HITACHI
Inspire the Next

Embedding Pentaho Business Analytics (BA) Content

HITACHI
Inspire the Next

Change log (if you want to use it):

Date	Version	Author	Changes

Contents

This page intentionally left blank.

Overview

This document covers some best practices on embedding Pentaho BA content into a third-party web application. Some of the things discussed here include integrating with web applications, security, and brief descriptions and examples of embedding individual Pentaho components.

The intention of this document is to speak about topics generally; however, these are the specific versions covered here:

Software	Version(s)
Pentaho	6.x, 7.x, 8.0

The Components Reference in Pentaho Documentation has a complete list of supported software and hardware.

© Hitachi Vantara Corporation 2018. All Rights Reserved

Best Practices for Embedding Pentaho Content

The most common way to embed content into another application is through iFrame integration, or with plugins such as CTools. Div integration also works, but may be harder to configure.

Pentaho integrates with third-party web applications using Representational State Transfer (REST) services that allow you to interact with different Pentaho components. Think about your particular use cases while you read through this document, as some sections may not be applicable.

You can find details on these topics in the following sections:

Integrating Pentaho with Third-Party Web Applications

Each Pentaho component establishes multiple connections and communication channels with the third-party web application, while REST executes or loads different scripts.

Because cross-site referencing could be a concern, we recommend that both Pentaho and the third-party application live within the same domain and proxy. If your third-party application is Java-based, we recommend deploying the Pentaho Server into the same application server. This approach makes for an easier embed configuration process.

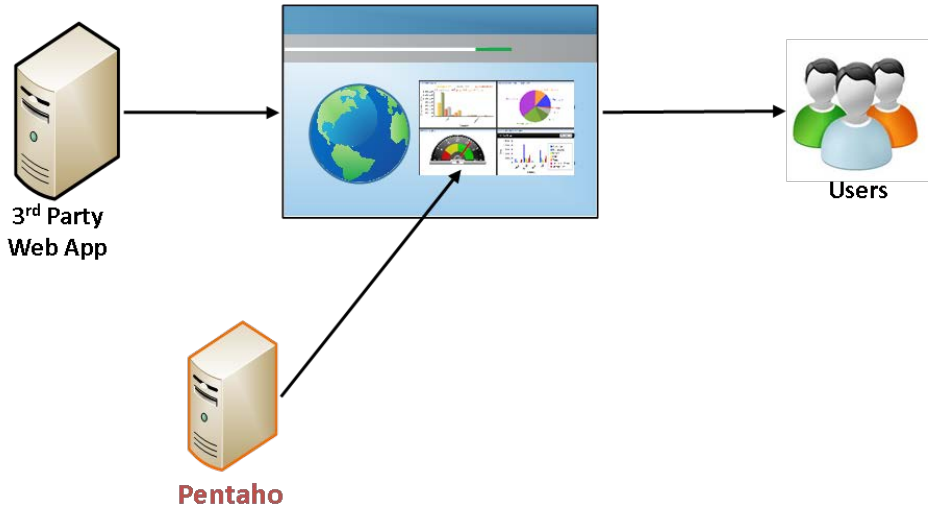

Figure 1: Embed Environment Diagram

In production environments, the Pentaho Server runs on a specific IP address, other than `127.0.0.1`, and you must modify the trusted proxy to match that address or `hostname` so that Pentaho plugins run as expected and can accept REST API calls. Configure the Proxy Trusting Filter in Pentaho documentation has instructions on how to do this.

© Hitachi Vantara Corporation 2018. All Rights Reserved

All Pentaho BA REST calls use a generic format:

```
http://<server>:<port>/pentaho/api/repos/:<solution>:<path>:<filename>/<action>
```

Security

One of the most important things to consider when embedding is how to handle security. Pentaho relies on the Spring Security pluggable authentication framework. Embeddable BA supports all security configurations based on the Spring framework. Implement Advanced Security in the Pentaho documentation has details for security configurations.

By default, the Pentaho Server uses a JDBC-based data access object that is tied to a Jackrabbit database. Users and roles are configured through the Administration perspective in the Pentaho User Console (PUC), and content authorization is controlled by the server administrator.

Authentication Before Rest API Calls

Before REST API calls will work, the requested action needs to be authenticated with the server.

We recommend Basic Authentication only for development or testing cases.

© Hitachi Vantara Corporation 2018. All Rights Reserved

Pentaho BA Components

This section describes the major services for integrating various kinds of content. Before you get started, we recommend creating a use case to identify needed services in your organization, such as rendering reports that were previously created, creating your own views, and so forth.

Previous experience with programming and API use is required to use Pentaho APIs.

Pentaho Analyzer

These APIs allow for more fine-grained interaction with the Analyzer viewer, reports, and data. API usage is divided into two areas:

- Render API analyzer content (Basic HTML iFrame population)
- JavaScript to interact on Operation, Events, and Report data with displayed view. The Pentaho API documentation has many different options available.

Here are some of the more common Analyzer-specific API calls:

- Displaying a report in viewer mode:

```
/pentaho/api/repos/:path:to:saved:analyzer:report.xanalyzer/viewer
```

  For example:

```
http://localhost:8080/pentaho/api/repos/:public:Steel%20Wheels:Leading%20Product%20Lines%20(pivot%20table).xanalyzer/viewer
```

- Displaying a report in editor mode:

```
/pentaho/api/repos/:path:to:saved:analyzer:report.xanalyzer/editor
```

  For example:

```
http://localhost:8080/pentaho/api/repos/:public:Steel%20Wheels:Leading%20Product%20Lines%20(pivot%20table).xanalyzer/editor?showRepositoryButtons=true
```

- Creating a new report:

```
/pentaho/api/repos/xanalyzer/editor?catalog=YourCatalog&cube=YourCube
```

  For example:

```
http://localhost:8080/pentaho/api/repos/xanalyzer/editor?catalog=<Schema>&cube=<CubeName>
```


© Hitachi Vantara Corporation 2018. All Rights Reserved

Pentaho Active Reports

Two options are available for interactive reporting services: render an existing. `prpti` file, or create a new one. More options are available in Pentaho documentation, but some of the more common API calls can be found below:

- Displaying an Interactive Report in view mode:

```
/pentaho/api/repos/:path:to:saved:pir:report.prpti/prpti.view
```

  For example:

```
http://localhost:8080/pentaho/api/repos/:public:Steel%20Wheels:Vendor%20Sales%20Report%20(interactive%20report).prpti/prpti.view
```

- Displaying an Interactive Report in edit mode:

```
/pentaho/api/repos/:path:to:saved:pir:report.prpti/prpti.edit
```

  For example:

```
http://localhost:8080/pentaho/api/repos/:public:Steel%20Wheels:Vendor%20Sales%20Report%20(interactive%20report).prpti/prpti.edit
```

- Exporting an Interactive Report:

```
/pentaho/api/repos/:path:to:saved:pir:report.prpti/report
```

  For example, to select the output type for export, pass the parameter output-target with one of the parameters shown in the table:

```
http://localhost:8080/pentaho/api/repos/:public:Steel%20Wheels:Vendor%20Sales%20Report%20(interactive%20report).prpti/report?output-target=table/csv;page-mode=stream
```

Table 1: Exporting an Interactive Report – Output Types and Parameters

Output Type	Parameter
HTML (paginated)	`table/html;page-mode=page`
HTML (single page)	`table/html;page-mode=stream`
PDF	`pageable/pdf`
Excel	`table/excel;page-mode=flow`
Excel 2007	`application/vnd.openxmlformats-officedocument.spreadsheetml.sheet;page-mode=flow`
CSV	`table/csv;page-mode=stream`
Rich Text Format	`table/rtf;page-mode=flow`
Text	`pageable/text`

© Hitachi Vantara Corporation 2018. All Rights Reserved

- Creating a new Interactive Report:

```
/pentaho/api/repos/pentaho-interactive-reporting/prpti.new
```

Report Viewer

Unlike Interactive Reporting, the report viewer gives you only one render option - view existing report - but with two different perspectives:

- Display a report passing parameters either as `GET` or `POST`:

```
/pentaho/api/repos/:path:to:saved:report.prpt/generatedContent
```

- Display a report with parameter panel:

```
/pentaho/api/repos/:path:to:saved:report.prpt/viewer
```

More information and some common samples can be found in Reporting Samples in Pentaho documentation.

Enterprise Dashboard

Enterprise dashboards can be integrated in view-only mode using the following example URL:

```
http://server:8080/pentaho/api/repos/:public:Steel%20Wheels:Regional%20Sales%20(dashboard).xdash/viewer
```

The results of this example URL would resemble the following:

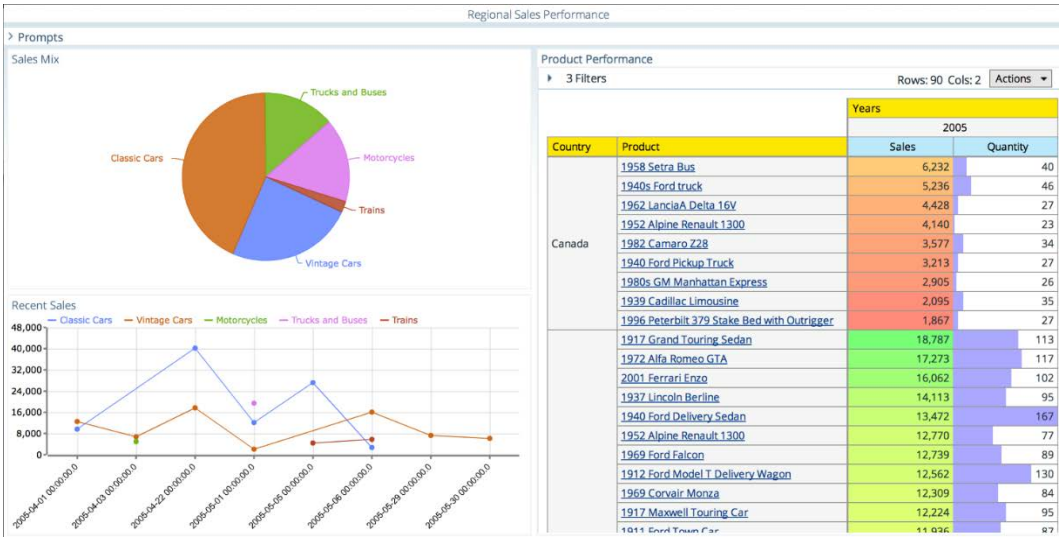

Figure 2: Example Dashboard

© Hitachi Vantara Corporation 2018. All Rights Reserved

Data Integration

REST APIs for the Pentaho Data Integration (DI) Server offer several methods to interact with repository scheduling and Carte servers:

- **Scheduling** – Create, read, update, delete, and list schedules on the DI server.
- **Carte** – Interact with Carte to manage DI servers and execute ETL processes (Jobs and Transformations).
- **Server** – Allocate sockets and ports, get a list of slave servers, and then register them.
- **Jobs** – Create, run, and delete jobs. Get images, statuses, or start and stop jobs.
- **Transformations** – Create, update, and delete transformations. Get images, statuses, or start, pause, stop transformations. Provides web service calls to create, update, and delete transformations.

A complete list of all available REST APIs for DI can be found in the Pentaho Developer Center under API Documentation for PDI Platform.

CTools

Here are some common API calls that you might want to use if you have custom dashboards that use CTools:

- Dashboards and Widgets:

```
/pentaho/api/repos/:path:to:saved:dashboard.wcdf/generatedContent?ts=undefined
```

  The `ts=undefined` parameter could be timestamped to prevent caching. This can be either removed or replaced with a random number or value. For example:

```
http://localhost:8080/pentaho/api/repos/:public:plugin-samples:pentaho-cdf-dd:cde_sample1.wcdf/generatedContent?ts=1405266243858
```

- Execute a CDA query:

```
/pentaho/plugin/cda/api/doQuery?path=/path/to/cda/datasource.cda&dataAccessId=ID
```

  You can also pass parameters for this query. For example:

```
http://localhost:8080/pentaho/plugin/cda/api/doQuery?path=/public/plugin-samples/cda/cdafiles/mondrian-jndi.cda&dataAccessId=1&paramstatus=Shipped
```

- Export a CDA query to Excel:

```
/pentaho/plugin/cda/api/doQuery?path=/path/to/cda/datasource.cda&dataAccessId=ID&outputType=xls.
```


© Hitachi Vantara Corporation 2018. All Rights Reserved

For example:

```
http://localhost:8080/pentaho/plugin/cda/api/doQuery?path=/public/plugin-samples/cda/cdafiles/mondrian-jndi.cda&dataAccessId=1&paramstatus=Shipped&outputType=xls
```

- List the available CDA queries:

```
/pentaho/plugin/cda/api/listQueries?path=/path/to/cda/datasource.cda
```

For example:

```
http://localhost:8080/pentaho/plugin/cda/api/listQueries?path=/public/plugin-samples/cda/cdafiles/mondrian-jndi.cda
```

- Refresh CDA Cache (full) – note that `method=removeAll` needs to be send as `POST`:

```
/pentaho/plugin/cda/api/cacheMonitor/removeAll
```

Server Administration

Besides being able to display content and manage ETL processes execution, REST-based API allows you to create custom applications which interact with Pentaho Data. Here are the categories of REST APIs:

- **Managing Data Sources** – List, download, upload, and remove data sources in the BA platform.
- **File Management** – List, publish, and manage files and folders on the server. This is recommended for your custom applications used to manage and administrate solutions outside of the Pentaho User Console (PUC).
- **Scheduling** – Create, read, update, delete, and list schedules across any Pentaho server.

A complete list of all available REST APIs for BA can be found in the Pentaho Developer Center under API Documentation for BA Platform.

Theming

The Pentaho User Console (PUC) is the standard web application for the Pentaho Server, and includes interactive elements for Analysis, Reporting, and Dashboards. The Administration page of the console is the central framework through which your server is configured and managed. You can alter the look and feel of PUC by editing its configuration files, graphics, and CSS style sheets manually. Normally, the individual CSS for every component must also be modified (Analyzer, Dashboard Designer and Interactive Reporting), when you want to embed Pentaho with your own style.

© Hitachi Vantara Corporation 2018. All Rights Reserved

These directories contain style sheets and theme material for each.

Table 2: Style Sheets and Theme Materials for Pentaho Products

Product Name	Location	Description
Analyzer	`/pentaho-solutions/system/analyzer/styles/themes/`	Style sheets and theming materials for Analyzer
Interactive Reporting	`/pentaho-solutions/system/pentaho-interactive-reporting/resources/web/themes/`	Style sheets and theming materials for Interactive Reporting
Dashboard Designer	`/pentaho-solutions/system/dashboards/themes/`	Style sheets and theming materials for Dashboard Designer
Mobile	`/pentaho-solutions/system/pentaho-mobile-plugin/resources/css`	Style sheets and theming materials for mobile

Pentaho User Console Styling has complete instructions on how to theme your console.

A Word About Multi-Tenancy

You may have business requirements in which individuals or groups share the same instance of a software application but have separate data and content. These individuals or groups are referred to as tenants. In a multi-tenancy architecture, customers share infrastructures, applications, or databases to gain performance advantages, while reducing overhead. A provider defines the rules for the tenants within the system. Each tenant can be restricted to see only their own secure data and content while using the same software. Note that multi-tenancy differs from multi-instance architecture, which is based on maintaining separate copies of the software to serve separate clients.

Pentaho Enterprise software is designed to work as a stand-alone, multi-tenant solution or to be embedded as part of a multi-tenant service. Pentaho is flexible enough to support a variety of multi-tenancy approaches. Taking advantage of Pentaho's multi-tenancy capabilities can provide sophisticated analytics while reducing complexity and cost.

*If this applies to your use case, we recommend reading through the Multi-Tenancy documentation **BEFORE** embedding Pentaho into your application. Security concerns and data management must work together between multi-tenancy and embedding.*

© Hitachi Vantara Corporation 2018. All Rights Reserved

Related Information

Here are some links to information that you may find helpful while using this best practices document:

- Customize PUC
- API Documentation for OEMs
- Embed BA into Web Applications
- Pentaho and Multi-Tenancy
- Configuring the Proxy Trusting Filter
- Support for all Security Configurations
- Implement Advanced Security
- Authenticate with the Server
- Basic HTML iFrame Population
- API Documentation
- Interactive Reporting Samples
- Reporting Samples
- API Documentation for PDI Platform
- Rest API Reference
- Pentaho Components Reference

© Hitachi Vantara Corporation 2018. All Rights Reserved