Eclipse Arrowhead 5
development status & feature highlight

Tamás Bordi | AITIA International Inc.

The Arrowhead fPVN project is supported by the Chips Joint Undertaking and its members, including the top-up funding by Finland, Denmark, Sweden, Spain, Italy, Rumania, Portugal, Hungary, and France

ARROWHEAD
fPVN

Development Roadmap

borditamas opened on Mar 6 · edited by borditamas — Edits

Core Systems

System	Code	Unit Tests	Documentation	Test build	Planned Release
ServiceRegistry	ready	in progress	in progress	14.03.2025	31.07.2025
ServiceOrchestration (Dynamic)	ready	pending	in progress	23.04.2025	31.07.2025
ServiceOrchestration (Simple store)	in progress	pending	pending	tbd	tbd
ServiceOrchestration (Flexible store)	pending	pending	pending	tbd	tbd
ConsumerAuthorization	in progress	pending	in progress	01.06.2025	31.07.2025
Authentication	ready	in progress	ready	23.04.2025	31.07.2025

Support Systems

System	Code	Unit Tests	Documentation	Test build	Planned Release
Blacklist	ready	in progress	pending	23.04.2025	31.07.2025
TranslationManager	pending	pending	in progress	tbd	01.09.2025
QoS Evaluator	pending	pending	in progress	tbd	15.11.2025

Migration Guideline from v4 to v5

Skeleton draft	Skeleton final	Ready to review	Publish
10.06.2025	24.06.2025	25.07.2025	31.07.2025

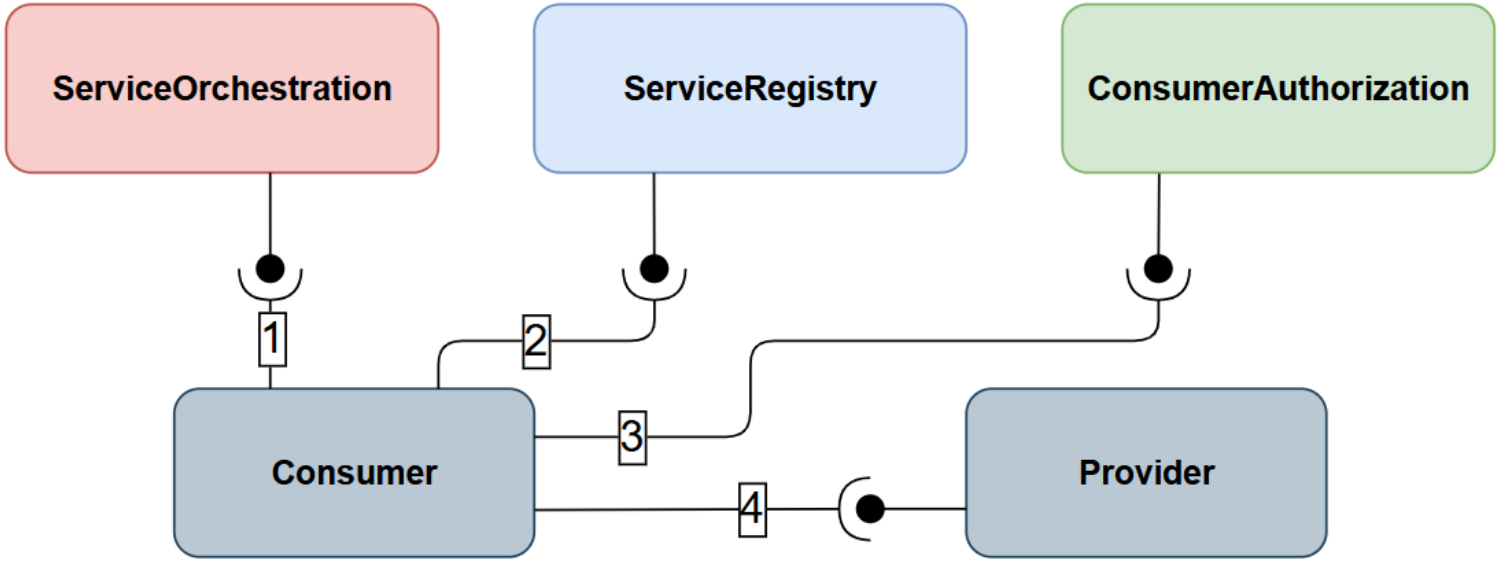
https://github.com/eclipse-arrowhead/roadmap/issues/93

ARROWHEAD fPVN

Main Focuses of 5th Generation

- More flexibility
  - Individual and cooperative working modes (Use case driven Local Clouds)
    - → Application systems are more autonomous and have more responsibilities, or
    - → Application systems could focus only to performing its micro services and let Arrowhead Framework manage all the other necessary tasks and responsibilities as much as possible
  - Various ways of system authentication
  - Various ways of service consumption authorization
    - → Policy based authorization rules
    - → Various type of authorization tokens
  - Various orchestration strategies and types
- Advanced translation support
  - Between communication protocols
  - Between payload data formats and data models
- Advanced user support
  - Global System-of-Systems Description (GSoSD)
  - Dedicated documentation site for the implementation

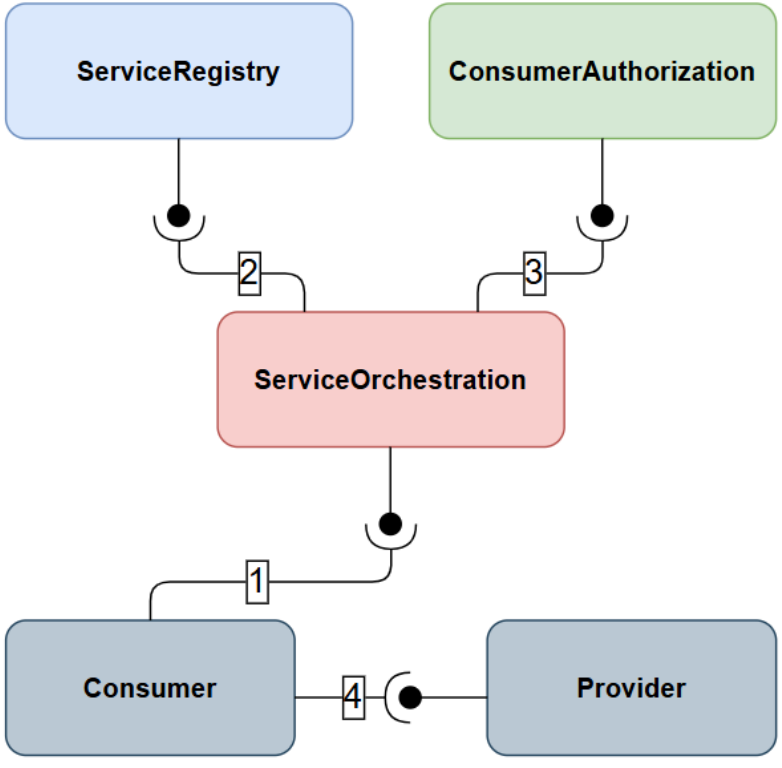
Flexibility | Individual working modes



1) Consumer's orchestration only for a service identifier [orchestration:pull]
2) Consumer's discovery for the service details [serviceDiscovery:lookup]
3) Consumer's request for authorization token(s) [authorizationToken:generate]
4) Actual service consumption

ARROWHEAD
fPVN

Flexibility | Cooperative working modes



1) Consumer's orchestration for everything what is needed [orchestration:pull]
2) ServiceOrchestration's discovery for the service details [serviceDiscovery:lookup]
3) ServiceOrchestration's request for authorization token(s) [authorizationToken:generate]
4) Actual service consumption

ARROWHEAD
fPVN

Flexibility | Ways of system authentication

Three authentication policy is offered:

- Declared

  The Core and Supports systems simply accept the identity provided by the application systems. Not recommended to use in production environment.

- Certificate

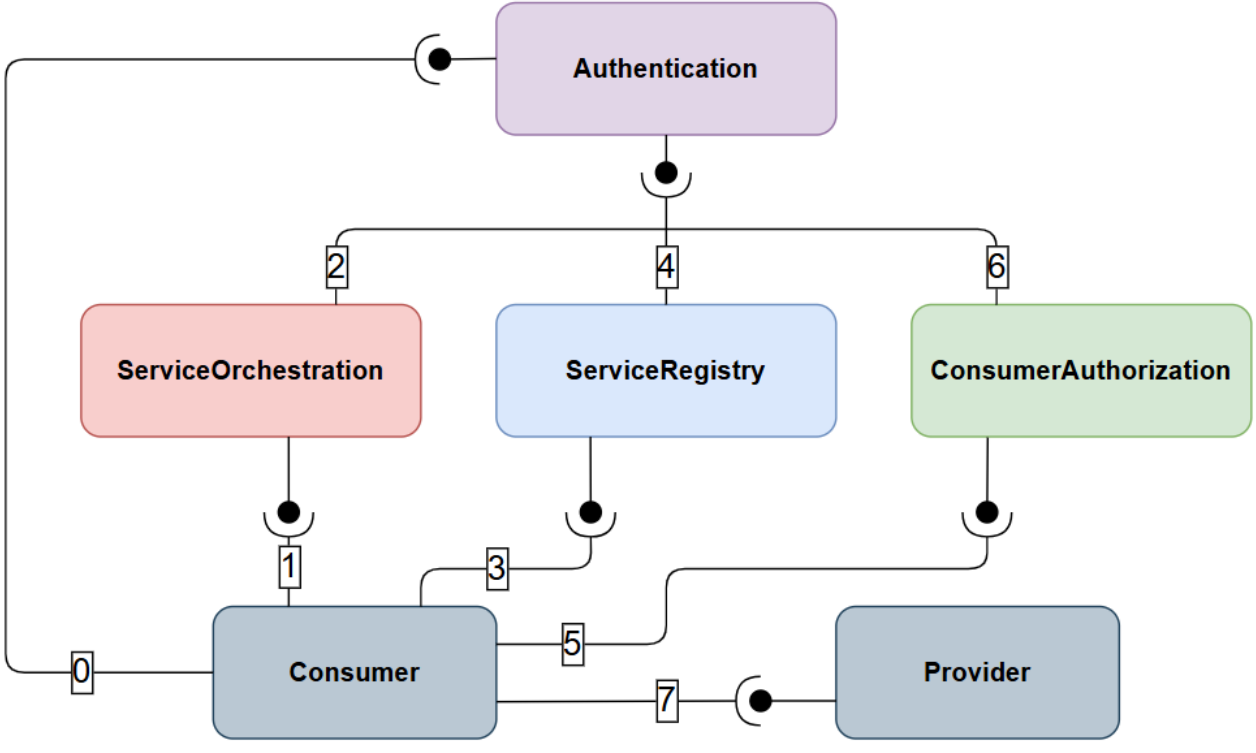
  Requires client side system profile certificate being provided by the application systems. The issuer and the identity related content of the certificate are always verified during every interaction with the Arrowhead Core and Support systems.

- Outsourced

  Requires the **identity** service being provided by a Core/Support system within the Local Cloud. This dedicated system can be the official **Authentication Core System** or any third party solution that implements this service. Identity tokens should only be shared with the trusted Arrowhead Core and Support systems and never with the providers in order to prevent identity theft.

ARROWHEAD
fPVN

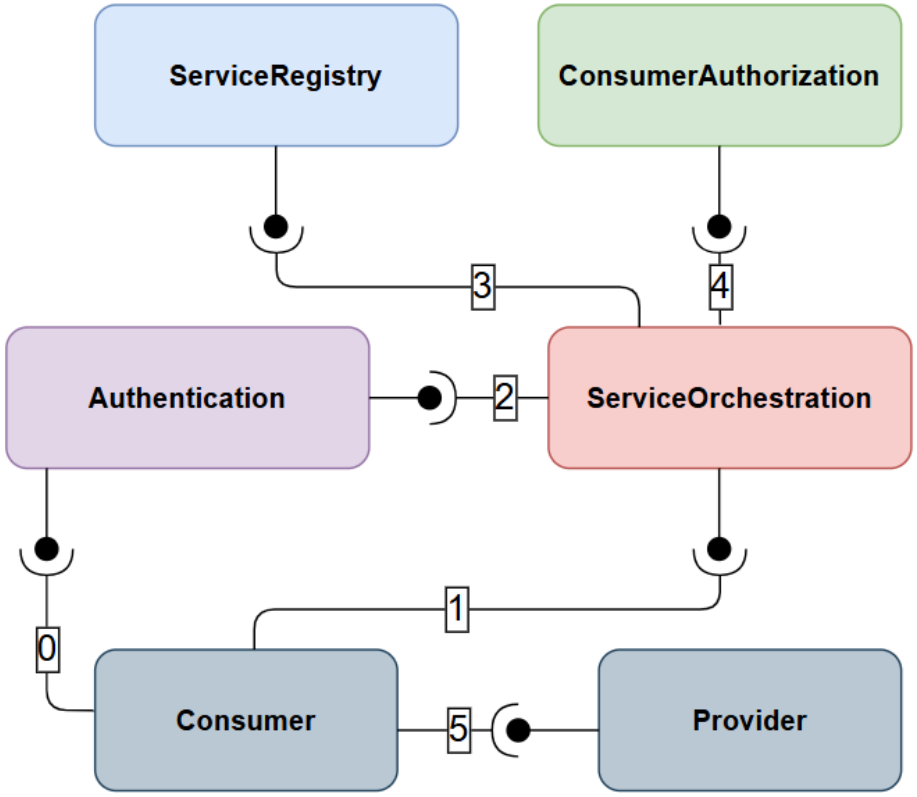
Flexibility | Outsourced authentication (Use case A)



0) Consumer's login for an identity token [identity:login]
1) Consumer's orchestration only for a service identifier [orchestration:pull]
2) ServiceOrchestration verifies the consumer's identity token [identity:verify]
3) Consumer's discovery for the service details [serviceDiscovery:lookup]
4) ServiceRegistry verifies the consumer's identity token [identity:verify]
5) Consumer's request for authorization token(s) [authorizationToken:generate]
6) ConsumerAuthorization verifies the consumer's identity token [identity:verify]
7) Actual service consumption

ARROWHEAD
fPVN

Flexibility | Outsourced authentication (Use case B)



0) Consumer's login for an identity token [identity:login]
1) Consumer's orchestration for everything what is needed [orchestration:pull]
2) ServiceOrchestration verifies the consumer's identity token [identity:verify]
3) ServiceOrchestration's discovery for the service details [serviceDiscovery:lookup]
4) ServiceOrchestration's request for authorization token(s) [authorizationToken:generate]
5) Actual service consumption

ARROWHEAD
fPVN

Flexibility | Consumer authorization

Providers can define their own rules, but management level rules always have priority.

The followings authorization rules are expressible:

- this service/operation is accessible anyone within the local cloud
- this service/operation is accessible anyone within the local cloud, except these consumers (blacklist)
- this service/operation is accessible anyone from the given list (whitelist)
- this service/operation is accessible anyone within the local cloud having the specified meta data

The following security levels are available on provider side

- None
  - No verification on provider side
- Certificate
  - Provider verifies only that the consumer is part of the same Local Cloud
- Token (expirable)
  - Provider verifies that the consumer has proper permission to consume its service operation at the moment of the consumption attempt

ARROWHEAD
fPVN

Flexibility | Consumer Authorization token types

- Simple Time Limited
  - The token expires within a given time.
  - Provider has to verify it with the ConsumerAuthorization Core System [authorizationToken:verify]
- Simple Usage Limited
  - The token expires after a certain number of uses.
  - Provider has to verify it with the ConsumerAuthorization Core System [authorizationToken:verify]
- Self Contained – Base64
  - The token expires within a given time.
  - Has a payload: consumer, provider, cloud, service, operation, expiry.
  - Provider has to verify it itself.
  - Can be encrypted.
- Self Contained – JWT
  - The token expires within a given time.
  - Has a payload: consumer, provider, cloud, service, operation, expiry.
  - Provider has to verify it itself.
  - Can be encrypted.
  - Signed by the ConsumerAuthorization Core System (requires SSL certificate) -> Provider can verify it.

Flexibility | Ways of service orchestrations

The three orchestration strategies is divided into three different systems:

- Simple store (peer to peer rules, no actual service details are provided)
- Flexible store (name and/or metadata based rules, actual service details are provided)
- Dynamic (on the fly discovery, actual service details are provided)

Two ways to use the orchestration service:

- Pull type of orchestration

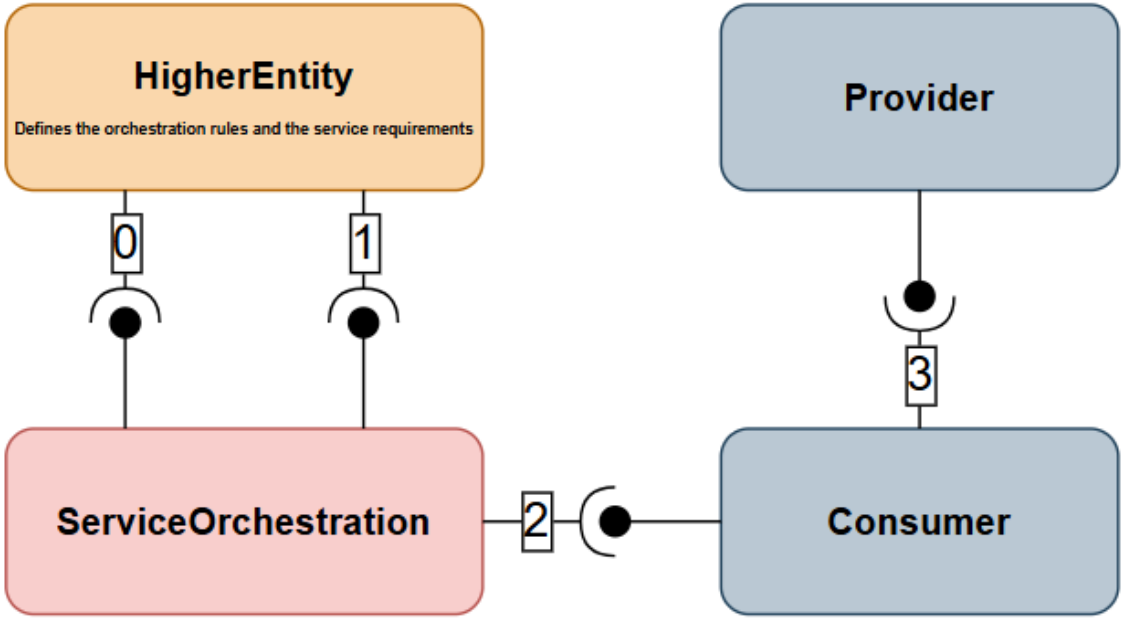
  A consumer system consumes the pull operation in order to immediately obtain the matching service instances.

- Push type orchestration

  A consumer system consumes the subscribe operation in order to receive new orchestration results every time when a higher entity triggers a related orchestration process. Hence, the actual matching service instances are being sent to the consumer without its direct request.

ARROWHEAD
fPVN

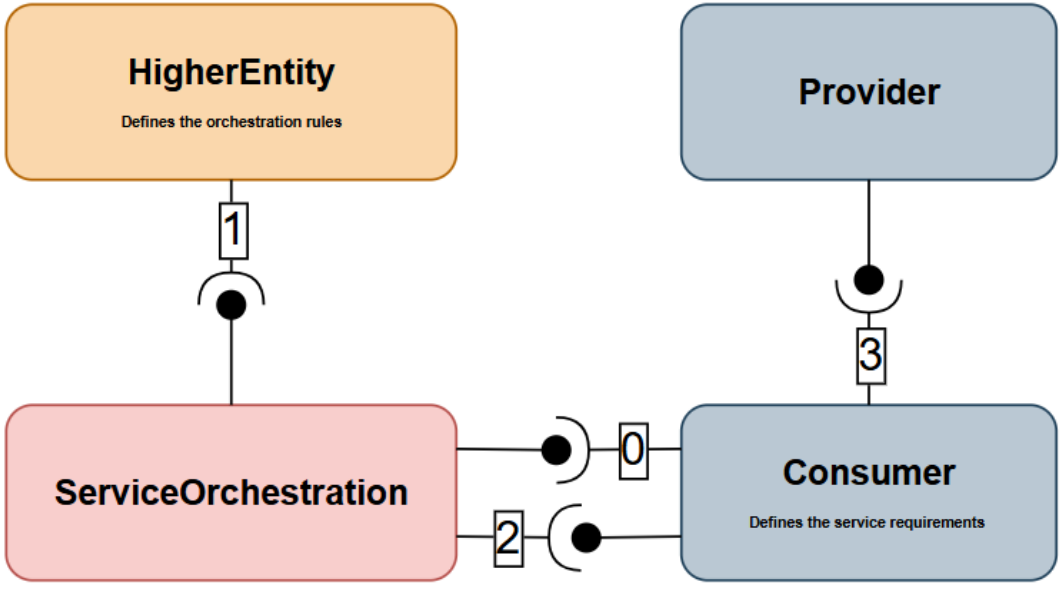
Flexibility | Push orchestration (Use case A)



0) A higher entity subscribes the consumer [orchestrationPushManagement:subscribe]
1) A higher entity trigges the orchestration process [orchestrationPushManagement:trigger]
2) Consumer receives the new orchestration results via its notify endpoint.
3) Consumer connects to the new provider

ARROWHEAD
fPVN

Flexibility | Push orchestration (Use case B)



0) Consumer subscribes for a service [orchestration:subscribe]
1) A higher entity triggers the orchestration process [orchestrationPushManagement:trigger]
2) Consumer receives the new orchestration results via its notify endpoint.
3) Consumer connects to the new provider

ARROWHEAD
fPVN

Translation Support (in v5.1.0)

Outsourcing of translation tasks to providers

Capable to translate one or more input protocol to one output protocol.

Can be developed individually.

Can be deployed in multiple instances.

Can be AI powered.

Capable to translate between two different data format in both direction.

Capable to translate between two different data model in both direction

.

Can be developed individually.

Can be deployed in multiple instances.

Can be AI powered.

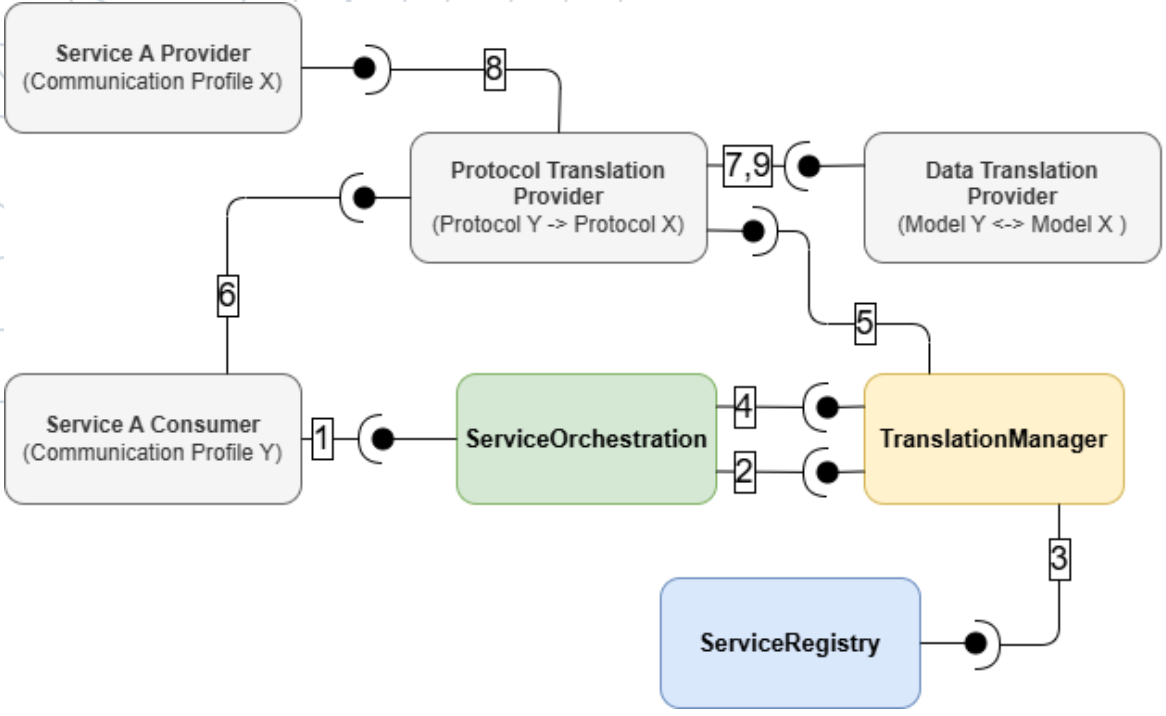
Translation Support (in v5.1.0)

Integration into the orchestration process (divided responsibilities)

ServiceOrchestration	TranslationManager
Decides that translation is necesarry or not.	Decides that a translation job can be performed or not.
Initiates a translation job when necesarry and is enabled.	Choose protocol and data translator provider for a translation job.
Choose a suitable service provider based on the translation possibilities.	Initiates the preparation for a translation job.

ARROWHEAD
fPVN

Translation Support (in v5.1.0)



1. Orchestration request
2. Translation discovery request
3. Translation service discovery
4. Init translation negotiation
5. Translation negotiation
6. Service consumption request
7. Input data model translation
8. Actual service consumption
9. Output data model translation

Still under specification:
https://github.com/eclipse-arrowhead/roadmap/issues/92

ARROWHEAD
fPVN

Advanced User Support

Generic System-of-Systems Description (GSoSD)

A high-level architectural description of what problems the Core Systems solve and how they interact.
https://github.com/eclipse-arrowhead/roadmap/tree/main/5.0%20Draft/GSoSD

Dedicated documentation site for the reference implementation

Detailed System and Service documentation, API descriptions, examples and tutorials.
https://aitia-iiot.github.io/ah5-docs-java-spring/home/welcome/

Test builds are published on this site as well:
https://aitia-iiot.github.io/ah5-docs-java-spring/downloads/test_builds/

ARROWHEAD
fPVN

Eclipse Arrowhead 5
development status & feature highlight

Q & A