

Vertx subsystem in WildFly

There is a [Vertx extension for WildFly](#) which can solve the problem in the [Configure Vertx Option discussion](#).

Capabilities in the extension

In general, the extension tries to introduce the capabilities from Vert.x to WildFly following the WildFly way.

Configure VertxOptions following WildFly way

The extension provides a Vert.x subsystem to configure the VertxOptions following the WildFly way, like:

```
Java
[standalone@localhost:9990 /]
/subsystem=vertx/vertx-option=vo:add(max-worker-execute-time=60,
max-worker-execute-time-unit=SECONDS)
{"outcome" => "success"}
[standalone@localhost:9990 /]
/subsystem=vertx/service=vertx:write-attribute(name=option-name,value=vo)
{
  "outcome" => "success",
  "response-headers" => {
    "operation-requires-reload" => true,
    "process-state" => "reload-required"
  }
}
```

Each AttributeDefinition in the extension corresponds to an option item in [VertxOptions](#), and is maintained manually, which means there is no automatic way to convert the option item to AttributeDefinition in WildFly.

There is only one Vertx currently in the extension as a Vertx instance is the entrance to utilize the significant components from Vertx ecosystem. But having multiple Vertx instances is doable too if necessary.

The Vertx management resource has an `option-name` attribute to refer to the defined VertxOptions, once set, it needs reload as it affects the runtime.

Configure clustered Vertx instance (experimental)

Vert.x has capability to [compose a cluster using Infinispan](#), which looks great if the Vertx instance inside WildFly can compose a cluster with external Vertx instances to benefit the distributed event bus.

WildFly main branch uses Vertx 4.5.9, but Vertx 4.x branch is on JDK 8 and uses Infinispan 13.0.x branch, which does not align with the Infinispan version(14.0.30.Final) used in WildFly.

So I think this should be tossed out when integrating into WildFly before applicable solutions.

Deploy Vertx Verticle to WildFly (optional)

The extension also provides a way to deploy a [Vertx Verticle](#) to the Vertx instance inside WildFly. The basic idea is a deployment descriptor called `vertx.json` together with the deployment archive, which defines the deployment options, like:

```
Java
{
  "deployments": [
    {
      "verticle-class":
"org.wildfly.extension.vertx.test.basic.deployment.TestVerticle",
      "deploy-options": {}
    }
  ]
}
```

The Verticle will be undeployed when the deployment gets undeployed.

This is optional to integrate to WildFly because it is out of the scope of the original issue above.

Get the Vertx instance from other WildFly subsystems

Once integrated into WildFly, tries to use the following way to get the Vertx instance:

```
Java
Supplier<VertxProxy> vertxProxySupplier =
sb.requires(VertxResourceDefinition.VERTEX_RUNTIME_CAPABILITY.getCapabilityServiceName());
Vertx vertx = vertxProxySupplier.get().getVertx();
```

Options to integrate to WildFly

After we decide which capabilities will be integrated into WildFly and apply the associated changes to the extension, it will be ready to get integrated.

I think there are 2 options to do the integration:

Option 1: Directly into WildFly main repository

As Vertx is a dependency of WildFly, it makes sense to have a Vertx subsystem directly in WildFly.

The benefits of this option are:

- The reactive messaging subsystem won't depend on an external artifact to get the Vertx instance
- Only copying the necessary capabilities codes into WildFly, which makes code clean.
- Keep the Vertx version alignment in the WildFly code repository.

Option 2: As a standalone extension in wildfly-extras

Keep the extension as a standalone project and move it to wildfly-extras, integrate it as an extra feature pack.

The benefits of this option are:

- Keep the development separated from WildFly.

Decisions

It would be great if we can decide on which feature should be integrated into WildFly, which should not, and in which option we should take to get the integration done.

Feature	Included	Not included
Configure VertxOptions	<input type="checkbox"/> Yes	<input type="checkbox"/> No
Clustered Vertx instance	<input type="checkbox"/> Yes	<input type="checkbox"/> No
Deploy Vertx Verticle	<input type="checkbox"/> Yes	<input type="checkbox"/> No

Integration Options	<input type="checkbox"/> Option 1	<input type="checkbox"/> Option 2
----------------------------	-----------------------------------	-----------------------------------

Votes from Kabir Khan are: +1 for Configure VertxOptions, -1 for others. +1 for Option 1