

# Prototyping WildFly Applications with JBang

# About Me



- ▶ Senior Principal Software Engineer @ Red Hat
- ▶ Joined the WildFly team in 2012
- ▶ Worked on Messaging & Core capabilities of WildFly
- ▶ Working on its Cloud enablement and User Experience
- ▶ <https://jmesnil.net/>

# Agenda



- ▶ Introduction to **JBang**
- ▶ Write a simple Web application running with **WildFly**
- ▶ WildFly **Glow**
- ▶ Write an AI Chat bot
- ▶ Q&A

A stylized landscape illustration featuring a dark green background. In the foreground, there are several overlapping, angular mountain shapes in shades of blue and grey. On the left side, a multi-colored rainbow (red, orange, yellow, red) is partially visible, appearing to rise behind the mountains.

JBang  
<https://jbang.dev>

JBang lets Students, Educators and Professional Developers create, edit and run self-contained source-only Java programs with unprecedented ease.

# JBang



## [Download JBang](#) (Java not required)

```
$ curl -Ls https://sh.jbang.dev | bash -s - app setup
```

```
$ jbang --help
```

```
jbang is a tool for building and running .java/.jsh scripts and jar packages.
```

```
Usage: jbang [-hV] [--insecure] [--preview] [--config=<config>] [--verbose |  
            --quiet] [-o | --fresh] [COMMAND]
```

```
[...]
```

# JBang



```
#!/usr/bin/env jbang "$@" ; exit $?
```

```
//DEPS com.github.lalyos:jfiglet:0.0.8
```

<https://gist.github.com/jmesnil/256b3fe1c07cc3ba1942034057f76b47>

```
import com.github.lalyos.jfiglet.FigletFont;

class myapp {
    public static void main(String[] args) throws Exception {
        String name = args.length==0 ? "World" : args[0];
        String hello = String.format("Hello, %s!", name);
        System.out.println(FigletFont.convertOneLine(hello));
    }
}
```



Demo









# JBang



- [Download JBang](#) (Java not required)
- [Documentation](#)
- [IDE Support](#)



WildFly  
<https://wildfly.org>

A powerful, modular, & lightweight application server that helps you build amazing applications.



# Web Application



```
#!/usr/bin/env jbang "$0" "$@" ; exit $?
```

```
//DEPS com.github.lalyos:jfiglet:0.0.8
```

```
//DEPS org.wildfly.bom:wildfly-expansion:35.0.1.Final@pom
```

```
//DEPS jakarta.ws.rs:jakarta.ws.rs-api
```

```
//DEPS jakarta.enterprise:jakarta.enterprise.cdi-api
```

```
//DEPS org.wildfly.glow:wildfly-glow:1.3.2.Final
```

```
import jakarta.enterprise.context.ApplicationScoped;
```

```
import jakarta.ws.rs.*;
```

```
import jakarta.ws.rs.core.*;
```

```
import com.github.lalyos.jfiglet.FigletFont;
```

```
@ApplicationPath("/")
```

```
public class myapp extends Application {
```

```
    @Path("/hello")
```

```
    @ApplicationScoped
```

```
    public static class Hello {
```

```
        @GET
```

```
        public String sayHello(@QueryParam("name") @DefaultValue("World") String name) throws Exception {
```

```
            String hello = String.format("Hello, %s!", name);
```

```
            return FigletFont.convertOneLine(hello);
```

```
        }
```

```
    }
```

```
}
```

<https://gist.github.com/jmesnil/5f86e521068f6921f80ebb142c7f4704>

Demo



# Web Application



```
$ jbang run myapp.java
[jbang] Resolving dependencies...
[jbang]   org.wildfly.glow:wildfly-glow:1.3.2.Final
[jbang]   com.github.lalyos:jfiglet:0.0.8
[jbang]   org.wildfly.bom:wildfly-expansion:35.0.1.Final@pom
[jbang]   jakarta.ws.rs:jakarta.ws.rs-api:3.1.0
[jbang]   jakarta.enterprise:jakarta.enterprise.cdi-api:4.0.1
[jbang] Dependencies resolved
[jbang] Building jar for myapp.java...
[jbang] Post build with org.wildfly.glow.jbang.JBangIntegration
...
17:02:39,613 INFO  [org.jboss.as.server] (ServerService Thread Pool -- 20) WFLYSRV0010Deployed "myapp.war"
(runtime-name : "myapp.war")
...
17:02:39,632 INFO  [org.jboss.as] (Controller Boot Thread) WFLYSRV0025WildFly 35.0.1.Final (WildFly Core
27.0.1.Final) started in 2225ms - Started 192 of 196 services (35 services are lazy, passive or on-demand) -
Server configuration file in use: standalone.xml - Minimum feature stability level: community
```



# Web Application



```
$ curl "http://localhost:8080/myapp/hello"
```

```
  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
 |  |  /  \  |  |  |  /  \  \  \  /  \  /  \  /  \  /  \  /  \  /  \  /  \
 |  |  /  \  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
 |  |  \  /  |  |  |  |  \  /  ( )  |  |  \  /  \  \  \  \  \  \  \  \  \
 |  |  \  /  |  |  |  |  \  /  ( )  \  \  /  \  /  \  /  \  /  \  /  \  /  \
```

```
$ curl "http://localhost:8080/myapp/hello?name=Alice"
```

```
  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _  _
 |  |  /  \  |  |  |  /  \  \  \  /  \  /  \  /  \  /  \  /  \  /  \  /  \
 |  |  /  \  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
 |  |  \  /  |  |  |  |  \  /  ( )  |  |  \  /  \  \  \  \  \  \  \  \  \
 |  |  \  /  |  |  |  |  \  /  ( )  \  \  /  \  /  \  /  \  /  \  /  \  /  \
```

# MicroProfile Application



```
#!/usr/bin/env jbang "$0" "$@" ; exit $?
//DEPS com.github.lalyos:jfiglet:0.0.8
//DEPS org.wildfly.bom:wildfly-expansion:35.0.1.Final@pom
//DEPS jakarta.ws.rs:jakarta.ws.rs-api
//DEPS jakarta.enterprise:jakarta.enterprise.cdi-api
//DEPS org.eclipse.microprofile.config:microprofile-config-api
//DEPS org.wildfly.glow:wildfly-glow:1.3.2.Final

import jakarta.enterprise.context.ApplicationScoped ;
import jakarta.ws.rs.*;
import jakarta.ws.rs.core.*;
import com.github.lalyos.jfiglet.FigletFont ;
import jakarta.inject.Inject ;
import org.eclipse.microprofile.config.inject.ConfigProperty ;

@ApplicationPath ("/")
public class myapp extends Application {
    @Path ("/hello")
    @ApplicationScoped
    public static class Hello {

        @Inject
        @ConfigProperty (name = "hello", defaultValue = "Hello")
        String helloWord ;

        @GET
        public String sayHello (@QueryParam ("name") @DefaultValue ("World") String name) throws Exception {
            String hello = String.format ("%s, %s!", helloWord, name);
            return FigletFont .convertOneLine (hello);
        }
    }
}
```

<https://gist.github.com/jmesnil/b0f05ca66648824a1ebb52cb4b7f7de4>

Demo



# MicroProfile Application



```
$ HELLO="Bonjour" jbang run myapp.java

[jbang] Resolving dependencies...
...
[jbang]   jakarta.enterprise:jakarta.enterprise.cdi-api:4.0.1
[jbang]   org.eclipse.microprofile.config:microprofile-config-api:3.1
[jbang] Dependencies resolved
[jbang] Building jar for myapp.java...
[jbang] Post build with org.wildfly.glow.jbang.JBangIntegration
...
17:11:18,658 INFO  [org.wildfly.extension.microprofile.config.smallrye] (ServerService Thread Pool -- 27)
WFLYCONF0001: Activating MicroProfile Config Subsystem
...
17:11:19,525 INFO  [org.jboss.as] (Controller Boot Thread) WFLYSRV0025: WildFly 35.0.1.Final (WildFly Core
27.0.1.Final) started in 2326ms - Started 192 of 196 services (35 services are lazy, passive or on-demand) -
Server configuration file in use: standalone.xml - Minimum feature stability level: community
```



# WildFly Glow



WildFly Glow is an evolution of the WildFly [Galleon](#) provisioning tooling.

It provides tooling allowing to identify the set of Galleon Feature-packs and Layers that are required by your application. The entry-point of this discovery is your application.

- [WildFly Glow Documentation](#)
-  [Introduction to WildFly Glow](#)
-  [WildFly Glow Lightning Talk @ WildFly Mini Conference Nov 2024](#)



# Glow Arguments



```
//GLOW --server-version=34.0.0.Final
```

👉 Specify the version of WildFly to run the application

```
//GLOW --add-ons=health,metrics,openapi
```

👉 Specify the add-ons to provision in WildFly that can not be inferred by your code

```
//GLOW --spaces=incubating
```

👉 Provision additional WildFly capabilities that are in the incubating space (more on that later)

# Glow Arguments



```
//GLOW --add-ons=health,metrics,openapi
```

```
$ curl "http://localhost:9990/metrics"
```

```
$ curl "http://localhost:9990/health"
```

```
$ curl "http://localhost:8080/openapi"
```

Demo



# AI Chat bot



Let's write an AI chat bot that explains to kids the etymology of a given word.

We will use an [Ollama](#) chat model and [LangChain4j](#) to write the Java code.

WildFly provides an [AI Feature Pack](#) that provides the integration with LangChain4j.

# AI Chat bot



<https://gist.github.com/jmesnil/4e4b7e44ec8174a79c85fe0c1c18e0d7>

```
#!/usr/bin/env jbang "$@" -- "$@" ; exit $?
//DEPS org.wildfly.bom:wildfly-expansion:35.0.1.Final@pom
//DEPS org.wildfly.glow:wildfly-glow:1.3.2.Final
//DEPS jakarta.ws.rs:jakarta.ws.rs-api
//DEPS jakarta.enterprise:jakarta.enterprise.cdi-api
//DEPS dev.langchain4j:langchain4j-bom:1.0.0-beta2@pom
//DEPS dev.langchain4j:langchain4j
//GLOW --spaces=incubating

import dev.langchain4j.data.message.*;
import dev.langchain4j.model.chat.ChatLanguageModel;
import jakarta.enterprise.context.RequestScoped;
import jakarta.inject.*;
import jakarta.ws.rs.*;
import jakarta.ws.rs.core.*;

@ApplicationPath("/")
public class myaiapp extends Application {
    @Path("/chat")
    @RequestScoped
    public static class Chat {
        @Inject
        @Named(value = "ollama")
        ChatLanguageModel chatModel;

        @GET
        @Produces(MediaType.TEXT_HTML)
        public String chatWithAssistant(@DefaultValue("Orange") @QueryParam("word") String word) throws Exception {
            return chatModel.chat(
                SystemMessage
                    .from("""
                        You are a teacher that explain to kids the origin of some words.
                        Your response must be polite, use the same language as the question, and be relevant to the question.
                        Your answer must be embedded in HTML.
                        """),
                UserMessage.from(String.format("What is the etymology of %s?" , word)))
                .aiMessage().text();
        }
    }
}
```

# Ollama Prerequisites



```
$ podman run -d --rm --name ollama -p 11434:11434 -v ollama:/root/.ollama ollama/ollama
$ podman exec -it ollama ollama pull llama3.2:1b

$ OLLAMA_CHAT_MODEL_NAME=llama3.2:1b ./myaiapp.java

$ open http://localhost:8080/myaiapp/chat?word=Orange
```



Demo



# References



- ▶ [Running WildFly with JBang](#)
- ▶ [JBang](#)
- ▶ [WildFly](#)

Q & A





Thanks!

Jeff Mesnil

[jmesnil@redhat.com](mailto:jmesnil@redhat.com)

<https://jmesnil.net/>