

CPK^{*}

Easy Plugin Development for Pentaho

*Community Plugin Kickstarter

Motivation

*“ **Pentaho** is very well known for being a very good Business Analytics software, but is in fact much more than that; Pentaho is a great platform to build on top of. ”*

(Totally stolen from [Kaapa's](#) blog)

Goal

“The goal of CPK is to provide a simple and easy way to develop pentaho plugins that behave like packaged applications, simplifying its structure.”

(Aaaand **again** :P)

Plugin structure

UI

CDE* for dashboards/pages

Basic sitemap

Default CPK Template

Server-side

Kettle Transformations/Jobs

Java Classes

(Server Side Javascript)

* overall it depends on **Pentaho**, **CDF**, **CDE**, **CDA** and **CPF**.

Is it mandatory to have strong knowledge
of Java?

Not at all!

The idea is to use **Kettle** to implement server side logic.

File structure

The proposed stub is subject to updates but already close to a stable version.

```
CPK_Plugin
|-- dashboards
|-- endpoints
|-- static
|-- resources
|-- lib
`-- plugin.xml
`-- settings.xml
```

Accessing Endpoints and Dashboards

Endpoints are listed from the */endpoints* folder. Dashboards from the */dashboards* folder.

They can be accessed using:

/pentaho/content/pluginName/endpointName

Passing parameters to Kettle

*

Endpoint:

`paramcustomParam`

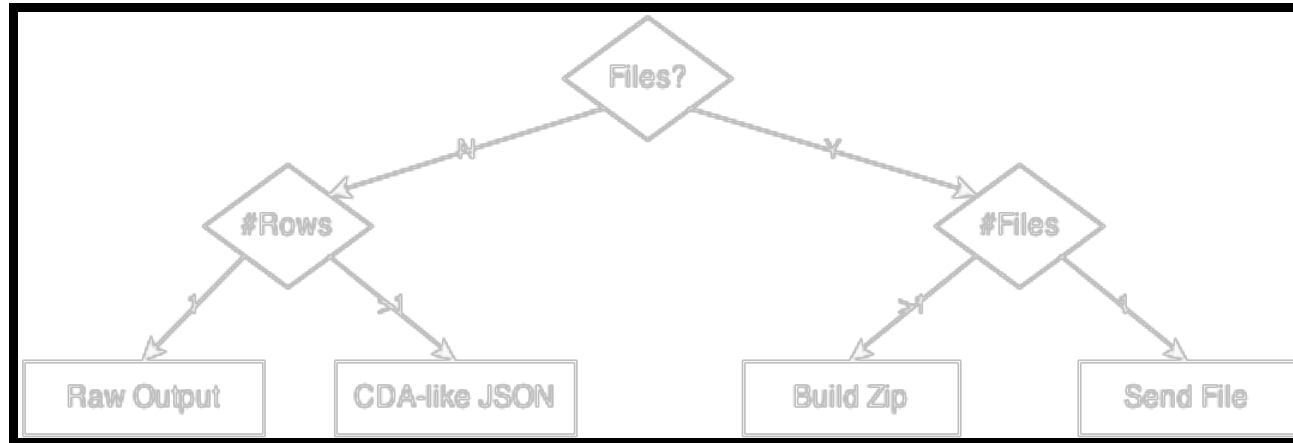
Kettle:

`${customParam}`

.../pluginName/endpointName?paramcustomParam=true

* Same mapping as CDA

Kettle Output



CVB^{*}

A Test Case for CPK

^{*}Community VFS Browser

Development driven by User Needs

Mozilla Metrics

Develop a plugin to browse a Virtual File System and download Files. The target case is **Hadoop** File System.

Start with an empty plugin

with a stub structure

```
CVB
|-- dashboards
    |-- initialTest.*
|-- endpoints
    |-- kettle
        |-- sampleTrans.ktr
|-- static
|-- resources
|-- lib
|-- plugin.xml
|-- settings.xml
```

Status

</pentaho/content/cvb/status>

Refresh

</pentaho/content/cvb/refresh>

Endpoints

- `initialtest` (dash): [**/pentaho/content/cvb/initialtest**](/pentaho/content/cvb/initialtest)
- `sampletrans` (trans):
[**/pentaho/content/cvb/sampletrans**](/pentaho/content/cvb/sampletrans)

Remove Sample Endpoints

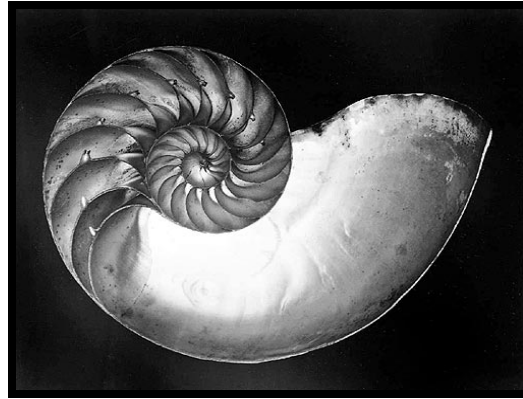
```
CVB
|-- dashboards
    |-- initialTest.*
|-- endpoints
    |-- kettle
        |-- sampleTrans.ktr
|-- static
|-- resources
|-- lib
|-- plugin.xml
|-- settings.xml
```

Add file list

Drop the filelist dashboard in the */dashboards* folder and the kettle transformations in */endpoints/kettle*.

```
CVB
|-- dashboards
|-- endpoints
    |-- kettle
|-- static
|-- resources
|-- lib
`-- plugin.xml
`-- settings.xml
```


#1



Refresh

</pentaho/content/cvb/refresh>

Endpoints

- `listfiles` (trans):
[**/pentaho/content/cvb/listfiles?paramURI=file://**](/pentaho/content/cvb/listfiles?paramURI=file://)
- `filelist` (dash): [**/pentaho/content/cvb/filelist**](/pentaho/content/cvb/filelist)
- `_others` (trans)

What about all the others with the underscore at the start?

Work in Progress

Filter out sub transformations that are not meant to be used
as endpoints.

Template / Sitemap

Sitemap is a part of the default CPK template. CPK provides an additional method to automatically fetch the dashboard list used by plugin UI.

</pentaho/content/cvb/getSitemapJson>

Change dashboard template (Edit Mode)

[**/pentaho/content/cvb/filelist?mode=edit**](/pentaho/content/cvb/filelist?mode=edit)

Try again!

[**/pentaho/content/cvb/filelist**](/pentaho/content/cvb/filelist)

Add other endpoints

`getLocations`: fetch the VFS locations from XML.

`downloadFile`: download files using CVB.

```
CVB
|-- dashboards
    |-- fileList.*
|-- endpoints
    |-- kettle
        |-- listFiles.ktr
|-- static
|-- resources
|-- lib
|-- plugin.xml
|-- settings.xml
```

#2



Refresh

</pentaho/content/cvb/refresh>

Activate Location Selector (Edit Mode)

</pentaho/content/cvb/filelist?mode=edit>

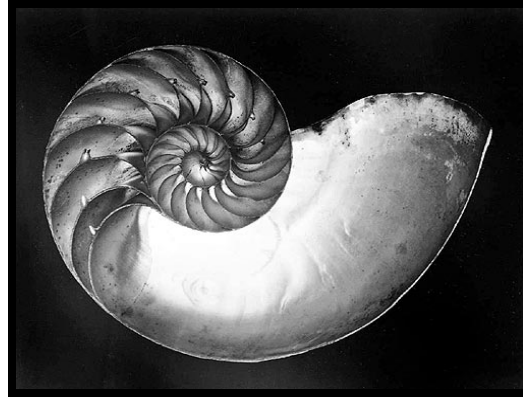
See it working!

</pentaho/content/cvb/filelist>

Add settings dashboard

```
CVB
|-- dashboards
  |-- fileList.*
|-- endpoints
  |-- kettle
    |-- listFiles.ktr
    |-- getLocations.ktr
    |-- downloadFile.kjb
|-- static
|-- resources
|-- lib
  |-- plugin.xml
  |-- settings.xml
  |-- cvb.xml
```

#3



Refresh

</pentaho/content/cvb/refresh>

Access Control*

Control top level locations.

Apply regular expressions to each of the locations.

```
<vfsLocation name="Home" topLevel="file://">  
  <access type="allow">.*Users/gundisalwa.*</access>  
  <access type="deny">.*Users.*</access>  
  <access type="allow">.*</access>  
</vfsLocation>
```

</pentaho/content/cvb/settings>

Pause for tea?



Back to CPK^{*}

Other Features and Roadmap

Access Control^{*}

Non admin users can not access dashboards, or endpoints, inside the */admin* folder, and the sitemap does not show them.

^{*}(Based on Pentaho roles)

Not admin

cvb/filelist?userid=pat&password=password

Admin

cvb/filelist?userid=joe&password=password

Download Zip^{*}

Although CVB does not expose that at the moment, folders can be downloaded in a .zip file.

CPK triggers zip when the kettle result is more than one file.

Test download

Next Steps: Control Panel

Administrative Features

- List existing plugins
- Detect if the plugins are up to date with the latest version of CPK
- Allow the creation of new plugins
- Allow to change plugin metadata
- List and register new endpoints (UI and code)

(Same **deal**)

Next Steps: Control Panel

Other goals

- Import UI from existing dashboards in solution
- Allow editing dashboards from this UI
- Submit marketplace metadata to *Pentaho*
- Generate distribution zip package
(Same **deal**)

How to build it?



Using CPK!

THE END