



Service Description - Pacemaker Process Pipelines v2.x

Technical Component Informationen

Company	TechDivision GmbH
Name	Pacemaker Process Pipelines
Version	v2.x
Web	Pacemaker Process Pipelines ,
Open Source	No
License	Proprietary
Pricing	n/a
Adobe Commerce Version	>= 2.4.4
PHP Version	>= 7.4

General

Pipelines are a common used solution to enable stable, reproducible and maintainable execution of processes.

A significant advantage of process pipelines is the ability to execute multiple processes simultaneously. That is practical for performance reasons but also urgently required in some scopes.

- It is possible to model any process with **Pacemaker Process Pipelines** and their mechanisms
- It is possible to split existing workflows and functions into smaller units, known as **steps**, to follow the single-responsibility principle
- It is possible to define the conditions of **pipelines** and their **steps** to allow or block their execution
- It is therefore possible to define dependencies between **pipelines** and/or their **steps** with such a condition
- Monitoring of each process and its **steps**, either through the **Command Line (CLI)** or through the **Adobe Commerce Admin GUI** (Adobe Commerce backend)
- Providing information with log outputs and process **IDs**
- Optimize the use of available resources to improve system performance in the right places significantly
- Application-side monitoring solution to detect process failures of pipelines

Feature Overview

Function	Description
Process integration through Adobe Commerce Standard functionality	Integration of multiple modules gets performed using the defined component, which uses standard Adobe Commerce functions and building blocks provided for execution
Creation, overview, and detail view in the Adobe Commerce backend	Pipelines can get created from the Adobe Commerce Admin Backend. Furthermore, the component provides a view of pipelines and their associated steps .
Creation, overview, detailed view, and control with command line input	Pipelines can get created and controlled with the command line. Furthermore, commands for detailed output and overview get provided.
Distributed processing and parallelization	The message queue or consumer concept of Adobe Commerce allows multiple processes to get used simultaneously to process functions or workflows. The logic of ranking (prioritization) and actual processing get separated within the component.
Process modeling based on conditions	The Process Pipeline System uses Conditions to check the sorting of pipelines and the steps . These implementations can be used to create dependencies between steps and pipelines . It is also possible to check for any other factor in the Adobe Commerce system to allow or disallow the enqueueing.
Detailed log output	The log outputs can get configured for all Pacemaker Process Pipelines Management sub-areas and provide information on the processing sequences.
Clean-Up Pipelines	The component provides a pre-built pipeline configuration for cleaning up Pacemaker Process pipelines in Adobe Commerce Systems. This pipeline is fully configurable.
Pipeline Monitor	The component provides its built-in application-side monitoring tool. The monitor is configurable and reports any anomalies in the process pipelines system using email.
Extensibility / Extension Points	The Process Pipelines system is also easily extensible because of using Adobe Commerce concepts and features. In addition to DI extensions, XML merges can also be used to declare pipelines. The component also provides usable classes and methods for custom pipelines or customizations to existing pipelines .

Distributed processing/process modeling

- To separate the prioritization of the **pipelines** and their steps from the actual processing, the **Heartbeat**, **Pipeline Runner** and **Pipeline Executors** implementations get used
- The **Heartbeat** is the impulse that represents the start of the condition check and is executed in regular intervals (every minute)
- By the following check of the **Conditions pipelines** are created and the steps get enqueued if the condition check was successful
- Parallel running "consumers" fetch the steps one after the other and instantiates **executors** to process the information

Pipeline Monitor

- The Pipelines Monitor is an **onboard tool** to report anomalies within **pipelines** and their processing
- It is designed to detect and report irregularities in processing time
- Due to the enormous potential complexity, automatic correction is not provided

Extensibility/Extension Points

As described, the component offers many concepts and code-based extension points for extension and adaptation of **pipelines**.

What is included in the license price?

- Major/Minor Updates
- Security Updates
- Bugfixes

What the component cannot do

Install & Play

- An infrastructural installation and configuration are required to use all necessary components (**MySQL-MQ/RabbitMQ, SupervisorD**).

Configuration of Pipelines with Drag&Drop

- Currently, **pipelines** can only get declared as **XML** or **PHP** implementations
- All necessary classes (conditions, executors) must be customized with **PHP** implementation and **DI** if necessary

Automatic correction

- Currently, the component only identifies anomalies/errors using the **Pipeline Monitor**
- Automatic **pipeline** correction or steps get not implemented; this requires manual error correction

Requirement for use

To use the Pacemaker Components Process Pipelines in Adobe Commerce, the following points must get observed

- Contract
- [Technical conditions](#)

Glossary

Function	Description
Pipeline	A pipeline is the chaining of commands.
Step	A Step is a chain link within a pipeline .
Heartbeat	The heartbeat is the steady pulse that triggers the condition check to create and enqueue pipelines and their steps .
Pipeline-Runner	The pipeline runner is a preconfigured Adobe Commerce consumer that exchanges dedicated information with a message queue .
Pipeline-Executor	<ul style="list-style-type: none"> • Pipeline executors represent PHP classes and their business logic • Pipeline executors are instantiated and executed by pipeline runners
Pipeline-Condition / Step-Condition	<ul style="list-style-type: none"> • Conditions are PHP classes that get used to checking pipelines and steps • Conditions get used to decide whether a step gets executed and a pipeline gets created
Consumer	<ul style="list-style-type: none"> • A consumer receives messages • A consumer knows which message queue to consume • A consumer can assign the message processors to a specific queue
Message Queue	<p>The message queue Paradigm is the publisher/subscriber pattern and is typically part of a larger message-oriented middleware system.</p> <ul style="list-style-type: none"> • MQF