

Organization

Missing project-wide Arrowhead community

Currently each WP working on their own strategy how to integrate or implement solutions into arrowhead. The biggest problem here is that the WPs are encapsulated from each other.

Suggestion:

1. community wiki (e.g. mediawiki) to share information between members and public. So we can decrease the pressure for trainings from the developers. This means that every member can create pages or edit other pages.
2. a repository/project for community demonstrators because the current examples in the repositories are too high level or misleading. Another advantage is that the community itself maintain their own demonstrators.

Organize knowledge for Repositories in Wiki instead of README

The problem is especially with the arrowhead core repository. The readme is much too long. Information can be found only with difficulty, since one simply does not see, when necessary passages begin and end.

Suggestion: Use the wiki tab for deeper explanations and use the Readme for short explanations like how to build/contribute and usage.

Quality-Assurance

Main punchline. Pretty much everything I checked out the first time, didn't work out of the box. Even though it was in the master branch.

Maybe I got a wrong assumption, but with a master/development strategy, I assume that master branch should work.

Setup Core Systems with Docker

(last checked - 10.02.2021)

- Tried to use everything under **docker/**
- Had a lot of routing issues (docker ip -> host ip--> ---- net ----- <-- client ip <-- docker ip>)
- So continued and tried **docker_all/**
- <https://github.com/eclipse-arrowhead/core-java-spring/pull/240>
- <https://github.com/eclipse-arrowhead/core-java-spring/pull/241>
- Punchline here: the docker part was outdated and didn't work out of the box (master and dev branch)
- Reason was an **old SQL Script / missing components** (docker was just untested)

Getting Arrowhead Core Systems running with Docker as Client-Server-Architecture

(past - 10.02.2021)

- ♦ Unable to make a successful setup
- ♦ Even in a special training **with Svetlin** we couldn't make it work.
- ♦ As a result we ended up starting everything without docker.
- ♦ I believe if I find the time, I can make it work now. (Host network/ not using NAT). And also I got other stuff to do.
- ♦ But the main point here: It should show that the documentation is just not enough.

Current Status of Docker Usage

(last checked - 10.02.2021)

- ♦ Using **Docker/**
 - Images relate to hub.docker from <https://hub.docker.com/u/svetlint>
 - Images seem to be 1-2 years old.
 - Which version is that?
 - This does not correlate with the master branch
 - Is this discrepancy save to use?

♦

-
- ♦ Using **Docker_all/**
 - Totally unsure about the status of this component
 - I think on the master branch it's broken
 - On the development branch (might work?)
 - Automated Verfication would be good

SOS-Examples weren't working / building (last checked - 10.02.2021)

In the beginning:

- I had big trouble getting this to work
- Students had big trouble getting this to work
- Colleagues had big trouble getting this to work
- ♦ One Issue was building the project
 - Last commit updated the project **client library** from **v4.1.3.11** to **v4.1.3.13**
 - I think currently, it is not possible to build this on your own. It's **closed source** or?
 - In the README.md Usage of github packages are mentioned as well as how to activate it.
 - Activation for this specific project? One needs to figure it out on their own..
 - Example: Where do I know what URL i should have used??

```

<profiles>
  <profile>
    <id>github</id>
    <repositories>
      <repository>
        <id>central</id>
        <url>https://repo1.maven.org/maven2</url>
        <releases><enabled>true</enabled></releases>
        <snapshots><enabled>true</enabled></snapshots>
      </repository>
      <repository>
        <id>github</id>
        <name>GitHub OWNER Apache Maven Packages</name>
        <url>https://maven.pkg.github.com/arrowhead-f/client-
library-java-spring</url> <-----
      </repository>
    </repositories>
  </profile>
</profiles>

<servers>
  <server>
    <id>github</id>
    <username>USERNAME</username>
    <password>TOKEN</password>
  </server>
</servers>

```

- ♦ Second Issue was about creating the Auth Rules
 - Doc mentions the Swagger-API with Reference to [POST /authorization/mgmt/intracloud](#)
 - From there you get a json which you should send

```

{
  "consumerId": 0,
  "providerIds": [
    0
  ],
  "interfaceIds": [
    0
  ],
  "serviceDefinitionIds": [
    0
  ]
}

```

- Nowhere is mentioned how to get these Values now.
- Since it is an **Examples project** it would be nice to get more help here

Third Issue was about getting it to work with SSL

- Took me ages with no luck!
- Through the workshop video from Infineon with Svetlin, I saw how he worked with the system.
- Side Note: Even though the Tutorial was very helpful, the video shows how error prone the setup can be this way. When even the developer can't make it work without errors in a prepared session.
- Managed to request the interfaces and found finally the Problem
 - CORE Services register using **HTTP-SECURE-JSON**
 - The SOS-Examples use **HTTPS-SECURE-JSON**
 - Finding this little difference?? as a beginner?? in a tutorial?

(Minor) Core Systems in Master can't be build - Student work

(last checked - fixed since Nov 23, 2020)

- ♦ <https://github.com/eclipse-arrowhead/core-java-spring/issues/273>
- ♦ Problem here was, that having a already compiled version -> Library is in the local cache
- ♦ A fresh checkout and build, the library is missing
- ♦ Issue can happen, nothing so serious
- ♦ Thing to think about here, how to deal with that in the future
- ♦ As seen from Ticket, problem stayed unnoticed for weeks (at least one)
- ♦ Also it was on the Master branch
- ♦ I told students to get familiar with the core systems. Then the gap:
 - Students trying to figure out what they did wrong or if it is their fault
 - Time passes until the next meeting where they're telling me their problem
 - Me verifying the issue
 - Creating a ticket
 - Waiting for a response
 - Fix occurred
- ♦ CI/CD could be helpful to periodically check if the master is still working (Would have diminished the issue)
 - Nothing fancy. Exists since ages. You can even insert icons, which display directly on github the current status

Trying out Kalix Examples

(last checked - 10.02.2021)

- ♦ repo: <https://github.com/arrowhead-f/arrowhead-kalix-examples>
- ♦ Started with checking out the **master** branch -> again, didn't work out of the box
 - ◊ Syntax error on building
 - ◊ <https://github.com/arrowhead-f/arrowhead-kalix-examples/issues/4> (got fixed)
- ♦ After the issue got fixed, was able to at least build the project
- ♦ I tried to build my own little PoC
 - ◊ Got stuck on setting up the client/provider for a remote core server system
 - Opened a ticket.. responding to ticket stopped
 - Wrote over slack channel.. no response
- ♦ Result: Abandoned work/experimenting with Kalix

Python library

(last checked - 09.02.2021)

- ♦ I know he develops this in his free time (again It's just feedback, I don't want to blame here !)
- ♦ Pretty happy to see the switch to async/await that is about to come!
- ♦ Initially, tried out the library from the **master** branch.
 - ◊ -> Again, didn't work out of the box
 - ◊ -> There were a lot of examples, from which some were/(are still) not meant to be used anymore
 - ◊ Even with examples that should be used I wasn't able to get it working
 - ◊ Reason is a problem during the library upload
 - See: <https://github.com/arrowhead-f/client-library-python/issues/16>
 - On Feb 2, 2021 (~9 days ago?) marked as **fixed with still some issues** . Haven't tested it yet
- ♦ no configuration possible (now possible) (minor thing / not needed to be mentioned)

ESPLibrary

(last checked - 09.02.2021)

- ♦ Repo: <https://github.com/arrowhead-f/ArrowheadESP>
- ♦ Says it's for ESP8266 and ESP32.
- ♦ Checked out **master** and tried to build it. Didn't work...
- ♦ Missing NTP dependency Issue (maybe forgot to mention in the README)
 - ◊ `arduino-libraries/NTPClient @ ^3.1.0` <- missing
- ♦ A student tried to work with the library for his project and couldn't make it work
- ♦ I encouraged the student to open a ticket and get help from the developer (encouraging the normal workflow with open-source projects)
 - ◊ until now, no response on his ticket
 - ◊ **I/ME** then took a peak into the source code and found

```
#ifdef ESP8266
#include <ESP8266WiFi.h>
```

```
#include <WiFiUdp.h>
#include <ESP8266WebServer.h>
#include <ESP8266WebServerSecure.h>
#include <ESP8266mDNS.h>
#endif

#ifdef ESP32
// TODO
#endif
```

- I'll take a further guess that it's about BearSSL which seems to not support ESP32. Probably, that's why development stopped there. But I really don't know.

Open-Source:

client-library:

- ♦ Issue that not all source code is available (see: client-library issue)
- ♦ Details from Paul
- ♦ Backup Explanation

```
Hi Szventlin,
Ingo here. We had a little Workshop meeting and also we met briefly during
the Infineon training.
Just wanted to ask. Do you got a minute?
I tried to get back to basic with Arrowhead and SSL. What I did:
checkout master of core components
run it with testcloud2 certs
checkout sos-examples
try to build sos-examples
here I got the first issue: it's trying to get the client-lib 4.1.3.13. But
this doesn't exist in the repos. On the master it's just up to 4.1.3.7.
Oddly, on the dev branch it even just goes to 4.1.3.5(?).
I then tried to build the client library. But even this doesn't work. It's
trying to find the core-client-skeleton version 4.1.3.11 but this didn't
exist. In the repo (client-library-java-spring) i can only find version
4.1.3.7.
Where can I build the core-client-skeleton now? I hope this makes somewhat
sense. I just want to get the sos example working.
```

■

Summarize:

- It is hard to keep up motivation, when everything is always broken. I can never be sure if it's me or just the repo.
- The lack of (working) examples to see how it is supposed to work, makes it hard to implement new concepts
- More documentation about how to work with the system would be tremendously helpful. (Workflow!)

Suggestions:

Workshops and Material:

- * Making ***Videos*** and ***Material*** from Workshops public
- * This way, we can reference this to students as a help (and knowing it's public content)
- * Notes:
 - * Last officially documented Workshop was 2016 -
<http://www.arrowheadproject.eu/events/past-events/>
 - * Infineon Workshop was indeed helpful
 - * Video Material is on private owncloud
 - * I'm not sure If I'm allowed to share that with students.
- * * Would be better to upload/host workshops in public (e.g. youtube) so I can link

Continue work on SOS-Examples

- ♦ Improving documentation on each example
 - ◊ Starting with (at least mentioning) starting up required core components.
 - ◊ Telling which certificate stack is used
 - ◊ Showing the whole process of getting example running (even with pictures if necessary)
 - Show the way, how it's supposed to be done (knowing that new people will probably then always use this way):
 - Over swagger api? then swagger api.
 - Mgmt tool? Then mgmt tool.
 - Cert Authority? then cert authority.
 - curl calls? then curl calls.
 - ◊ Linking to pdf's on how to make own certs (whatever is required for given example)
 - ◊ With each example, introducing a new component and explaining what it does (in comparison to the example before)

Automated Software Quality Control (CI/CD)

- ♦ Some automated reference that shows the new developer if the current software on **master** (according to the tests) is running as expected.
- ♦ This would avoid ***syntax errors*** on master branches as well as ***missing libraries issues***
- ♦ Also, the new developer knows, that it's more likely his/her fault rather than the software/library

Organize knowledge for a repository in Wiki module

Move knowledge like meta information, program/api structure or addition documents and data to the wiki module of the repository. For example, it is very difficult to find or search for information in the repository of the core framework because the information is very confusing.

Community helps community

Maybe it could be very useful to create wiki (e.g. mediawiki) to organize community information accessible for everyone. So it is easier to find information and project members can share information without borders. On the other hand it is particularly hard to find the suitable contact person for a special topic. Further mailinglists could be created for this:

- ♦ developers: addresses all developers in the project
- ♦ all: addresses all project members

Repository/Project in GitHub for Demonstrators and community examples