a feed entry template in the Feed Entry Content Mapping grid).

Image: Feed Element Mapping Builder page

This example illustrates the fields and controls on the Advanced Feed Option — Feed Element Mapping Builder page.



Advanced Feed Option - Feed Element Mapping Builder

Feed Element: Entry Updated

Mapping Builder

*Type: System Variables

Value: %DateTime

Entry Template:

%DateTime%

You edit feed entry templates using Mapping Builder, which enables you to create custom output for feed entries using your own feed entry templates. You do this by assembling various combinations of static text and variable elements, or by writing directly in the rich text editor.

Note: Currently, sharing templates among feed administrators is not supported. However, you can use the Download icon to export the grid values so that they can be used with other query feeds.

Assembling Feed Entry Template Elements

You can edit these feed entry elements by assembling combinations of element types:

- Entry Author
- Entry Categories
- Entry Content URL
- Entry Contributors
- Entry Copyright
- Entry Enclosures
- Entry ID
- Entry Published
- Entry Updated
- Expires
- Max Age

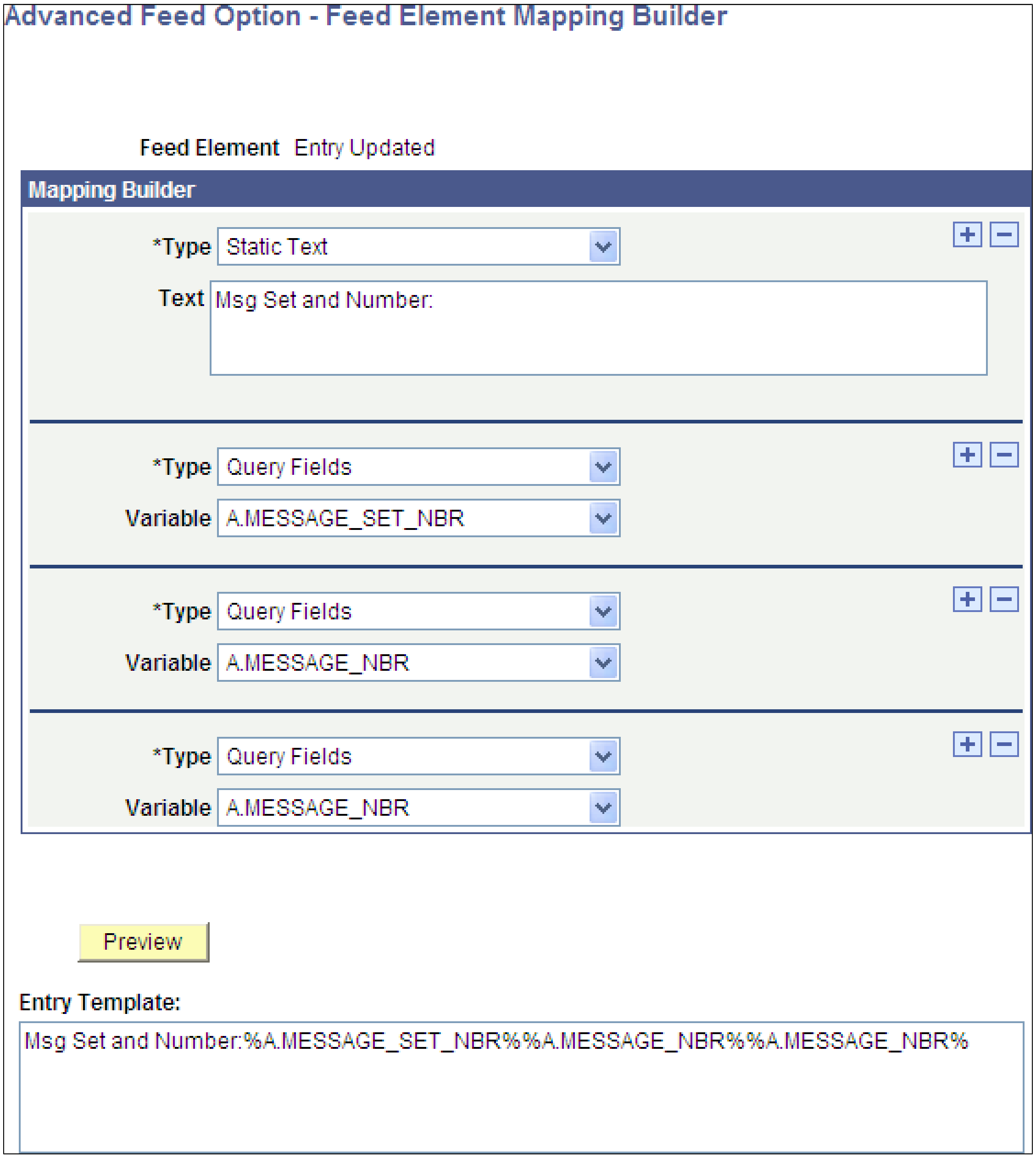
To assemble template elements into a feed entry template:

1. On the Advanced Options page, click the Edit button associated with the feed entry template you want to edit.
2. In the Mapping Builder grid, select the Type of element to add to the feed entry template.
3. Click the + button to add another element.
4. Continue adding rows and constructing the feed entry template.
5. Click the Preview button.
6. Click the OK button.

This example shows an assembled feed entry template:

Image: Example of the Feed Element Mapping Builder page showing an assembled feed entry template

This example illustrates the fields and controls on the Advanced Feed Option — Feed Element Mapping Builder page showing an assembled feed entry template. You can find definitions for the fields and controls later on this page.



Advanced Feed Option - Feed Element Mapping Builder

Feed Element Entry Updated

Mapping Builder

*Type Static Text

Text Msg Set and Number:

*Type Query Fields

Variable A.MESSAGE_SET_NBR

*Type Query Fields

Variable A.MESSAGE_NBR

*Type Query Fields

Variable A.MESSAGE_NBR

Entry Template:

Msg Set and Number:%A.MESSAGE_SET_NBR%%A.MESSAGE_NBR%%A.MESSAGE_NBR%

The Mapping Builder fields are described as follows.

Type

Select from these Mapping Builder types to assemble the feed entry template:

Message Catalog Explanation

Message Catalog Text

Query Fields

Static Text

System Variables

Templates

Message

This field is visible when the Type field value is *Message Catalog Explanation* and *Message Catalog Text*.

Enter the message number to appear in the entry.

Note: Use the Message Catalog rather than static text to generate a template that is language independent.

Msg Set (message set)

Visible when the Type field value is *Message Catalog Explanation* and *Message Catalog Text*.

Enter the message set number to appear in the entry.

Note: Use the Message Catalog rather than static text to generate a template that is language independent.

Text

This field is visible when the Type field value is *Static Text*.

Enter the text that you want to appear in the entry.

Note: To have a percent sign appear in the text of the template, it needs to escaped using `%%`. For example, if the template reads `%%`, then at runtime the builder replaces it with `%`.

Value

This field is visible when the Type field value is *System Variables*.

Select from any of the available system variables.

See "System Variables Reference" (PeopleTools 8.59: PeopleCode Language Reference).

Variable

This field is visible when the Type field value is *Templates* or *Query Fields*.

When the Type field value is *Templates,* select from these Template options:

- *Row in HTML Format*
- *Row in TEXT Format*

When the Type field value is *Query Fields,* select from any field comprising the query definition.

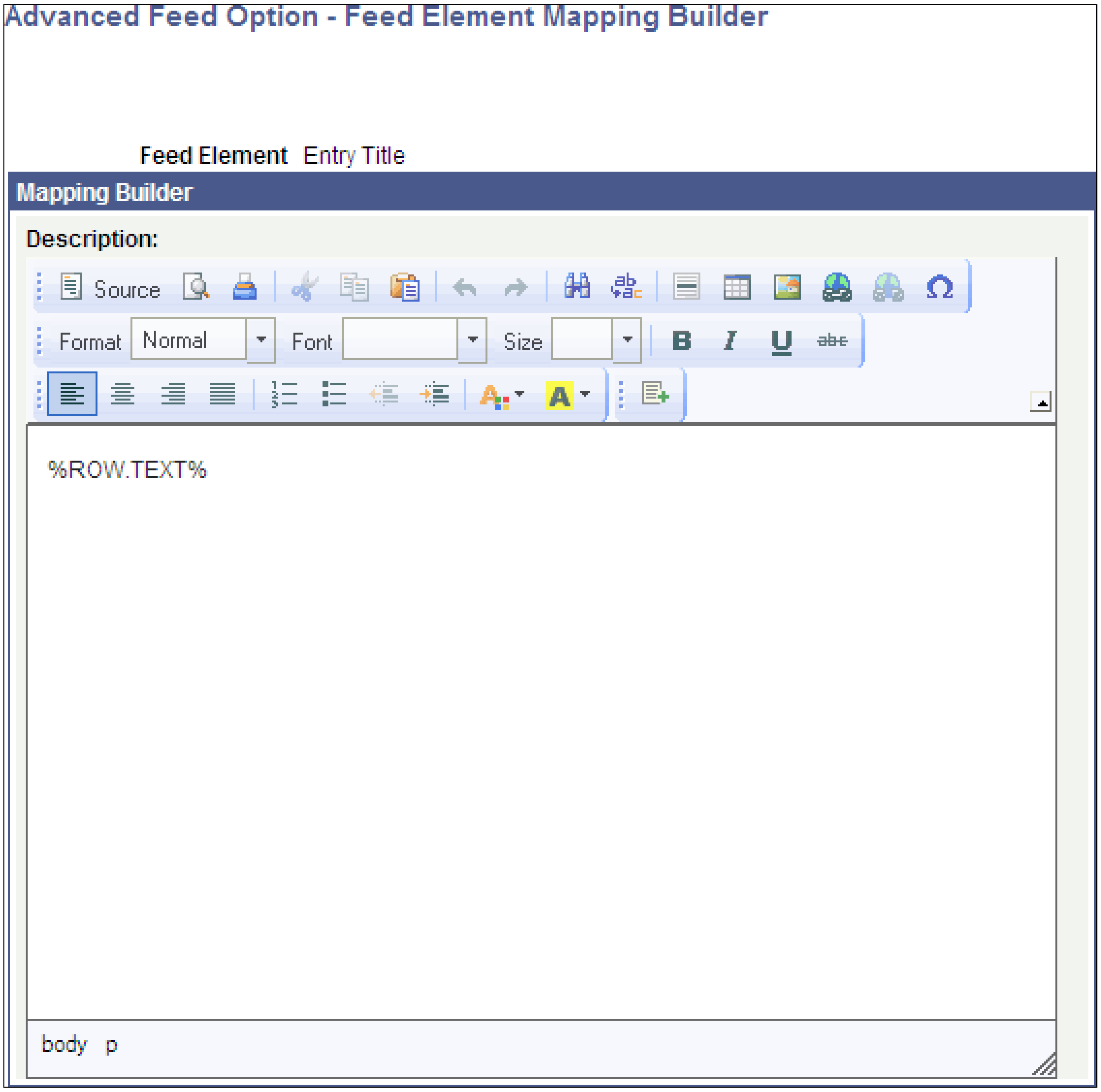
Creating Feed Entry Templates in the Rich Text Editor

These feed entry elements present the rich text editor, where you can enter HTML, rich text, and bind variables to create a variety of output types to the feed reader:

- Entry Description
- Entry Full Content
- Entry Title

Image: Example of Feed Element Mapping Builder page showing a feed entry template built by using the rich text editor

This example illustrates the fields and controls on the Advanced Feed Option — Feed Element Mapping Builder page showing a feed entry template built by using the rich text editor.



Advanced Feed Option - Feed Element Mapping Builder

Feed Element Entry Title

Mapping Builder

Description:

%ROW.TEXT%

Accessing Bind Variables in the Rich Text Editor

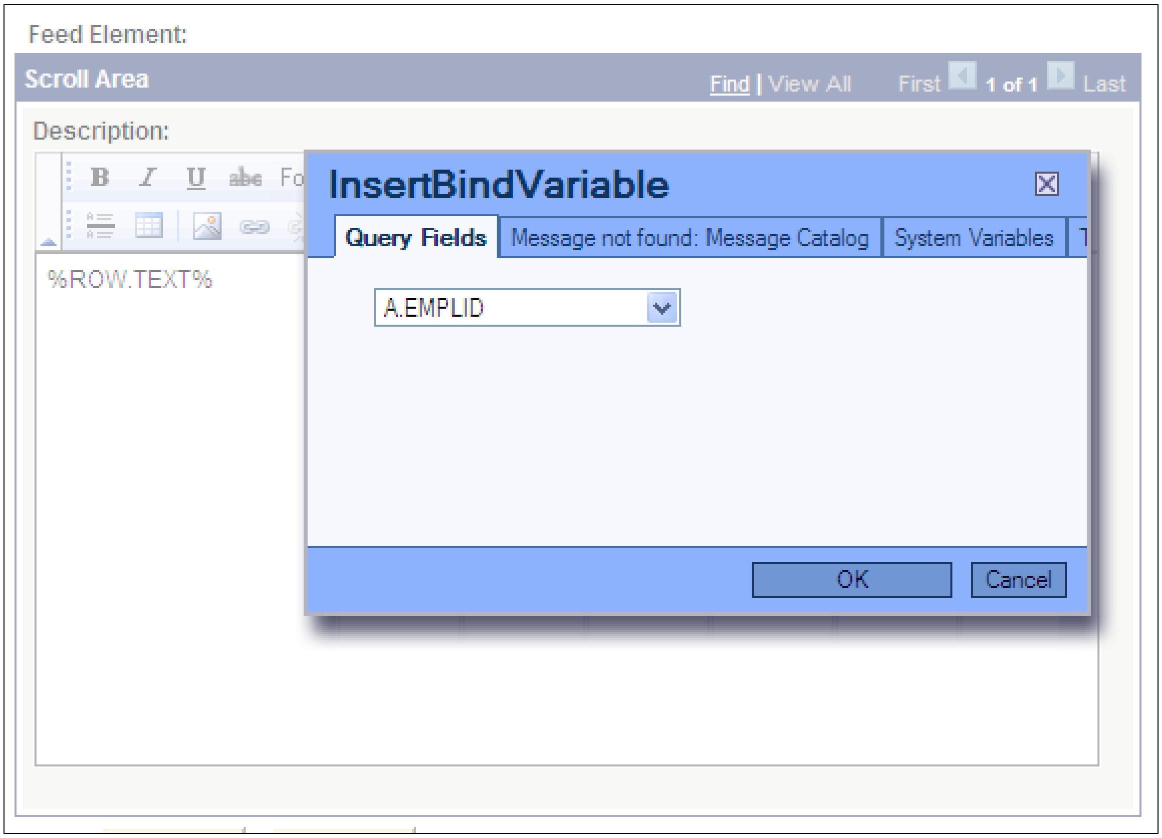
You can access these types of bind variables in the rich text editor:

- Query Fields
- Message Catalog
- System Variables
- Templates

Click the Insert Bind Variable button or right-click and select Insert Bind Variable to access the InsertBindVariable dialog box:

Image: Example of InsertBindVariable dialog box

This example illustrates the fields and controls on the InsertBindVariable dialog box.



Using Query Feeds

You can view query feeds by selecting either of the following navigation paths:

- Access the My Feeds page.
  1. Select My Feeds in the Main Menu.
  2. Enter search criteria and click Search.
  3. Select the desired query feed.

- Access Query Manager and use the related feeds hover menu.
  1. Select Reporting Tools, Query, Query Manager.
  2. Enter search criteria and click Search.
  3. Select a query that has already been published as a feed.
  4. From the query definition pages, select the feed from the related feeds hover menu.

Copying Query Feeds as Managed Objects

You can copy a query feed definition as managed objects from one system to another system.

Note: When query feed definitions are copied as managed objects between systems using Copy Project To/From file, ensure the query is present in the target system and edit the entry content URL in the Select Data Source page of Feed Publishing Wizard to point the feeds entry content link to target system instead of source system.

Chapter 8

Creating and Using Worklist Feeds

Understanding Worklist Feeds

PeopleTools provides the WORKLIST feed data type to publish worklist feeds through the Feed Publishing Framework.

Worklist feeds provide the ability to view a user's worklist as a feed in real time. The feed is generated in the security context of the signed on user and displays the user’s current worklist items; however, it does not show worked items. By using worklist feeds, the worklist items can be available on systems other than the system that originated the entries without copying the data.

If worklist items are hosted on another system, then you must complete additional configuration in the local database for the node that represents that system.

Additional Configuration for Worklist Feeds

The local host node that contains the Worklist and Worklist Details pages must have defined content uniform resource identifier (URI) text and portal URI text. This configuration data will be used to generate the worklist item URLs. To find the node name, navigate to the Worklist page or the Worklist Details page and look at the URL, which will be similar to the following:

```
http://machine:port/psp/ps/EMPLOYEE/ERP/w/WORKLIST?
ICAction=ICViewWorklist&Menu=Worklist&Market=GBL&PanelGroupName=WORKLIST.
```

In the previous URL, `ERP` is the local host node; therefore, you must define the content URI text and portal URI text for the `ERP` node so that worklist feeds contain valid feed entries.

See Setting URI Text for Local Host Nodes.

Publishing Worklist Feeds

This section discusses how to publish worklist feeds.

Understanding the Steps to Publish Worklist Feeds

You can use either of these methods to publish a worklist as a feed:

- Using the Publish as Feed pages.

  When you use the Publish as Feed pages to publish a worklist as a feed, you will have access to the Advanced Options page, which facilitates setting advanced feed options for this type of feed. However, you will be unable to personalize any data source parameters — that is, as user-specified or administrator-specified.

- Using the Feed Publishing Wizard.

  When you use the Feed Publishing Wizard to publish a worklist as a feed, you will be able to personalize data source parameters — that is, as user-specified or administrator-specified. However, it is more difficult to set advanced feed options through the Feed Publishing Wizard because they are set by value, and not by selecting a native language option.

Because there advantages and limitations of both methods of publishing a worklist as a feed, you might want to publish the feed using one methodology, and then use the other methodology to edit the feed definition.

Publishing Using the Publish as Feed Pages

To publish a worklist feed using the Publish as Feed pages:

Note: Any user with the role *PeopleSoft Administrator* can publish worklist feeds from either the Worklist page or the Worklist Details page.

1. Select Worklist > My Worklist - Summary View to access the Worklist page.
2. Click the Publish as Feed link.

   See Publishing Worklists as Feeds.
3. On the Publish Feed Definition page, define the feed properties.

   See Defining Feed Properties.
4. Click the Advanced Options link.
5. Define the advanced feed options for the worklist feed.

   See Defining Advanced Options for Worklist Feeds.
6. Click the OK button.
7. Click the Publish button.

Publishing Using the Feed Publishing Wizard

To publish a worklist feed using the Publish as Feed pages:

1. Select PeopleTools > Feeds > Feed Publishing Wizard.
2. Add a new value as the feed ID.
3. On the Specify Feed Properties page, define feed properties, such as the feed title, as well as additional feed properties.

   See Step 1: Specifying Feed Properties.
4. Click the Next button.
5. On the Select Data Source page, select *Worklist Feeds* as the feed data type.

   See Step 2: Selecting a Data Source.

6. Click the Next button.
7. On the Select Data Source Parameters page, set the usage type and default value for the data source parameters

   See Step 3: Specifying Data Source Parameters.

   If you set a parameter to be User Specified, click the Value link to specify the prompt values that are displayed to users when they personalize the feed; otherwise, continue with the next step.
8. Click the Next button.
9. On the Specify Publishing Options page, specify the publishing options for the feed, including feed security and sites.

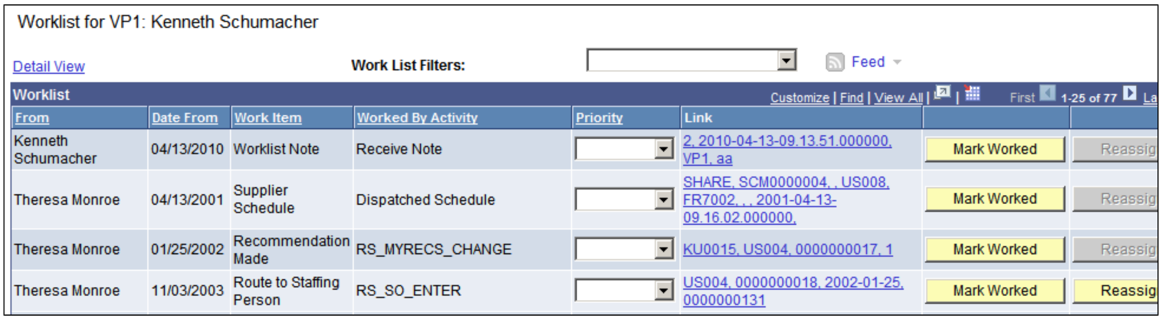
   See Step 4: Specifying Publishing Options.
10. Click the Save button to publish the feed.

Publishing Worklists as Feeds

Access the Worklist page or the Worklist Details page (Worklist, My Worklist - Summary View; or Worklist, My Worklist - Detail View).

Image: Worklist page

This example illustrates the fields and controls on the Worklist page.



Worklist for VP1: Kenneth Schumacher

Detail View

Work List Filters:

From	Date From	Work Item	Worked By Activity	Priority	Link		
Kenneth Schumacher	04/13/2010	Worklist Note	Receive Note		2, 2010-04-13-09.13.51.000000, VP1, aa	Mark Worked	Reassig
Theresa Monroe	04/13/2001	Supplier Schedule	Dispatched Schedule		SHARE, SCM0000004, , US008, FR7002, , , 2001-04-13-09.16.02.000000,	Mark Worked	Reassig
Theresa Monroe	01/25/2002	Recommendation Made	RS_MYRECS_CHANGE		KU0015, US004, 0000000017, 1	Mark Worked	Reassig
Theresa Monroe	11/03/2003	Route to Staffing Person	RS_SO_ENTER		US004, 0000000018, 2002-01-25, 0000000131	Mark Worked	Reassig

Use the Worklist page to publish a worklist as a feed. Click the Publish as Feed link to begin creating the feed definition.

Create the feed definition on the Publish Feed Definition page.

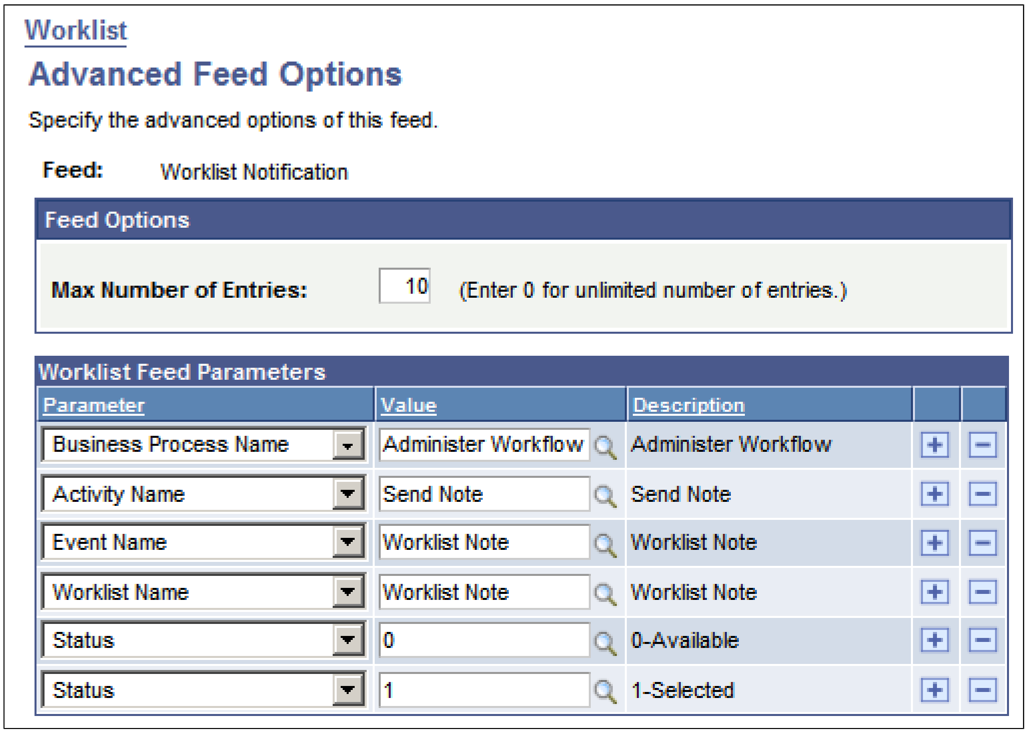
See Defining Feed Properties.

Defining Advanced Options for Worklist Feeds

Access the Advanced Feed Options page (click the Advanced Options link on the Publish Feed Definition page).

Image: Worklist - Advanced Feed Options page

This example illustrates the fields and controls on the Worklist - Advanced Feed Options page. You can find definitions for the fields and controls later on this page.



Worklist

Advanced Feed Options

Specify the advanced options of this feed.

Feed: Worklist Notification

Feed Options

Max Number of Entries: 10 (Enter 0 for unlimited number of entries.)

Worklist Feed Parameters

Parameter	Value	Description		
Business Process Name	Administer Workflow	Administer Workflow	+	−
Activity Name	Send Note	Send Note	+	−
Event Name	Worklist Note	Worklist Note	+	−
Worklist Name	Worklist Note	Worklist Note	+	−
Status	0	0-Available	+	−
Status	1	1-Selected	+	−

Use the Advanced Feed Options page to determine which worklists and worklist items are to be published in the worklist feed.

Max Number of Entries	Enter the maximum number of entries that the feed should return to the user. Enter *0* for unlimited entries up to the maximum row limit specified on the Feed Options page, which is 300 by default. **Note:** You can configure the upper limit on the Feed Options page.
Worklist Feed Parameters	Select the parameters and values for the worklists and worklist items that you want to publish. A status of *0* (available) or *1* (selected) will always be included in the feed parameters. After a worklist item is worked, it will no longer appear in the feed. **Note:** Even if rows for statuses 0 and 1 are deleted from the grid, worklist feeds will always show all available and selected worklist items.
Reset to Defaults	Click to reset advanced options to their default values.

Worklist Feed Parameters

This table lists worklist feed parameters, selectable values, and their data source parameter names:

Worklist Parameter	***Values***	***Data Source Parameter***
Activity Name	Select from any workflow activity.	ACTIVITYNAME
Business Process Name	Select from any business process.	BUSPROCNAME
Event Name	Select from any event.	EVENTNAME
From	Select from any user.	ORIGINATORID
Priority	*1* (high), *2* (medium), or *3* (low)	WL_PRIORITY
Status	*0* (available) or *1* (selected)	INSTSTATUS
Timed Out	*0* (available) or *1* (timed out)	TIMEDOUT
Worklist Name	Select from any worklist.	WORKLISTNAME

Note: The Priority parameter denotes the priority that the user that is the receiver of the notification has assigned to the worklist item in the Worklist page and not the priority with which the worklist item was created.

When you specify multiple feed parameters, similar parameters use OR, while distinct parameters use AND in the WHERE clause. For example, consider the following list of worklist feed parameters and runtime values:

Worklist Parameter	***Runtime Value***
Business Process Name	Administer Workflow
Activity Name	Send Note
Event Name	Worklist Note
Worklist Name	Worklist Note
Priority	1
Status	1
Status	0
Timed Out	0
From	PSADMIN
From	PTSECADM

The WHERE clause for the SQL statement generated for the parameters shown in the preceding table is:

```
WHERE (BUSPROCNAME='Administer Workflow') AND (ACTIVITYNAME ='Send Note') AND  (EVE⇒
NTNAME='Worklist Note') AND (WORKLISTNAME='Worklist Note') AND (WL_PRIORITY='1') AN⇒
D ((INSTSTATUS=0) OR (INSTSTATUS=1)) AND (TIMEDOUT=0) AND((ORIGINATORID='PSADMIN') ⇒
OR (ORIGINATORID='PTSECADM'))
```

This table describes how the Feed Publishing Framework automatically maps the feed properties to Atom 1.0 data source parameters:

Feed Property	***Data Source Parameter***
Author	ORIGINATORID and primary email address
Category	BUSPROCNAME, ACTIVITYNAME, EVENTNAME, WORKLISTNAME, WL_PRIORITY, TIMEDOUT, INSTSTATUS, ORIGINATORID
ContentURL	Worklist entry URL
Copyright	None
Contributor	None
Description	BUSPROCNAME | "/" | ACTIVTYNAME | "/" | EVENTNAME Created On: INSTAVAILABLEDTTM Instance ID: INSTANCEID From: ORIGINATORID
Enclosure	None
FullContent	None
GUID	Worklist URL | " " | INSTAVAILABLEDTTM
Published	INSTAVAILABLEDTTM
Title	INSTANCEID | "." | BUSPROCNAME | "/" | ACTIVTYNAME | "/" | EVENTNAME
Updated	LASTUPDDTTM

Note: INSTAVAILABLEDTTM is the time at which the worklist item was created. LASTUPDDTTM is the time at which the worklist item was last updated or selected.

Using Worklist Feeds

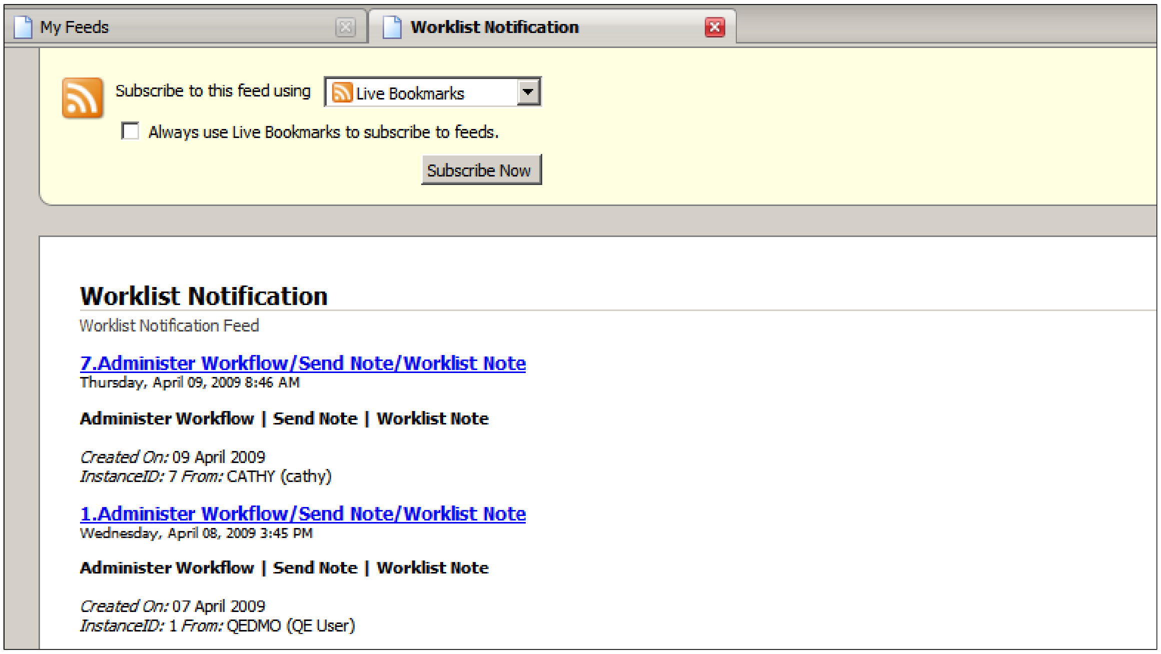
You can view the feeds by selecting any of the following navigation paths:

- Access the My Feeds page:
  1. Select My Feeds in the Main Menu.
  2. Enter search criteria and click Search.
  3. Select the desired worklist feed.
- Access the Worklist page:
  1. Select Worklist > My Worklist - Summary View.
  2. Select a worklist feed from the related feeds hover menu.
- Access the Worklist Details page:
  1. Select Worklist > My Worklist - Detail View.
  2. Select a worklist feed from the related feeds hover menu.

The following example shows a feed titled Worklist Notification in the browser:

Image: Example of a worklist feed

This example illustrates a worklist feed titled Worklist Notification.



Subscribe to this feed using Live Bookmarks

□ Always use Live Bookmarks to subscribe to feeds.

Worklist Notification

Worklist Notification Feed

7.Administer Workflow/Send Note/Worklist Note
Thursday, April 09, 2009 8:46 AM

Administer Workflow | Send Note | Worklist Note

Created On: 09 April 2009
InstanceID: 7 *From:* CATHY (cathy)

1.Administer Workflow/Send Note/Worklist Note
Wednesday, April 08, 2009 3:45 PM

Administer Workflow | Send Note | Worklist Note

Created On: 07 April 2009
InstanceID: 1 *From:* QEDMO (QE User)

When you click a worklist item, the appropriate page will appear.

Chapter 9

Developing New Feed Data Types

Analyzing Requirements for New Feed Data Types

When developing a new feed data type, you should consider these issues:

- Decide how to distinguish feeds of the same data type, which you will implement as data source settings for the feed. For example, the data source setting for a discussion forum is the forum ID; the data source settings for a content management folder are the portal name and folder ID.
- Decide how you will configure the feeds and what the default value is for each data source parameter. For example, the data source parameter for a discussion forum is the maximum number of entries with a default of *10.* The data source parameters for a content management folder are the maximum number of entries with a default of *10* and an include subfolder flag with a default of *yes.*
- Decide what data will be published as feed entries. For example, a discussion forum publishes the complete post, author, tags, attachment, and timestamp in each feed entry; a content management folder publishes the content summary, author, tags, attachment, and timestamp in each feed entry.
- Decide whether you want the feed to be generated as a scheduled feed or as a real-time feed.
- Decide who will have the authority to create and manage feed definitions.
- (Optional) Decide where you will locate the Publish as Feed link.

  Note: Because any feed definition can be published through the Feed Publishing Wizard, it is optional to update the property maintenance component to add the Publish as Feed pages.
- Decide where you will locate the related feeds hover menu.
- Decide how you want to handle real-time feed security and GETFEED viewer permission requests.

Creating the Feed Data Source Application Class

The feed data source application class manages all aspects of data collection and data security. This application class encapsulates all business logic about how to define feed definitions and how to gather feed data.

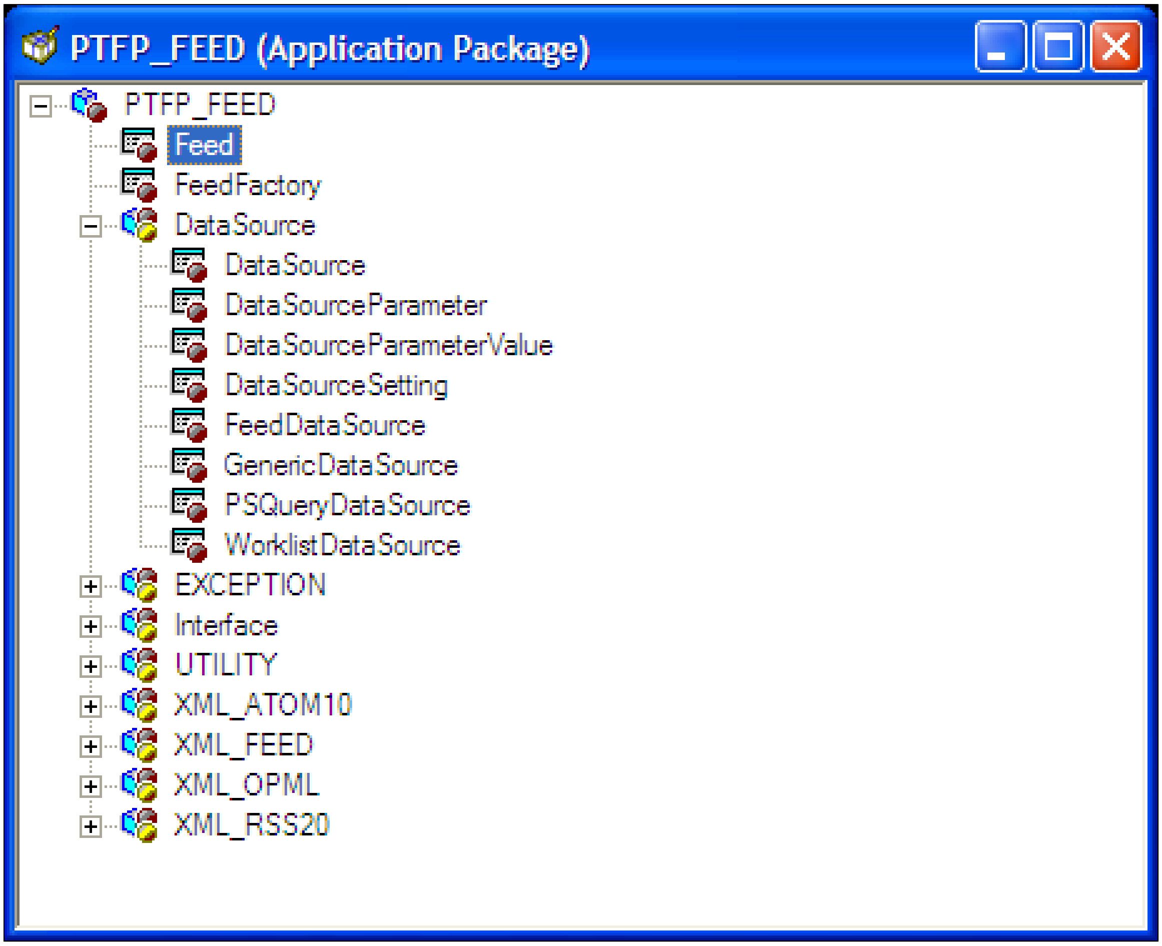
This section discusses the steps involved to create a feed data source application class for your new feed data type.

Extending the PTFP_FEED:DataSource:DataSource Base Class

Implement a new class that extends the PTFP_FEED:DataSource:DataSource base class.

Image: PTFP_FEED application package showing the DataSource base class

This example illustrates the DataSource base class in the PTFP_FEED application package.



PTFP_FEED (Application Package)

- PTFP_FEED
  - Feed
  - FeedFactory
  - DataSource
    - DataSource
    - DataSourceParameter
    - DataSourceParameterValue
    - DataSourceSetting
    - FeedDataSource
    - GenericDataSource
    - PSQueryDataSource
    - WorklistDataSource
  - EXCEPTION
  - Interface
  - UTILITY
  - XML_ATOM10
  - XML_FEED
  - XML_OPML
  - XML_RSS20

See "Creating Application Packages" (PeopleTools 8.59: PeopleCode Developer’s Guide).

Implementing the DataSource Methods

When implementing the methods of the DataSource class, consider these method types:

- Required methods
- Recommended methods
- Optional methods

Required Methods

This table describes the DataSource methods that you must implement:

Method	***Purpose***
`clone`	Clone the data source object.
`getContentUrl`	Return the feed content URL.
`getDataSecurity`	Return the *allowed* list of viewer roles, permission lists, or both.
`isCurrentUserAuthorized`	Validate whether the current user has permission to view the feed.
`initializeSettings`	Initialize the data source setting collection and other class properties.
`processSettingsChange`	Validate data source setting values and generate a data source parameter list accordingly.
`execute`	Collect data based on user permissions and fill in the feed document.

Recommended Methods

This table describes the DataSource methods that you should consider implementing:

Method	***Purpose***
`isCurrentUserAdmin`	Validate whether the current user has permission to administer the feed.
`copyProperties`	The clone method uses protected methods for copying class properties.

Optional Methods

This table describes the DataSource methods that you might consider implementing:

Method	***Purpose***
`onSave`	Perform tasks after saving the feed definition.
`onDelete`	Perform tasks before deleting the feed definition.
`getSettingDetail`	Return data source setting details as HTML.
`getParameterDetail`	Return data source parameter details as HTML.

Setting Read-Only Flags by Using Protected Methods

You should use these protected methods to set read-only flags:

- `setDataSourceType`
- `setSettingsCompleted`
- `setAllowRealTimeFeedSecurity`

Creating the Feed Data Type Definition

This section lists the steps for creating a new feed data type definition and discusses how to:

- Define a new feed data type.
- Determine whether additional advanced options are available.

Steps for Creating a New Feed Data Type Definition

You complete these steps to define a new feed data type:

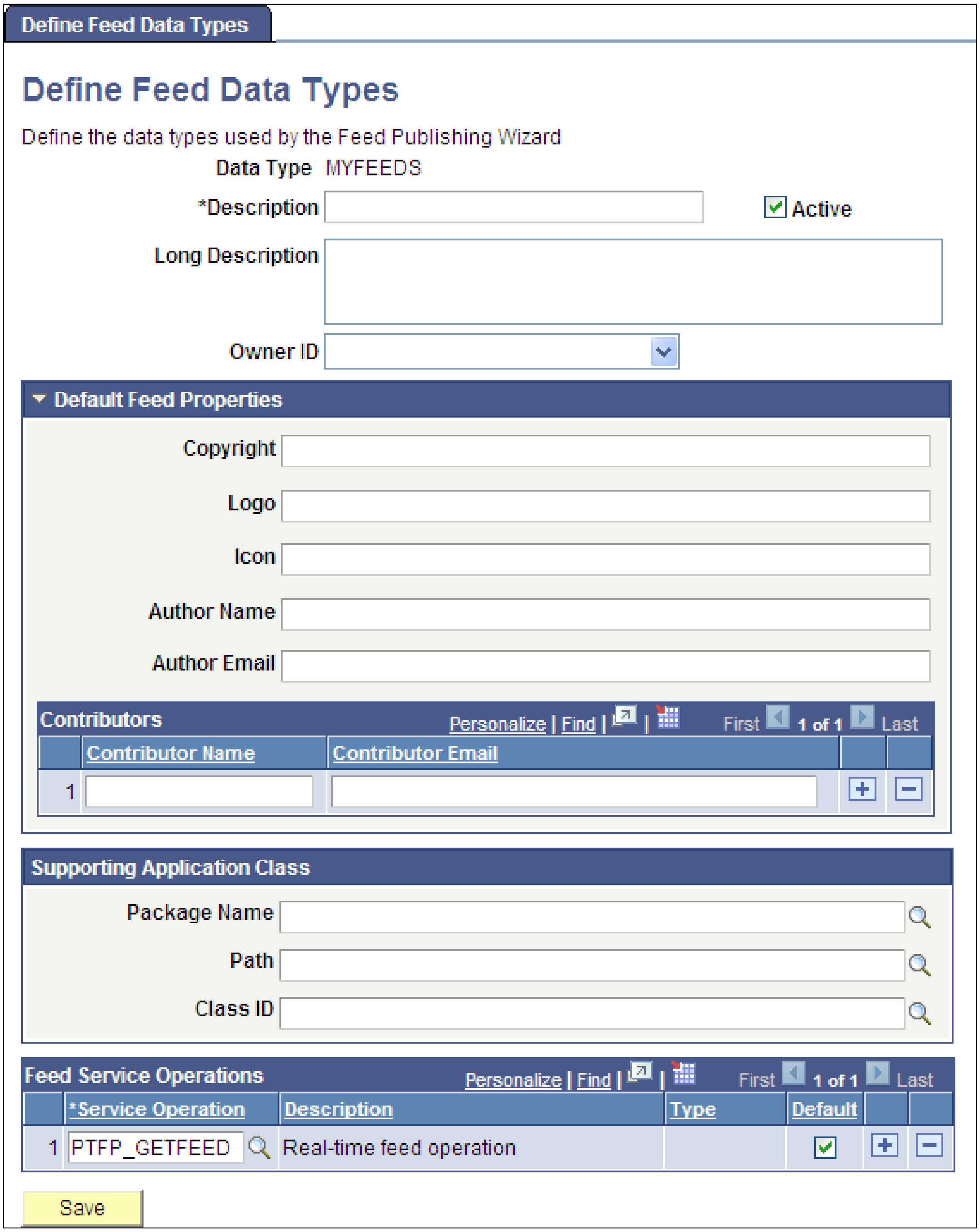
1. Select PeopleTools > Feed > Define Data Types.
2. Specify the data source application class.
3. Select the service operations to be used by feeds of this data type and specify the default service operation.
4. Specify the default feed head level attributes for feeds of this data type.
5. (Optional) Click the Publish as Feed link to create a list of feeds feed, which lists all feeds of this data type that the user can access.
6. Determine whether additional advanced options are available.

Creating a New Feed Data Type Definition

Access the Define Feed Data Types page (PeopleTools, Feeds, Define Feed Data Types).

Image: Define Feed Data Types page

This example illustrates the fields and controls on the Define Feed Data Types page. You can find definitions for the fields and controls later on this page.



Define Feed Data Types

Define Feed Data Types

Define the data types used by the Feed Publishing Wizard

Data Type MYFEEDS

*Description ☑ Active

Long Description

Owner ID

Default Feed Properties

Copyright

Logo

Icon

Author Name

Author Email

Contributors

	Contributor Name	Contributor Email		
1			+	−

Supporting Application Class

Package Name

Path

Class ID

Feed Service Operations

	*Service Operation	Description	Type	Default		
1	PTFP_GETFEED	Real-time feed operation		☑	+	−

Save

PeopleTools delivers the following feed data types: FEED, GENERICFEED, PSQUERY, PTSF_SES_FEED_DT, and WORKLIST. To create a new feed data type, use the Add New Value field on the search page.

Data Type

Displays the type of feed that you are creating or editing. This field is display-only.

Description

Enter a short description of the type of feed. You may enter up to 30 characters.

Long Description

Enter a long description of the type of feed to clearly describe its purpose. You may enter up to 255 characters.

Owner ID

Select the owner ID for the feed data type definition.

The owner ID is a way to identify which definitions are owned by which PeopleSoft applications, such as PeopleSoft General Ledger, Accounts Receivables, PeopleTools, and so on. The values in the drop-down list box are Translate table values associated with the OBJECTOWNERID field.

Active

Select to activate the feed definition.

Default Feed Properties

Not all readers display all properties. This table describes the default feed properties that some feed readers process and display.

Note: These properties are the defaults. You can change any of these default properties at the individual feed level.

Copyright

Enter copyright information to be included in the XML.

Logo

Enter a URL to the logo to be included in the XML, for example, *http://example.com/img/logo.gif*.

Icon

Enter a URL to an icon to be included in the XML, for example, *http://example.com/img/icon.gif*.

Author Name

Enter an author name to be included in the XML.

Author Email

Enter an author's email address to be included in the XML.

Contributor Name

Enter a contributor's name to be included in the XML.

Contributor Email

Enter a contributor's email address to be included in the XML.

Supporting Application Class

Package Name

Enter the application class package name that you want to use for the data type.

Each Feed Data Type application class should be associated with one Feed Data Type service operation.

Path

Enter the application class path that you want to use for the data type.

Application Class ID

Enter the name of the application class that you want to use for the data type. The class must exist in the application package name that you specify.

Feed Service Operations

Service Operation

Enter the name of the service operations associated with the feed definition that are used to retrieve data.

Type

Displays whether the service operation is real-time or scheduled. This field is display-only.

Default

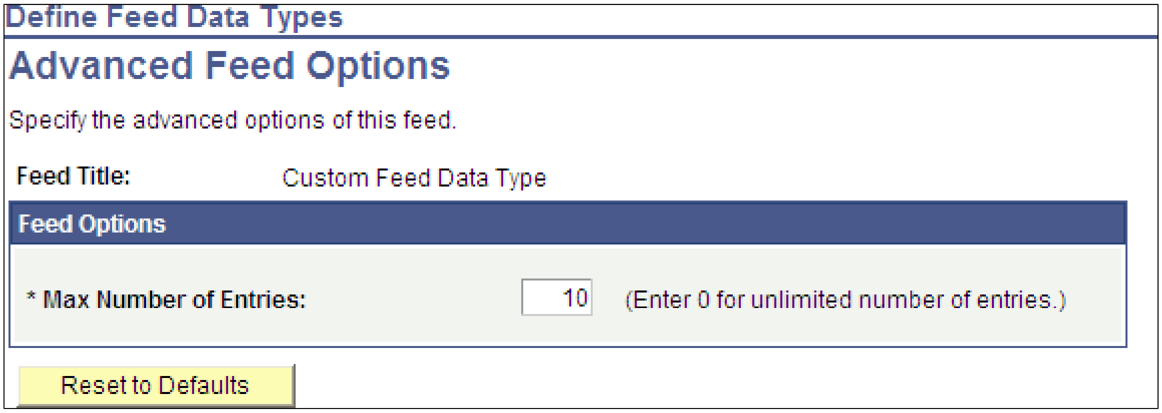
Select this check box to make this service operation the default.

Determine Whether Additional Advanced Options Are Available

The standard advanced option page, PTFP_PUB_AS_ADVOPT, has one advanced option: Max Number of Entries. If your new feed data type has additional data source parameters (such as a paged feed, an incremental feed, or other parameters), then you must create a custom advanced options page. Otherwise, you can use the standard advanced options page shown in this example:

Image: Example of the standard Advanced Feed Options page

This example illustrates the fields and controls on the standard Advanced Feed Options page.



Define Feed Data Types

Advanced Feed Options

Specify the advanced options of this feed.

Feed Title: Custom Feed Data Type

Feed Options

* Max Number of Entries: 10 (Enter 0 for unlimited number of entries.)

Related Links

Creating an Advanced Options Page

(Optional) Updating the Property Maintenance Component

Because any feed definition can be published through the Feed Publishing Wizard, it is optional to update the property maintenance component to add the Publish as Feed pages. If you decide to update the property maintenance component, then content and feed administrators can use either methodology to publish feeds for this type of data. To update property maintenance components, you complete the tasks described in this section.

This section discusses how to update the property maintenance components.

Adding the Four Standard Publish as Feed Pages

You must add these four standard Publish as Feed pages to the component as hidden pages:

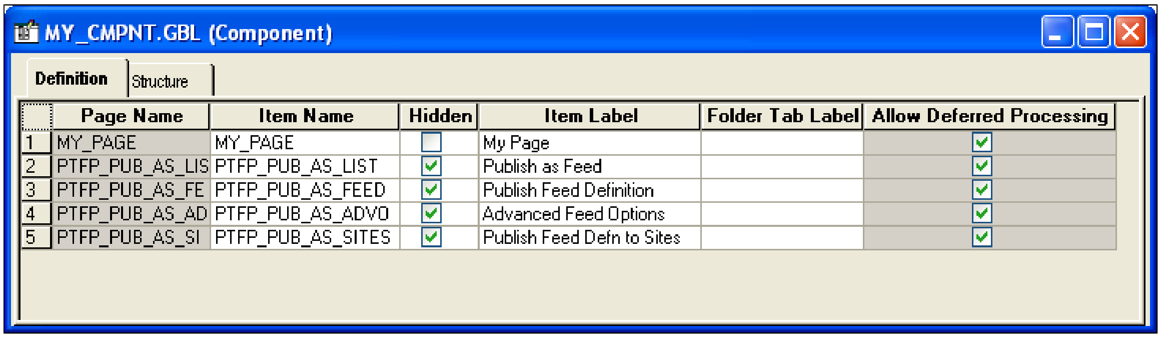
- Publish Feed Definition (PTFP_PUB_AS_FEED)
- Advanced Feed Options (PTFP_PUB_AS_ADVOPT)
- Publish as Feed (PTFP_PUB_AS_LIST)
- Publish Feed Definition to Sites (PTFP_PUB_AS_SITES)

Note: You can clone and then modify all four pages to suit the unique requirements of the new feed data type.

Access the component, add the pages, and configure the pages as hidden, as shown in this example:

Image: Example of MY_CMPNT component showing the four hidden, standard Publish as Feed pages

This example illustrates the MY_CMPNT component that has the four hidden, standard Publish as Feed pages.



MY_CMPNT.GBL (Component)

	Page Name	Item Name	Hidden	Item Label	Folder Tab Label	Allow Deferred Processing
1	MY_PAGE	MY_PAGE	☐	My Page		☑
2	PTFP_PUB_AS_LIS	PTFP_PUB_AS_LIST	☑	Publish as Feed		☑
3	PTFP_PUB_AS_FE	PTFP_PUB_AS_FEED	☑	Publish Feed Definition		☑
4	PTFP_PUB_AS_AD	PTFP_PUB_AS_ADVO	☑	Advanced Feed Options		☑
5	PTFP_PUB_AS_SI	PTFP_PUB_AS_SITES	☑	Publish Feed Defn to Sites		☑

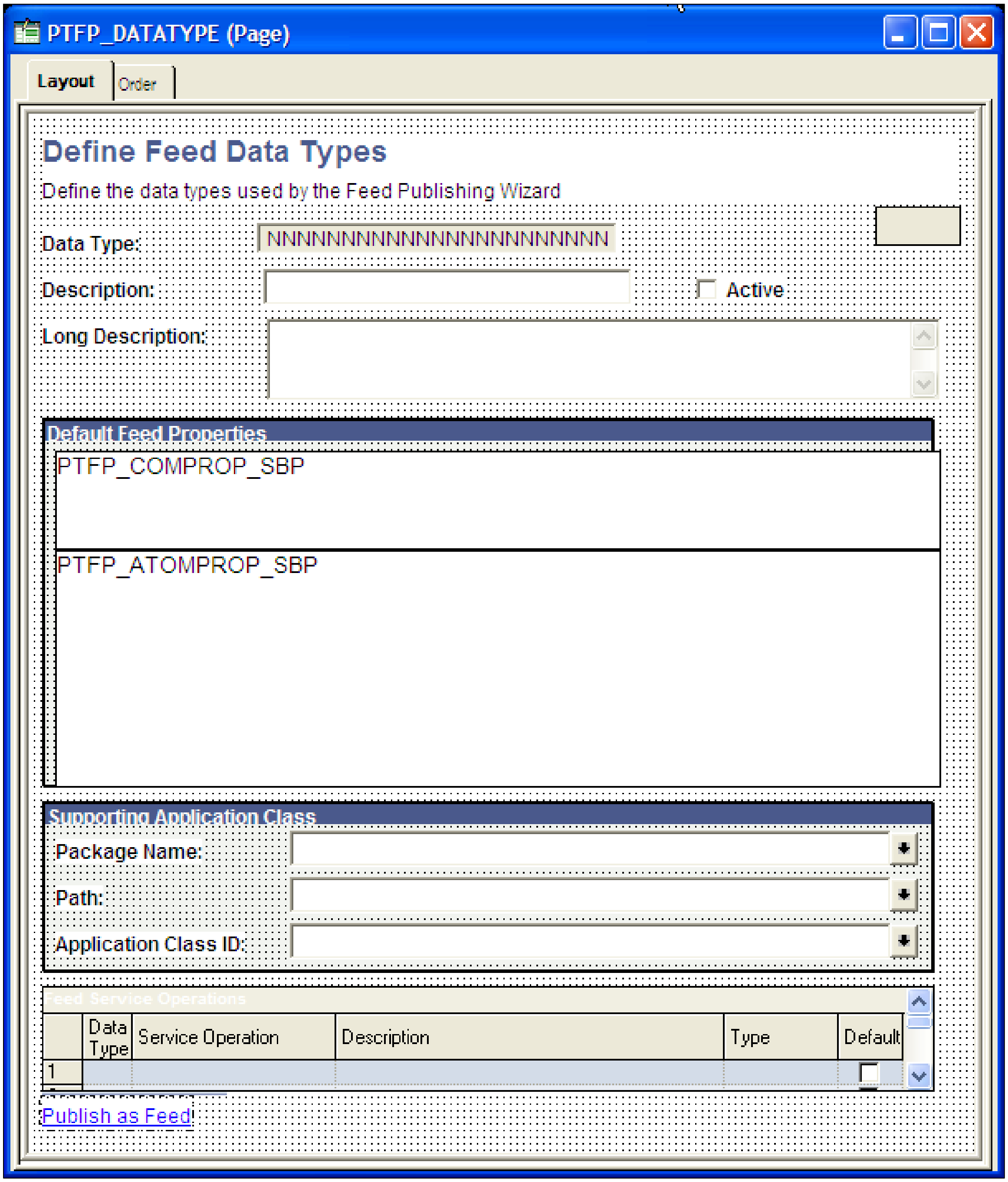
See "Adding Pages to Components" (PeopleTools 8.59: Application Designer Developer's Guide).

Adding the Publish as Feed Link to a Page in the Component

To publish the new feed data type, the feed administrator must have access to the Publish as Feed pages. To access these pages, you must add the Publish as Feed link to a page in the component.

Image: PTFP_DATATYPE page showing the Publish as Feed link

This example illustrates the fields and controls on the PTFP_DATATYPE page showing the Publish as Feed link.



Define Feed Data Types

Define the data types used by the Feed Publishing Wizard

Data Type: NNNNNNNNNNNNNNNNNNNNNN

Description: □ Active

Long Description:

Default Feed Properties

PTFP_COMPROP_SBP

PTFP_ATOMPROP_SBP

Supporting Application Class

Package Name:

Path:

Application Class ID:

Feed Service Operations

	Data Type	Service Operation	Description	Type	Default
1					□

Publish as Feed

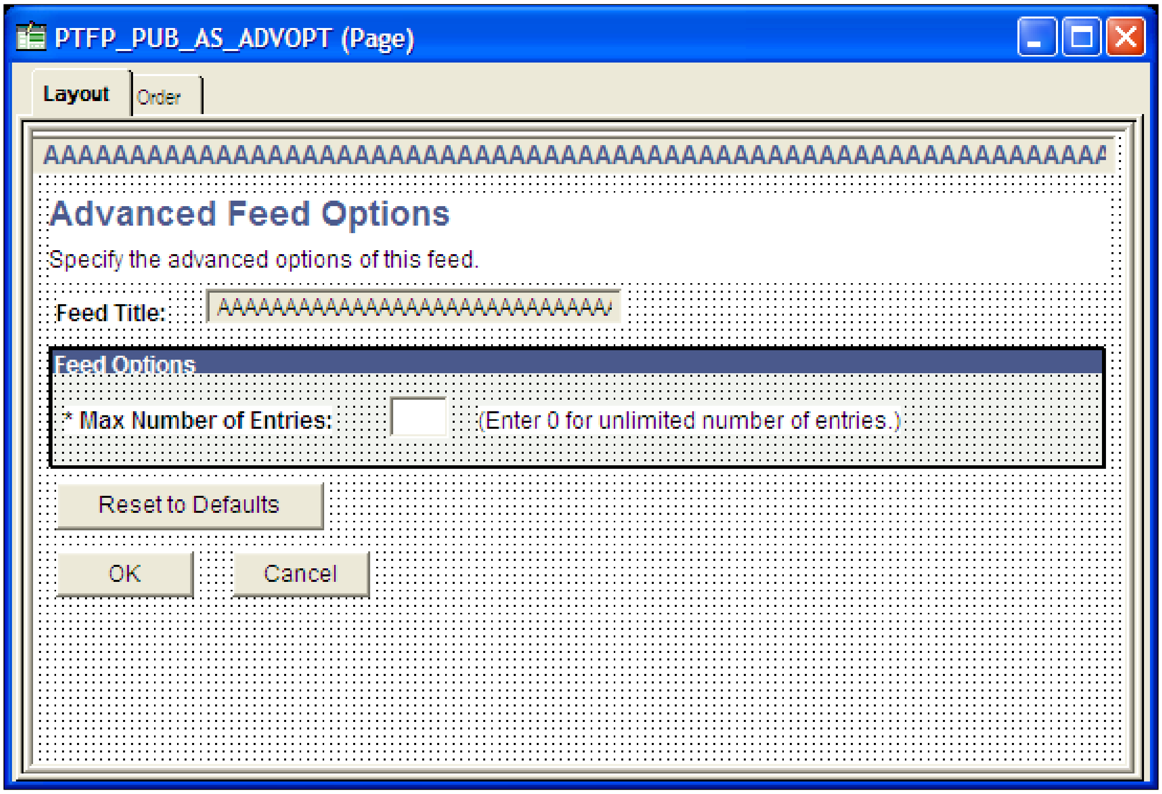
See "Using Push Buttons and Links" (PeopleTools 8.59: Application Designer Developer's Guide).

Creating an Advanced Options Page

The advanced options page is used to set data source parameters for each feed definition. The standard advanced option page, PTFP_PUB_AS_ADVOPT, has one advanced option: Max Number of Entries. If your new feed data type has additional data source parameters (such as a paged feed, an incremental feed, or other parameters), then you must create a custom advanced options page. Otherwise, you can use the standard advanced options page shown in this example:

Image: Example PTFP_PUB_AS_ADVOPT page (the standard advanced options page)

This example illustrates the fields and controls on the PTFP_PUB_AS_ADVOPT page.



PTFP_PUB_AS_ADVOPT (Page)

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Advanced Feed Options

Specify the advanced options of this feed.

Feed Title: AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

Feed Options

* Max Number of Entries: (Enter 0 for unlimited number of entries.)

To create a custom advanced options page:

1. Clone the PTFP_PUB_AS_ADVOPT page as a feed data type-specific advanced options page.

2. Add the feed data type-specific data source parameters to the page.

3. In the page Activate event, create a PeopleCode program to read the data source parameter values from the feed definition.

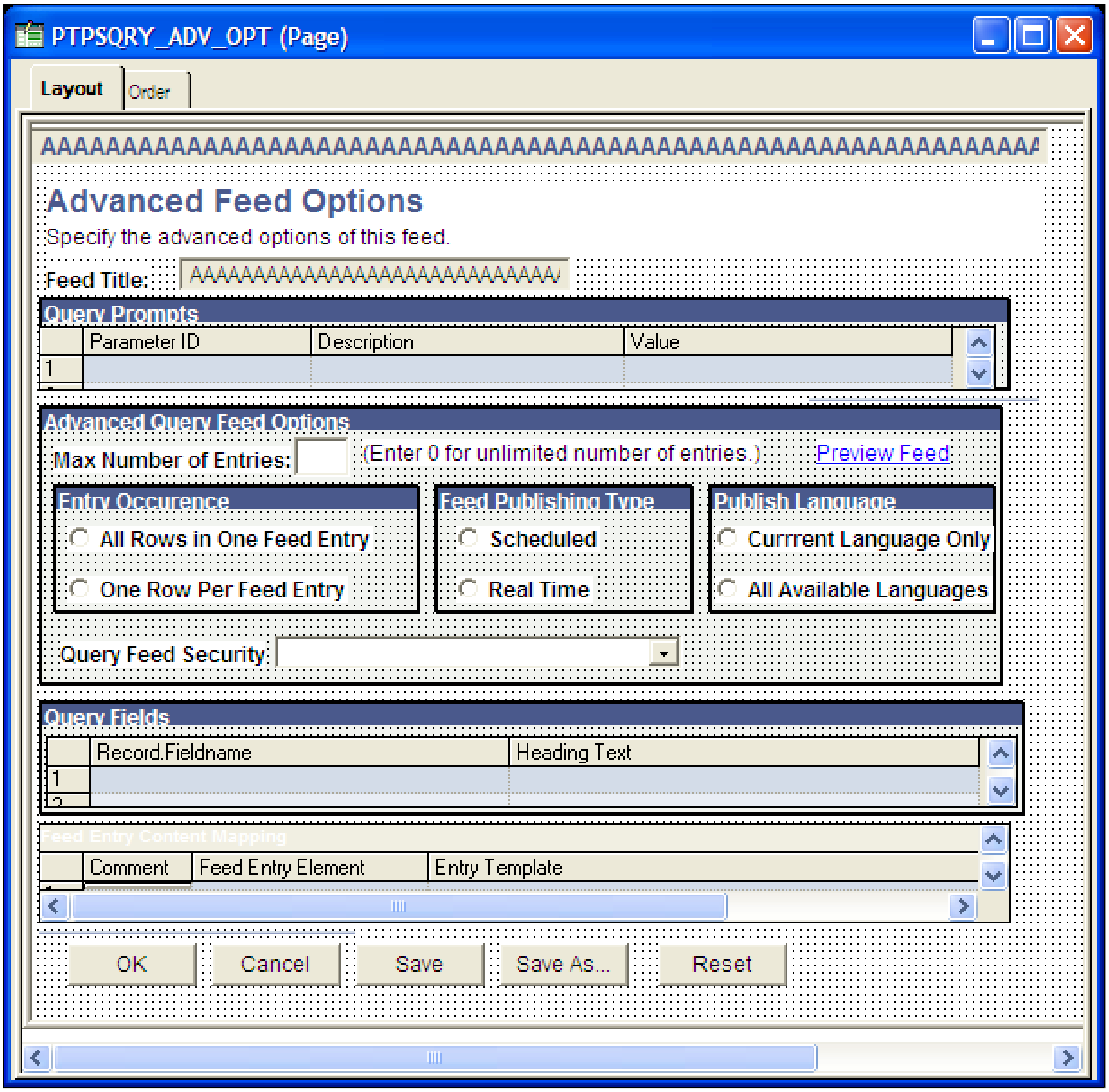
4. Create an additional PeopleCode program to set the data source parameter values to the feed definition when the page is closed.

Example

This is the Query Advanced Options page. Notice how this page differs from the standard page.

Image: Query feeds advanced options page

This example illustrates the fields and controls on the Query feeds advanced options page.



Related Links

"Understanding Page Design and Development" (PeopleTools 8.59: Application Designer Developer's Guide)

Adding Record PeopleCode

Add the following code to the FieldChange event for the “Publish as Feed” component record field:

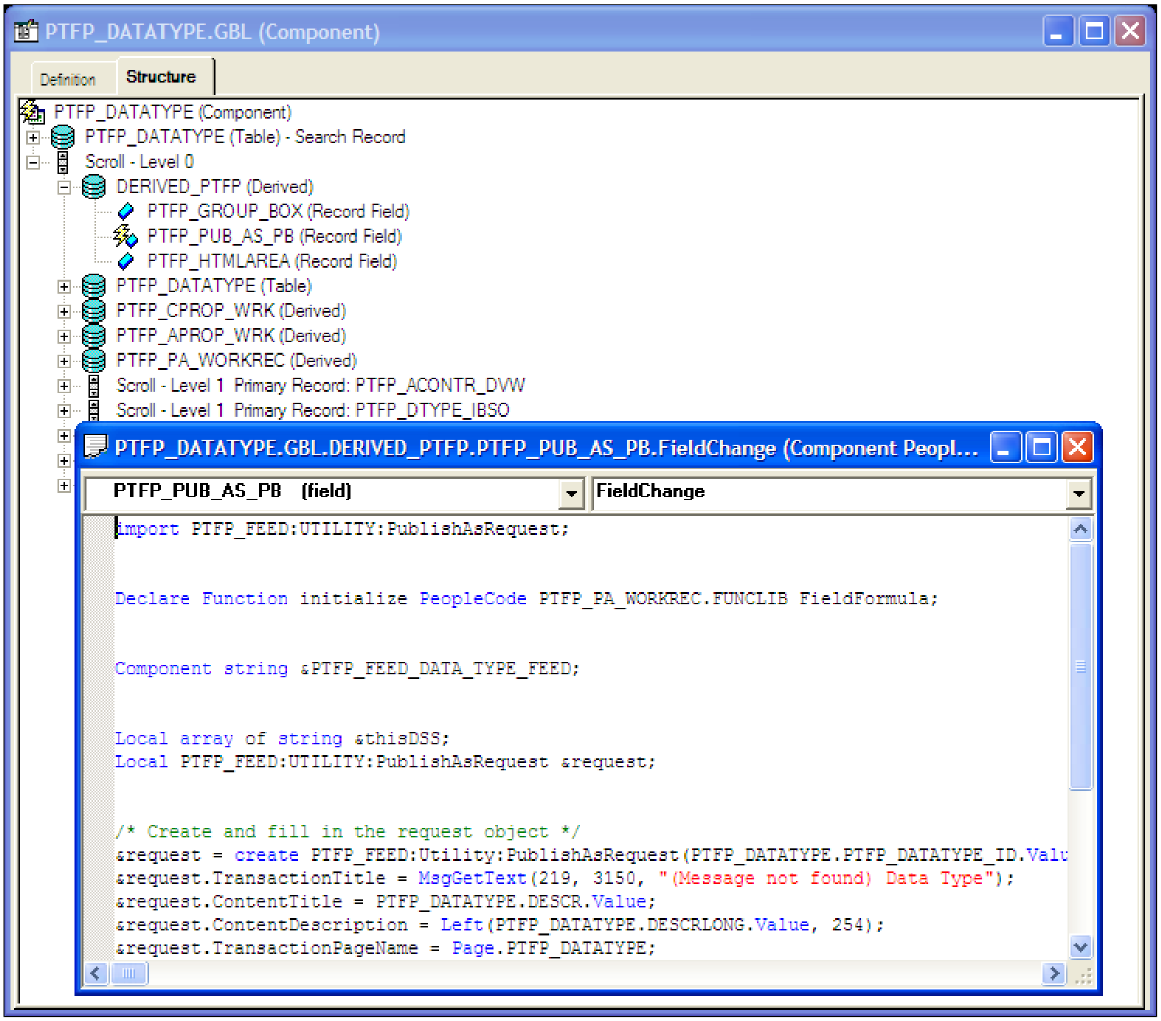
```
import PTFP_FEED:UTILITY:PublishAsRequest;
Declare Function initialize PeopleCode PTFP_PA_WORKREC.FUNCLIB FieldFormula;
Local PTFP_FEED:UTILITY:PublishAsRequest &request;
Local array of string &thisDSS;

/* Create and fill in the request object */
```

```
&request = create PTFP_FEED:Utility:PublishAsRequest("unique ID");

&request.TransactionPageName = Page.PAGE_NAME;
&request.TransactionTitle = "Page Title";
&request.ContentTitle = "Default Feed Title";
&request.ContentDescription = "Default Feed Description";
&request.AdvancedOptionsPageName = Page.PAGE_NAME;
&request.DataTypeID = "yourDataTypeID";

/* Fill in the data source setting values */
&thisDSS = CreateArray("Data Source Setting Name", "Data Source Setting Value");
&request.DataSourceSettings.Push(&thisDSS);

/* Start the process */
initialize(&request);
```

Example

Image: PTFP_PUB_AS_PB field showing PeopleCode for the FieldChange event

This example illustrates the PTFP_PUB_AS_PB field showing PeopleCode for the FieldChange event.


PTFP_DATATYPE.GBL (Component)
Definition
Structure
PTFP_DATATYPE (Component)
PTFP_DATATYPE (Table) - Search Record
Scroll - Level 0
DERIVED_PTFP (Derived)
PTFP_GROUP_BOX (Record Field)
PTFP_PUB_AS_PB (Record Field)
PTFP_HTMLAREA (Record Field)
PTFP_DATATYPE (Table)
PTFP_CPROP_WRK (Derived)
PTFP_APROP_WRK (Derived)
PTFP_PA_WORKREC (Derived)
Scroll - Level 1 Primary Record: PTFP_ACONTR_DVW
Scroll - Level 1 Primary Record: PTFP_DTYPE_IBSO
PTFP_DATATYPE.GBL.DERIVED_PTFP.PTFP_PUB_AS_PB.FieldChange (Component Peopl...
PTFP_PUB_AS_PB (field)
FieldChange

```
import PTFP_FEED:UTILITY:PublishAsRequest;


Declare Function initialize PeopleCode PTFP_PA_WORKREC.FUNCLIB FieldFormula;


Component string &PTFP_FEED_DATA_TYPE_FEED;


Local array of string &thisDSS;
Local PTFP_FEED:UTILITY:PublishAsRequest &request;


/* Create and fill in the request object */
&request = create PTFP_FEED:Utility:PublishAsRequest(PTFP_DATATYPE.PTFP_DATATYPE_ID.Valu
&request.TransactionTitle = MsgGetText(219, 3150, "(Message not found) Data Type");
&request.ContentTitle = PTFP_DATATYPE.DESCR.Value;
&request.ContentDescription = Left(PTFP_DATATYPE.DESCRLONG.Value, 254);
&request.TransactionPageName = Page.PTFP_DATATYPE;
```

Note: Where you add the record PeopleCode depends on which record field is used for the Publish as Feed page. If you use the standard page, DERIVED_PTFP.PTFP_PUB_AS_PB, then you should only add code in the component record field FieldChange event. If the field is in your own work record, you can use the record FieldChange event directly.

Related Links

"Accessing Record Field PeopleCode" (PeopleTools 8.59: PeopleCode Developer's Guide)

"Accessing Component Record Field PeopleCode" (PeopleTools 8.59: PeopleCode Developer's Guide)

Updating the View Content Component or Pagelet

This section discusses how to update the view content component or pagelet.

This task applies equally to feeds published with the Feed Publishing Wizard as well as to feeds published using the Publish as Feed pages. Completing the items in this task will make it easier for your users to discover published feeds.

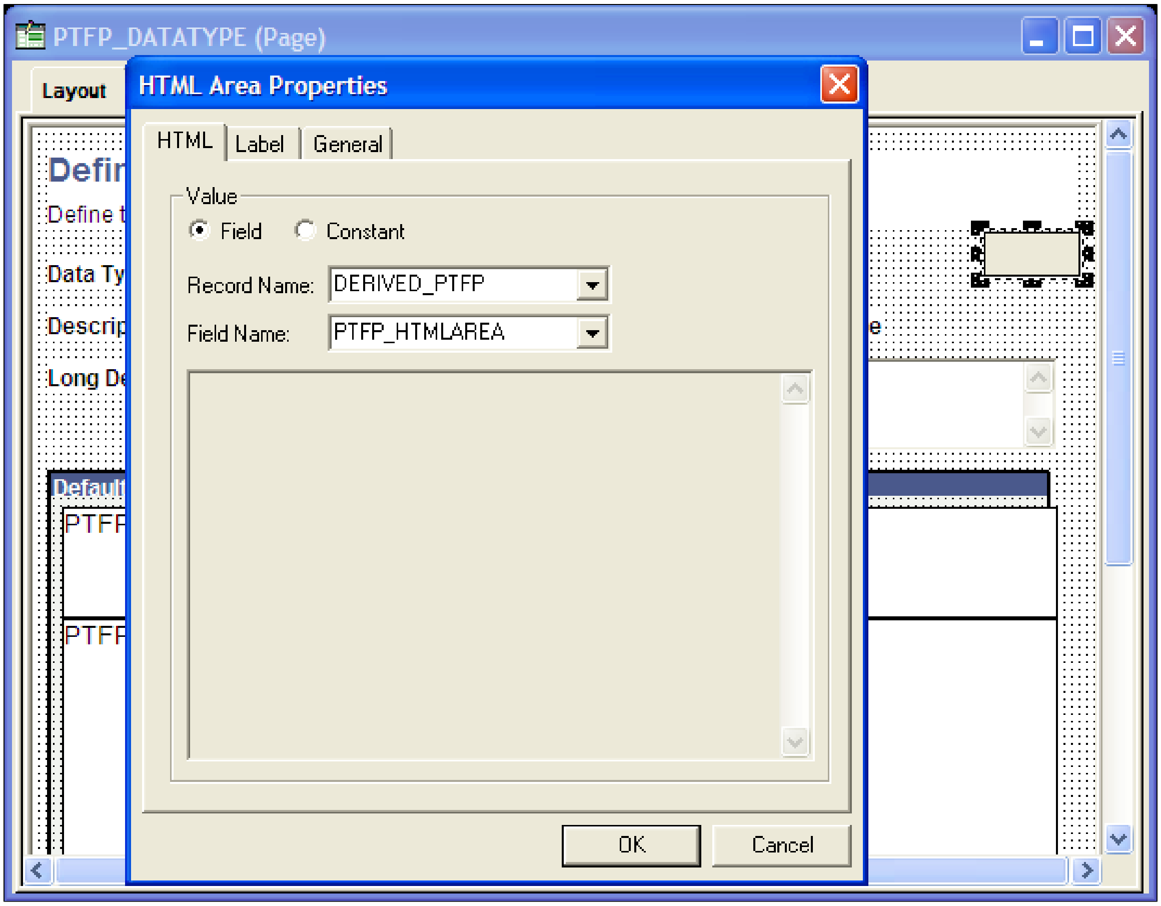
Adding the Related Feeds Hover Menu to Pages

Add an HTML area to the page for hosting the related feeds hover menu.

Example

Image: PTFP_DATATYPE page showing HTML area and HTML area properties dialog box

This example illustrates the PTFP_DATATYPE page showing HTML area and HTML area properties dialog box.



Value: ● Field ○ Constant

Record Name: DERIVED_PTFP

Field Name: PTFP_HTMLAREA

Related Links

"Using HTML Areas" (PeopleTools 8.59: Application Designer Developer's Guide)

Adding Component or Page PeopleCode

Add the following PeopleCode to the Activate event for the page:

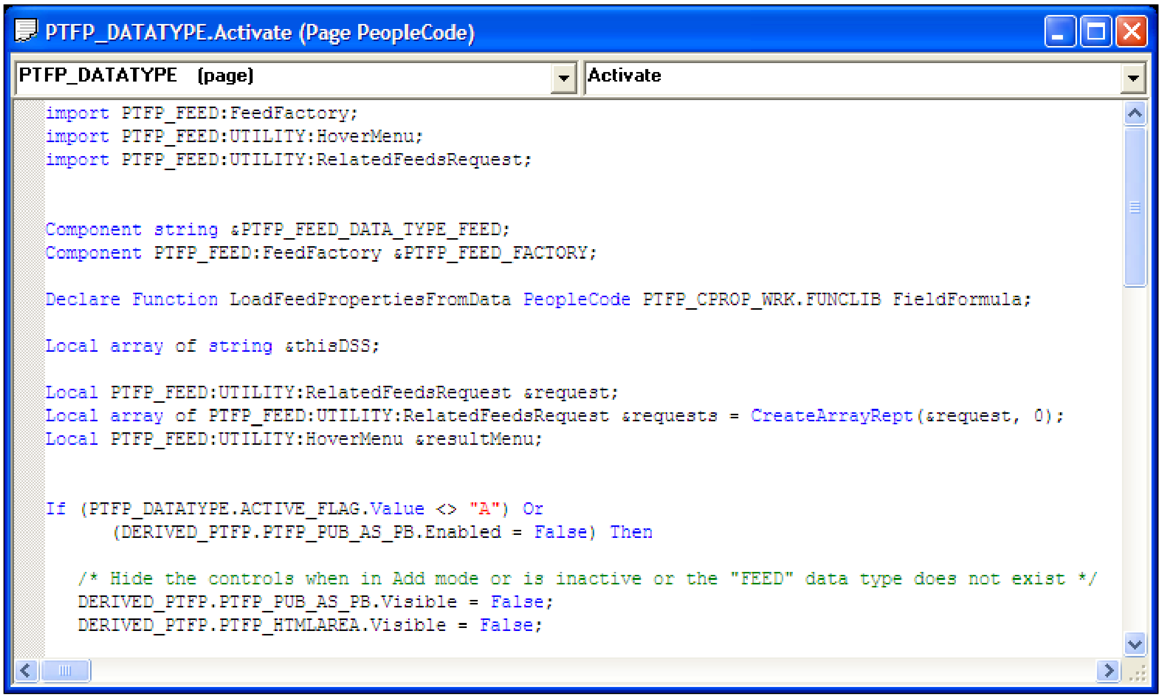
```
import PTFP_FEED:FeedFactory;
import PTFP_FEED:UTILITY:HoverMenu;
import PTFP_FEED:UTILITY:RelatedFeedsRequest;

Local PTFP_FEED:FeedFactory &PTFP_FEED_FACTORY;
Local PTFP_FEED:UTILITY:RelatedFeedsRequest &request;
Local array of PTFP_FEED:UTILITY:RelatedFeedsRequest &requests;
Local PTFP_FEED:UTILITY:HoverMenu &resultMenu;

&PTFP_FEED_FACTORY = create PTFP_FEED:FeedFactory();

/* Fill in the search criteria */
&requests = CreateArrayRept(&request, 0);
&request = create PTFP_FEED:UTILITY:RelatedFeedsRequest("unique ID");
&request.DataTypeID = "yourDataTypeID";
&request.DataSourceSettings.Push(CreateArray("Data Source Setting Name", "Data Sour⇒
ce Setting Value"));
&requests.Push(&request);

try
   /* Generate the menu */
   &resultMenu = &PTFP_FEED_FACTORY.getRelatedFeedsHoverMenu(&requests);
   DERIVED_PTFP.PTFP_HTMLAREA.Value = &resultMenu.getHtml();
catch Exception &e
   WinMessage(&e.ToString(), %MsgStyle_OK);
end-try;
```

See "Accessing Page PeopleCode" (PeopleTools 8.59: PeopleCode Developer’s Guide).

Example

Image: Example of PTFP_DATATYPE Page PeopleCode

This example illustrates Page PeopleCode on the Activate event of PTFP_DATATYPE.



```
import PTFP_FEED:FeedFactory;
import PTFP_FEED:UTILITY:HoverMenu;
import PTFP_FEED:UTILITY:RelatedFeedsRequest;


Component string &PTFP_FEED_DATA_TYPE_FEED;
Component PTFP_FEED:FeedFactory &PTFP_FEED_FACTORY;

Declare Function LoadFeedPropertiesFromData PeopleCode PTFP_CPROP_WRK.FUNCLIB FieldFormula;

Local array of string &thisDSS;

Local PTFP_FEED:UTILITY:RelatedFeedsRequest &request;
Local array of PTFP_FEED:UTILITY:RelatedFeedsRequest &requests = CreateArrayRept(&request, 0);
Local PTFP_FEED:UTILITY:HoverMenu &resultMenu;


If (PTFP_DATATYPE.ACTIVE_FLAG.Value <> "A") Or
      (DERIVED_PTFP.PTFP_PUB_AS_PB.Enabled = False) Then

   /* Hide the controls when in Add mode or is inactive or the "FEED" data type does not exist */
   DERIVED_PTFP.PTFP_PUB_AS_PB.Visible = False;
   DERIVED_PTFP.PTFP_HTMLAREA.Visible = False;
```

Adding the Related Feeds Hover Menu to Pagelet Wizard Pagelets

When adding the hover menu, consider these points:

- Only home page pagelets and pagelets, which can be embedded, on application pages support the related feeds hover menu.
- Transformer output must be XML or XHTML.
- Multi-group feed menu is supported.

Example: Related Feeds Hover Menu XSL

This is sample XSL:

```
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:variable name="NumRows">
  <xsl:value-of select="count(/queryresult/queryrows/row)" />
</xsl:variable>

<xsl:template match="/">
  <table border="0" cellpadding="1" cellspacing="1" width="100%" summary="">
    <xsl:if test="$NumRows=0">
      <tr><td class="PSTEXT">no data available</td></tr>
    </xsl:if>

    <xsl:if test="$NumRows>0">
      <tr><td>
        <PSRELATEDFEEDSLINK>
          <feed id="ADMN_LIST_OF_FEEDS" />
          <feeds>
            <label>PSQUERY Feed List Feeds</label>
            <description>List of all PSQUERY feed list feeds</description>
            <feedDataType id="FEED" />
            <dataSourceSetting id="PTFP_DATATYPE_ID" value="PSQUERY" />
          </feeds>
        </PSRELATEDFEEDSLINK>
      </td></tr>
    </xsl:if>
  </table>
</xsl:template>

</xsl:stylesheet>
```

Examples of Specific Feed Types

This section provides examples of specific steps required when developing different feed types.

Up-front Scheduled Feeds

Creating up-front scheduled feeds requires additional steps. You must complete the following tasks when creating up-front scheduled feeds:

1. Ensure that the service operation you use to publish the up-front feed messages to the Integration Broker queues satisfies these conditions:

    - It must be an asynchronous, one-way service operation.
    - It must have PT_FEED_REQUEST.VERSION_1 as the message.
    - It must be secured appropriately. This security is enforced by the scheduled feed GetFeed service operation handler at run time.
    - The queue used in the service operation must have the Archive option enabled if the up-front feeds are to be archived. If this option is not enabled, then the Archival Feeds will delete the feed messages in the Integration Broker queue.

2. List the service operation used to publish the up-front feed messages in the Define Feed Data Types page. The system uses this information to archive feeds.

3. Set the feed format and language as message attributes before publishing the message:

    ```
     /* Set the feed format of the message (i.e. Atom 1.0). */
    &succeeded = &responseMsg.IBInfo.AddAttribute(&feedFactory.Utility.QUERYPARAME⇒
    TER_FEEDFORMAT, &feedDoc.FeedFormat);

    /* Set the language of the message. */
    &succeeded = &responseMsg.IBInfo.AddAttribute(&feedFactory.Utility.QUERYPARAME⇒
    TER_LANGUAGE, %Language);
    ```

4. Define the DSPARAMETER_MAXROW, DSPARAMETER_SF_PAGING, DSPARAMETER_INCREMENTAL, and DSPARAMETER_SF_MAXMINUTES data source parameters found in the PTFP_FEED:UTILITY:Utility application class and set them to appropriate values in your implementation of the processSettingsChange method for your data source. The system requires DSPARAMETER_MAXROW and DSPARAMETER_SF_MAXMINUTES

for archiving feeds and it uses DSPARAMETER_SF_PAGING for paged feeds and DSPARAMETER_INCREMENTAL for incremental feeds.

For example:

```
&thisDSP = %This.addParameter(&utility.DSPARAMETER_MAXROW, String(&utility.SF_⇒
MAXROWOPTION_LATESTMSG));
&thisDSP.Name = &thisDSP.ID;
&thisDSP.Description = MsgGetText(219, 3005, "Message Not Found - Max Entries"⇒
);
&thisDSP.FieldType = &utility.FIELDTYPE_NUMBER;
&thisDSP.DefaultValue = String(&utility.SF_MAXROWOPTION_LATESTMSG);
&thisDSP.Value = &thisDSP.DefaultValue;
&thisDSP.Required = True;

&thisDSP = %This.addParameter(&utility.DSPARAMETER_SF_PAGING, String(&utility.⇒
SF_PAGINGOPTION_NOPAGING));
&thisDSP.Name = &thisDSP.ID;
&thisDSP.Description = MsgGetText(219, 3006, "Message Not Found - Paging");
&thisDSP.FieldType = &utility.FIELDTYPE_SIGNEDNUMBER;
&thisDSP.DefaultValue = String(&utility.SF_PAGINGOPTION_NOPAGING);
&thisDSP.Value = &thisDSP.DefaultValue;
&thisDSP.Required = True;

&thisDSP = %This.addParameter(&utility.DSPARAMETER_INCREMENTAL, String(&utilit⇒
y.INCREMENTALOPTION_NO));
&thisDSP.Name = &thisDSP.ID;
&thisDSP.Description = MsgGetText(219, 3008, "Message Not Found - Incremental"⇒
);
&thisDSP.FieldType = &utility.FIELDTYPE_SIGNEDNUMBER;
&thisDSP.DefaultValue = String(&utility.INCREMENTALOPTION_NO);
&thisDSP.Value = &thisDSP.DefaultValue;
&thisDSP.Required = True;

&thisDSP = %This.addParameter(&utility.DSPARAMETER_SF_MAXMINUTES, String(&util⇒
ity.SF_MAXMINUTES_ALLMSGS));
&thisDSP.Name = &thisDSP.ID;
&thisDSP.Description = MsgGetText(219, 3007, "Message Not Found - Max Min");
&thisDSP.FieldType = &utility.FIELDTYPE_SIGNEDNUMBER;
&thisDSP.DefaultValue = String(&utility.SF_MAXMINUTES_ALLMSGS);
&thisDSP.Value = &thisDSP.DefaultValue;
&thisDSP.Required = True;
```

Important! The incremental feed option is incompatible with the paged feed option. Do not allow both options to be set simultaneously.

5. Modify the associated advanced feed options page to allow feed administrators the ability to set these options.

Related Links

Archiving Feeds

Real-Time Incremental Feeds

Creating real-time incremental feeds also requires additional steps. You must complete the following tasks when creating real-time incremental feeds:

1. Define the DSPARAMETER_INCREMENTAL data source parameter found in the PTFP_FEED:UTILITY:Utility application class and set it to an appropriate value in your implementation of the processSettingsChange method for your data source :

   ```
   &thisDSP = %This.addParameter(&utility.DSPARAMETER_INCREMENTAL, String(&utilit⇒
   y.INCREMENTALOPTION_NO));
   &thisDSP.Name = &thisDSP.ID;
   &thisDSP.Description = MsgGetText(219, 3008, "Message Not Found - Incremental"⇒
   );
   &thisDSP.FieldType = &utility.FIELDTYPE_SIGNEDNUMBER;
   &thisDSP.DefaultValue = String(&utility.INCREMENTALOPTION_NO);
   &thisDSP.Value = &thisDSP.DefaultValue;
   &thisDSP.Required = True;
   ```

   Important! The incremental feed option is incompatible with the paged feed option. Do not allow both options to be set simultaneously.

2. Modify the associated advanced feed options page to allow feed administrators the ability to set this option.

3. Generate delta feed entries in your implementation of the execute method for your data source based on the QUERYPARAMETER_IFMODIFIEDSINCE query parameter of the PTFP_FEED:UTILITY:Utility application class.

   Your implementation of the execute method must contain both the QUERYPARAMETER_IFNONEMATCH and the QUERYPARAMETER_IFMODIFIEDSINCE query parameters. QUERYPARAMETER_IFNONEMATCH is the feed ID and QUERYPARAMETER_IFMODIFIEDSINCE is the time at which the feed was last requested.

   The following code excerpt shows how to get the QUERYPARAMETER_IFNONEMATCH and QUERYPARAMETER_IFMODIFIEDSINCE query parameters using RequestInfo in the execute method of the data source:

   ```
   Local PTFP_FEED:DataSource:DataSourceParameter &thisDSP;
   Local string &ifNoneMatch, &ifModifiedSince, &select;
   Local datetime &lastmodified_dt = DateTime6(1900, 1, 1, 0, 0, 0);
   Local boolean &incremental;

   /* Get the Incremental Parameter */
   &thisDSP = %This.getParameterById(&utility.DSPARAMETER_INCREMENTAL);
   If &thisDSP <> Null And
   (&thisDSP.EvaluatedValue = String(&utility.INCREMENTALOPTION_YES)) Then
      &incremental = True;
   Else
      &incremental = False;
   End-If;

   &ifNoneMatch = &utility.RequestInfo.getParameter(&utility.QUERYPARAMETER_IFNON⇒
   EMATCH);
   &ifModifiedSince = &utility.RequestInfo.getParameter(&utility.QUERYPARAMETER_I⇒
   FMODIFIEDSINCE);
   If All(&ifModifiedSince) Then
      &lastmodified_dt = &utility.httpStringToDatetime(&ifModifiedSince);
   End-If;

   /* Compare and verify that &ifNoneMatch is same as the feed ID            */

   /* Compare the &lastmodified_dt with appropriate datetime column like the */
   /* LASTUPDDTTM field in the record used for generating the feed entries   */
   ```

4. When the execute method of a data source returns no feed entries, the Feed Publishing Framework issues a 304-Not Modified HTTP header. If you are using a custom feed handler—that is,

a service operation different from the PTFP_GETFEED service operation —then use the setMessageHeadersAndMimeType method to set HTTP conditional headers.

For example:

```
method OnRequest
   /+ &pRequestMsg as Message +/
   /+ Returns Message +/
   /+ Extends/implements PS_PT:Integration:IRequestHandler.OnRequest +/

   Local Message &responseMsg;
   Local XmlDoc &xmlDoc;
   Local string &temp, &errorText;

   Local PTFP_FEED:UTILITY:Utility &utility = &feedFactory_inst.Utility;
   Local PTFP_FEED:XML_FEED:FeedDoc &feedDoc;
   Local PTFP_FEED:UTILITY:FeedRequest &request;

   /* Ccreate the Search Request object */
   &request = create PTFP_FEED:UTILITY:FeedRequest("FeedRequest");

   ...

   try

      &feedDoc = &feedFactory_inst.getFeedDoc(&request);

   catch PTFP_FEED:EXCEPTION:NotFoundException &ex1
      &errorText = MsgGetExplainText(219, 3112, "(Message not found) Not Found⇒
");

   catch PTFP_FEED:EXCEPTION:PrivilegeException &ex2
      &errorText = MsgGetExplainText(219, 3113, "(Message not found) Not Autho⇒
rized");

   catch PTFP_FEED:EXCEPTION:FeedException &ex3
      &errorText = &utility.getExceptionText(&ex3);

   end-try;

   /* Create the response message */
   &responseMsg = CreateMessage(Operation.PTFP_GETFEED, %IntBroker_Response);

   If None(&errorText) Then
      &responseMsg = &utility.setMessageHeadersAndMimeType(&responseMsg, &feed⇒
Doc, &request);
   Else
      &temp = "<?xml version='1.0' encoding='UTF-8'?><ErrorMessage>" | &errorT⇒
ext | "</ErrorMessage>";
      &xmlDoc = CreateXmlDoc(&temp);
      &responseMsg.SetXmlDoc(&xmlDoc);
      &responseMsg.SegmentContentType = &utility.MIMETYPE_XML;
   End-If;

   Return &responseMsg;

end-method;
```

Paged Feeds

Creating paged feeds also requires additional steps.

Note: Paged feeds are supported for scheduled feeds only. The framework supports paged feeds via Integration Broker message segments. %MaxMessageSize is recommended when creating Integration Broker message segments for paged feeds.

You must complete the following tasks when creating paged feeds:

1. Define the DSPARAMETER_SF_PAGING data source parameter found in the PTFP_FEED:UTILITY:Utility application class and set it to an appropriate value in your implementation of the processSettingsChange method for your data source.

   For example:

   ```
   /* PAGING parameter */
   &thisDSP = %This.addParameter(&utility.DSPARAMETER_SF_PAGING, String(&utility.⇒
   SF_PAGINGOPTION_NOPAGING));
   &thisDSP.Name = &thisDSP.ID;
   &thisDSP.Description = MsgGetText(219, 3007, "Message Not Found - Page Size");
   &thisDSP.FieldType = &utility.FIELDTYPE_SIGNEDNUMBER;
   &thisDSP.DefaultValue = String(&utility.SF_PAGINGOPTION_NOPAGING);
   &thisDSP.Value = &thisDSP.DefaultValue;
   &thisDSP.Required = True;
   ```

   Important! The paged feed option is incompatible with the incremental feed option. Do not allow both options to be set simultaneously.

2. Modify the associated advanced feed options page to allow feed administrators the ability to set this option.

Appendix A

Troubleshooting Tips

Common Problems

This table lists some common errors that may occur when setting up the feed framework:

Problem	***Resolution***

When you click a Publish as Feed link, the system displays the following pop-up error message:
 	
`Some required configuration steps are no⇒`
`t done for the Feed Publishing Framework⇒`
`. Please check the IB Service Configurat⇒`
`ion.`
	Complete configuration of your PeopleSoft system to support feeds.
	
	See Configuring Integration Broker Service Target Locations.

When you click the link for a feed, the system displays the following error message:
 	
`<?xml version="1.0" ?>`
`- <IBResponse type="error">`
`  <DefaultTitle>Integration Broker Respo⇒`
`nse</DefaultTitle>`
`  <StatusCode>20</StatusCode>`
`  <MessageSet>158</MessageSet>`
`  <MessageID>10733</MessageID>`
`  <DefaultMessage>Integration Gateway fa⇒`
`iled  while processing the message.</Def⇒`
`aultMessage>`
`- <MessageParameters>`
`  <Parameter>password</Parameter>`
`  </MessageParameters>`
`  </IBResponse>`
	Complete configuration of your PeopleSoft system to support feeds.
	
	See Configuring Integration Broker Service Target Locations.

When you click the link for a feed—for example, on the My Feeds page—a new browser window opens and redisplays the page that you are on instead of the feed.	The feed was published even though configuration of the PeopleSoft system was incomplete. Complete configuration of your PeopleSoft system to support feeds. See Configuring Integration Broker Service Target Locations.

During configuration of Integration Broker, you get the following error message when you attempt to ping the default local node:
 	
`Integration Broker Authentication: No no⇒`
`de Authentication Option set for node no⇒`
`de_name on Target sys`
	Set the node authentication to either password or certificate.
	
	See Configuring the Default Local Node.

Problem	***Resolution***
When you click the link for a feed, the system displays one of the following error messages: - `Address not found` - `Could Not Connect to Server` - `Unable to connect` - `Cannot display the webpage`	The Target Location or the Secure Target Location is not set *properly* on the Service Configuration page. See Configuring Integration Broker Service Target Locations.

When you click a feed entry in a feed, the system displays the following error message:
`Error 404--Not Found`
`From RFC 2068 Hypertext Transfer Protoco⇒`
`l -- HTTP/1.1:`
`10.4.5 404 Not Found`
`The server has not found anything matchi⇒`
`ng the Request-URI. No indication is giv⇒`
`en of whether the condition is temporary⇒`
` or permanent.`
`If the server does not wish to make this⇒`
` information available to the client, th⇒`
`e status code 403 (Forbidden) can be use⇒`
`d instead. The 410 (Gone) status code SH⇒`
`OULD be used if the server knows, throug⇒`
`h some internally configurable mechanism⇒`
`, that an ol resource is permanently una⇒`
`vailable and has no forwarding address.`
	The Portal Context URI text or Portal URI text for the local host node has not been configured on the Node Definitions - Portal page.
	See Setting URI Text for Local Host Nodes.

When you open a worklist feed, the system displays feed entries but not any hyperlinks for viewing the details for each feed entry.	The Portal Context URI text or Portal URI text for the local host node for Workflow has not been configured on the Node Definitions - Portal page. See Setting URI Text for Local Host Nodes. See Understanding Worklist Feeds.

When you click a feed entry in a feed, the system displays the following error message:
`Site name is not valid. Check your synta⇒`
`x and try again.`
	The Portal Context URI text or Portal URI text for the local host node is not configured *properly* on the Node Definitions - Portal page.
	Note: Both values require a terminating / to be correct.
	See Setting URI Text for Local Host Nodes.

The feed document displays the feed title but no feed entries.	Feed entries appear in the context of the logged in user. One of the following conditions could result in no feed entries in the feed document: - The current user has no feed entries, for example, the user does not have any unused items in the specified worklist. - The feed definition was not specified correctly.

Problem	***Resolution***
When you click the link for a feed, a new browser window opens, but you are prompted to authenticate yourself with a user name and password: `A username and password are being reques⇒ ted by` *`server_name`*	A mismatch occurred in the way the system was identified in the login URL versus how the system was identified for the Integration Broker service configuration. For example, this problem can occur when the login URL is specified as a numeric IP address (that is, 192.0.2.1) and the service configuration is specified by a fully qualified domain name (that is, myserver.myco.com), or vice versa.
When you click the link for a feed, a new browser window opens, but you are prompted to authenticate yourself to the Oracle WebLogic Server with the following message: `The server` *`server_name`* `at WebLogic Serve⇒ r requires a username and password.`	You must configure the Oracle WebLogic Server to disable its own authentication. See Configuring Oracle WebLogic Server to Disable Authentication.
When you request a feed, you are challenged for authentication even though you are already signed into the PeopleSoft system.	The PeopleSoft login URL and the feed URL are of different protocols, for example, one uses HTTP and the other uses HTTPS. If a secure target location was specified on the Integration Broker Service Configuration page, then the PeopleSoft login URL should be HTTPS. Conversely, if no secure target location was specified on the Service Configuration page and only a target location was specified, then the PeopleSoft login URL should be HTTP. In either case, when a mismatch of protocols is used on your PeopleSoft system, you will be prompted for credentials again when you attempt to view a feed from a PeopleSoft page. **Note:** Certain integrations indicate the use of HTTPS. PeopleSoft feeds support integration with third-party feed readers through basic authentication only. However, since basic authentication is not secure, you should use HTTPS for feeds in this scenario. Set it by specifying a secure target location on the Integration Broker Service Configuration page. See Configuring Integration Broker Service Target Locations.

<table>
<tr><th><em>Problem</em></th><th><em>Resolution</em></th></tr>
<tr><td>

When you click the link for a feed, the system displays one of the following error messages:

IBResponse type="error"

```
<?xml version="1.0" ?>
- <IBResponse type="error">
  <DefaultTitle>Integration Broker Respo⇒
nse</DefaultTitle>
  <StatusCode>20</StatusCode>
  <MessageSet>158</MessageSet>
  <MessageID>536</MessageID>
  <DefaultMessage>UserID QE_GUEST not au⇒
thorized for Service Operation PTFP_GETP⇒
REPUBFEED</DefaultMessage>
- <MessageParameters>
  <Parameter>QE_GUEST</Parameter>
  <Parameter>PTFP_GETPREPUBFEED</Paramet⇒
er>
  </MessageParameters>
  </IBResponse>
```

You are not authorized

```
<?xml version="1.0" ?>
  <ErrorMessage>You are not authorized t⇒
o view this feed.</ErrorMessage>
```

</td><td>

The user ID assigned to the ANONYMOUS node does not have sufficient privileges. At a minimum, this user ID must be assigned the PTPT1000 (PeopleSoft User) permissions.

See Configuring the Default User of the ANONYMOUS Node.

</td></tr>
</table>

Appendix B

Disabling Authentication on Oracle WebLogic Server

Configuring Oracle WebLogic Server to Disable Authentication

The Feed Publishing Framework requires that the Oracle WebLogic Server's own authentication be disabled. Disabling Oracle WebLogic Server's authentication allows authentication to be passed through and handled by the PeopleSoft servlet. The config.xml file is shipped with this authentication disabled and, therefore, no action is required unless you have changed the setting in this file.

For example, if Oracle WebLogic Server's authentication has been re-enabled, when you click the link for a feed, a new browser window opens, but you are prompted to authenticate yourself to the Oracle WebLogic Server with the following message:

```
The server server_name at WebLogic Server requires a username and
password.
```

By default, the delivered config.xml file is set to disable the Oracle WebLogic Server's own authentication. No additional configuration is required unless you have changed this authentication setting.

To configure Oracle WebLogic Server to disable authentication:

1. Stop the web server.

2. Go to the *PS_HOME*\webserv\web_server\config folder.

3. Edit the config.xml file by adding the following tag before the closing `</security-configuration>` tag:

   ```
   <enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-credenti⇒
   als>
   ```

   The edited file looks similar to the following with the added line in bold near the end:

   ```
     <security-configuration xmlns:xacml="http://www.bea.com/ns/weblogic/90/secur⇒
   ity/xacml">
       <name>peoplesoft1</name>
       <realm>
         <sec:authentication-provider xsi:type="wls:default-authenticatorType"/>
         <sec:authentication-provider xsi:type="wls:default-identity-asserterType⇒
   ">
           <sec:active-type>AuthenticatedUser</sec:active-type>
         </sec:authentication-provider>
         <sec:role-mapper xsi:type="xacml:xacml-role-mapperType"/>
         <sec:authorizer xsi:type="xacml:xacml-authorizerType"/>
         <sec:adjudicator xsi:type="wls:default-adjudicatorType"/>
         <sec:credential-mapper xsi:type="wls:default-credential-mapperType"/>
         <sec:cert-path-provider xsi:type="wls:web-logic-cert-path-providerType"/⇒
   >
   ```

```
      <sec:cert-path-builder>WebLogicCertPathProvider</sec:cert-path-builder>
      <sec:name>myrealm</sec:name>
    </realm>
    <default-realm>myrealm</default-realm>
    <credential-encrypted>{3DES}XLLC9Wru5qKeMAlvEULruO9LodVs7o3du4WVMtMs/ffmyP⇒
16aD4NKEv0va5IxytcWvGRV50mB5dYbzhos9XWNN0Lz4mQoXiy</credential-encrypted>
    <node-manager-username>system</node-manager-username>
    <node-manager-password-encrypted>{3DES}c/WblCLbZubUdNamvjN1sw==</node-mana⇒
ger-password-encrypted>
    <enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-cred⇒
entials>
</security-configuration>
```

4. Save the file.

5. Restart the web server.

Appendix C

Feed Metadata Table Structures in PeopleTools 8.52 and Later Releases

Feed Metadata Tables

Note: As of PeopleTools 8.52, feed metadata is stored in tables with names different from previous releases. The UPGPT852PTFP Application Engine program of the PeopleTools 8.52 upgrade process automatically converts pre-8.52 metadata to the 8.52 table structure. Because the old tables are used during the upgrade process and during ad hoc conversions, these tables remain in the PeopleTools 8.52 and later databases.

The following table lists the metadata table names prior to PeopleTools 8.52, new table names in 8.52, and any columns added in 8.52:

Pre-8.52 Table Name	***8.52 Table Name***	***New Columns in 8.52***
PS_PTFP_CATEGORY	PSFP_CATEGORY	VERSION OBJECTOWNERID
PS_PTFP_CATG_LANG	PSFP_CATG_LANG,	
PS_PTFP_FEED	PSFP_FEED	VERSION PTFP_FEED_TEMPLATE
PS_PTFP_FEED_LANG	PSFP_FEED_LANG	
PS_PTFP_SETTINGS	PSFP_SETTINGS	
PS_PTFP_ADMN_PREF	PSFP_ADMN_PREF	
PS_PTFP_USER_PREF	PSFP_USER_PREF	
PS_PTFP_SECURITY	PSFP_SECURITY	
PS_PTFP_PARMS	PSFP_PARMS	
PS_PTFP_PARMS_LANG	PSFP_PARMS_LANG	
PS_PTFP_PVALS	PSFP_PVALS	
PS_PTFP_PVALS_LANG	PSFP_PVALS_LANG	
PS_PTFP_ATTRS	PSFP_ATTRS	

Pre-8.52 Table Name	***8.52 Table Name***	***New Columns in 8.52***
PS_PTFP_ATTRS_LANG	PSFP_ATTRS_LANG	
PS_PTFP_DATATYPE	PSFP_DATATYPE	VERSION OBJECTOWNERID
PS_PTFP_DTYPE_LANG	PSFP_DTYPE_LANG	
PS_PTFP_DTYPE_ATTR	PSFP_DTYPE_ATTR	
PS_PTFP_DATTR_LANG	PSFP_DATTR_LANG	

Note: Any direct reference to the pre-8.52 tables in any existing custom implementations must be modified to use the newer tables.