

Arrowhead 4.6.0.

Repository: [eclipse-arrowhead/core-java-spring](https://github.com/eclipse-arrowhead/core-java-spring) · Tag: [4.6.0](#) · Commit: [5605ba3](#) · Released by: [borditamas](#)

Arrowhead 4.6.0

The release zip files contains pre built JAR files of the core systems, certificates, scripts. The sources zip contains the source code for this release.

What's Changed

General:

- The core system are logging into separate files but they also logging into the database. Since they use the same table, using the contents of this table were confusing. We changed the logs table to store the source system's name in each entry. We also increased the precision of the entries' timestamp from seconds to milliseconds.
- We provided an endpoint in every core system to list, filter and sort these log entries. For example, in the Service Registry log entries can be accessed using the `/serviceregistry/mgmt/logs` endpoint (GET request).
- Fixing null general advertisement message in the Relay Library (related issue: 354)
- Various bug fixes

Core common module:

- Fixing null flag in `OrchestrationFlags` class (related issue: 351)
- Fixing Resource leak in `JDBCConnectionFactoryForLog4J2` class (related issue: 353)

Service Registry

- We are calculating and storing the address type (DNS name, IPv4 or IPv6) of systems. This information can be used to filtering providers during service discovery.

Orchestrator:

- New orchestration flags are introduced: `ONLY_IP_ADDRESS_RESPONSE`, `ONLY_IPV4_ADDRESS_RESPONSE` and `ONLY_IPV6_ADDRESS_RESPONSE`. The providers' address types must meet the requirements specified by the flag values in the orchestration response.

Gatekeeper:

- Fixing Potential Thread-safety violation in `RelaySubscriberDataContainer` class (related issue: 356)
- In the previous versions. the ActiveMQ servers (brokers) automatically closed all communication queues and connections in case of inactivity. However, in some cases the brokers were unable to do this task and this results stuck "zombie" queues and connections doing nothing but still using resources. In version 4.5, we refactored the Gatekeeper system to actively close unused communication queues and connections. The ActiveMQ task now only serves as a backup method.

- New endpoint: /gatekeeper/init_multi_gsd (POST request). It performs a global service discovery with using multiple service query form. It supports the Workflow Choreographer's plan execution validation in intercloud environment.

Gateway:

- Similar to Gatekeeper, the Gateway system actively close unused communication queues and connections to the ActiveMQ brokers.
- New endpoint: /gateway/close_sessions: Immediately closes the specified (by their ports) active sessions. It supports the Workflow Choreographer's plan execution in intercloud environment.

QoS Monitor:

- One of the tasks of the QoS Monitor to collect data about intercloud communication through ActiveMQ brokers. During these tests, the QoS Monitor now actively close communication queues and connection to the ActiveMQ servers after the necessary measurements are done.

Workflow Choreographer:

- The Workflow Choreographer improved in various aspects, including:
- exhaustive validation and normalization during plan registration such as circle detection between actions and between steps of each action; step graph normalization to eliminate unnecessary dependencies, etc.
- exhaustive plan execution validation before performing a plan: the choreographer made sure the every service that is necessary to execute the plan is available.
- can use intercloud orchestration to execute a plan (if the plan and the starter of the plan allow it).
- more robust plan execution: the Choreographer can change executors during the plan execution if previously selected executors are failed to finish the step.
- option to execute a plan multiple times (the maintainer of the Choreographer can specify a maximum iteration value via configuration)

Full Changelog: [4.5.0...4.6.0](#)

—

This release has 2 assets:

- Source code (zip)
- Source code (tar.gz)

Visit the [release page](#) to download them.