



***From the
announcement of
Keycloak.X
to a **Quarkus** based future***



*How the transition went **down**.*

Keycloak

- *Short overview of the project*
- *The announcement of Keycloak.X*

How it went down

- *Digging into releases*
- *Architectural transformation*
- *Everything is under constant change*

Learning from a big project

- *Did it work out?*
- *What can one take away from this?*



\$ whois *Kilian*



Software Architecture

- insert some buzzwords -



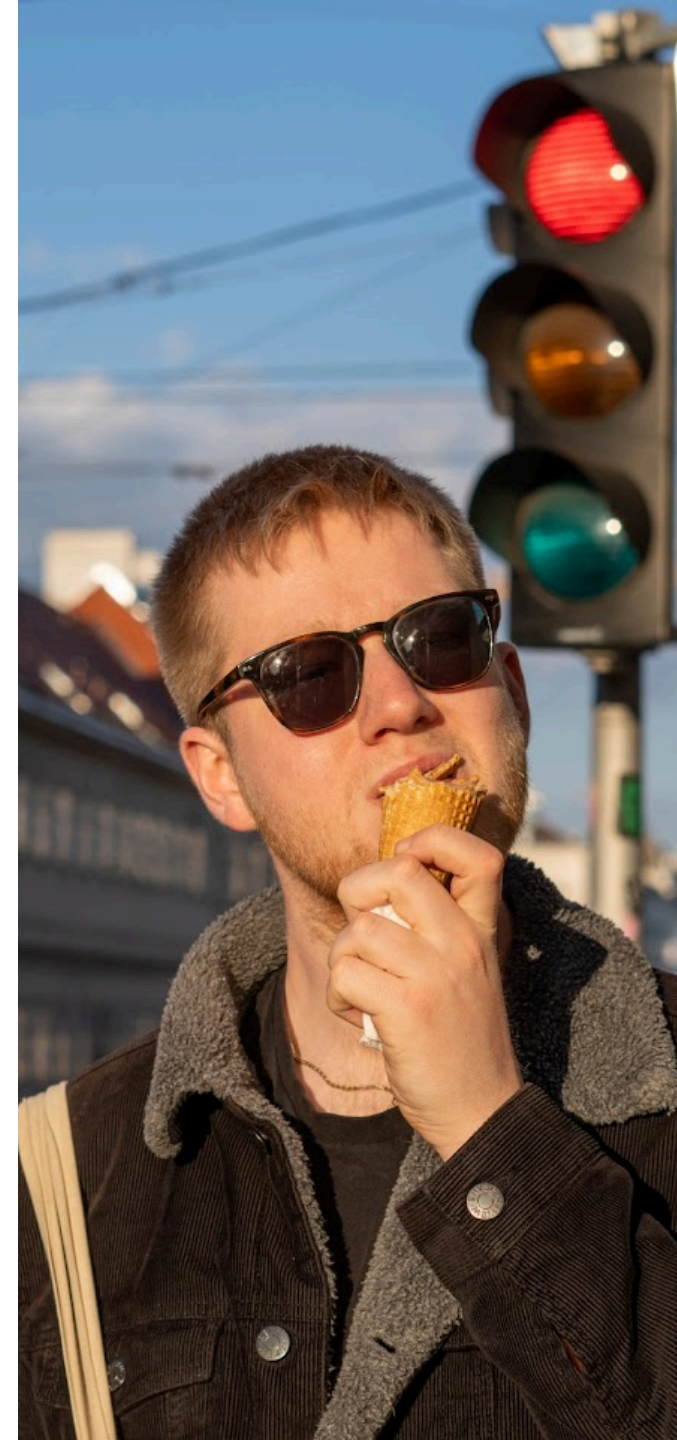
DevOps

- even more buzzwords -



Ice-cream

apparently a huge thing in my life





*How the transition went **down**.*

Keycloak

- *Short overview of the project*
- *The announcement of Keycloak.X*

How it went down

- *Digging into releases*
- *Architectural transformation*
- *Everything is under constant change*

Learning from a big project

- *Did it work out?*
- *What can one take away from this?*





The gist.

IAM

Identity and access management

*OpenID Connect, OAuth 2,
SAML 2*

User Federation

Flexible options

*LDAP, Active Directory,
different RDBMS*

Built for extensibility

Service Provider Interfaces

*Extendable with custom JARs
or HTML and JS.*



Before Keycloak.X.

Wildfly based

Built for standalone deployment

There is a docker image, but it is a wildfly server with the installation.

Configuration

XML/properties based


Leveraging the Wildfly ecosystem for configuration.

Hot swapping extensions


Jars and JS modules

Integrating self built modules via Java-Script upload or as JARs with XML config.

Introduction of *Keycloak.X*

- 
- 1. Make it easier to configure*
 - 2. Make it easier to scale, including multi-site support*
 - 3. Make it easier to extend*
 - 4. Reduce startup time and memory footprint*
 - 5. Support zero-downtime upgrades*
 - 6. Support continuous delivery*

*Introduction of **Keycloak.X***

- 
- 1. A new and improved storage layer*
 - 2. A new distribution powered by **Quarkus***
 - 3. A new approach to custom providers*



*How the transition went **down**.*

Keycloak

- *Short overview of the project*
- *The announcement of Keycloak.X*

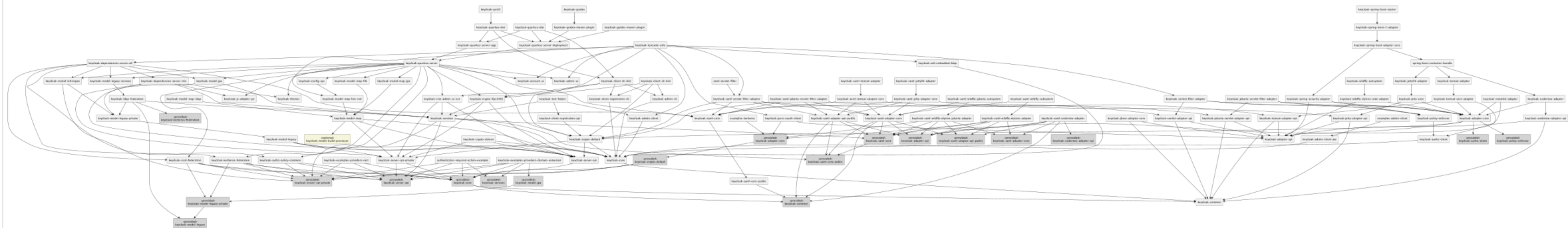
How it went down

- *Digging into releases*
- *Architectural transformation*
- *Everything is under constant change*

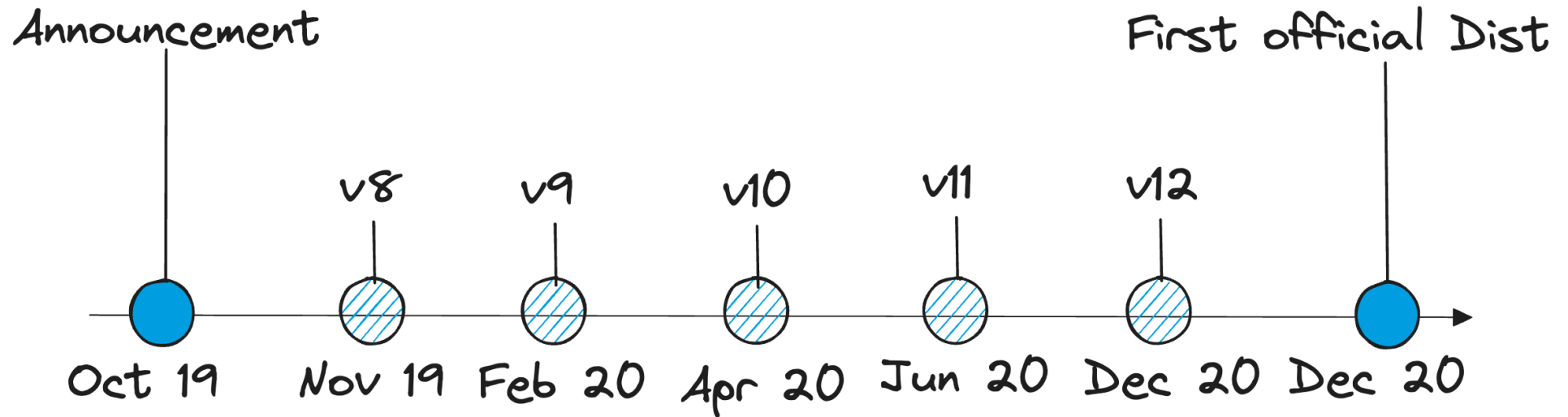
Learning from a big project

- *Did it work out?*
- *What can one take away from this?*





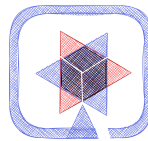
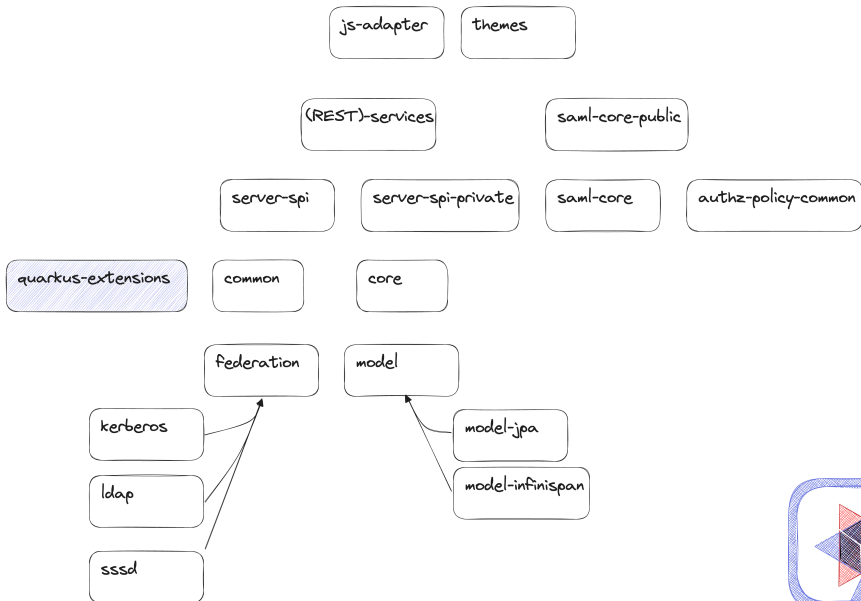
"The aim is to have a fully functional stable distribution by the end of 2019"



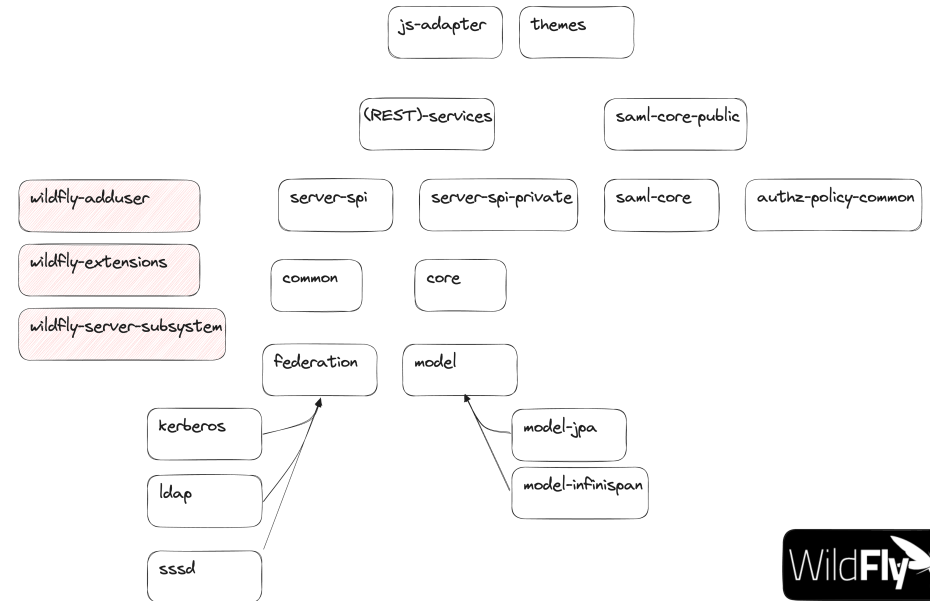
It's so *simple*.



keycloak quarkus modules



keycloak wildfly modules

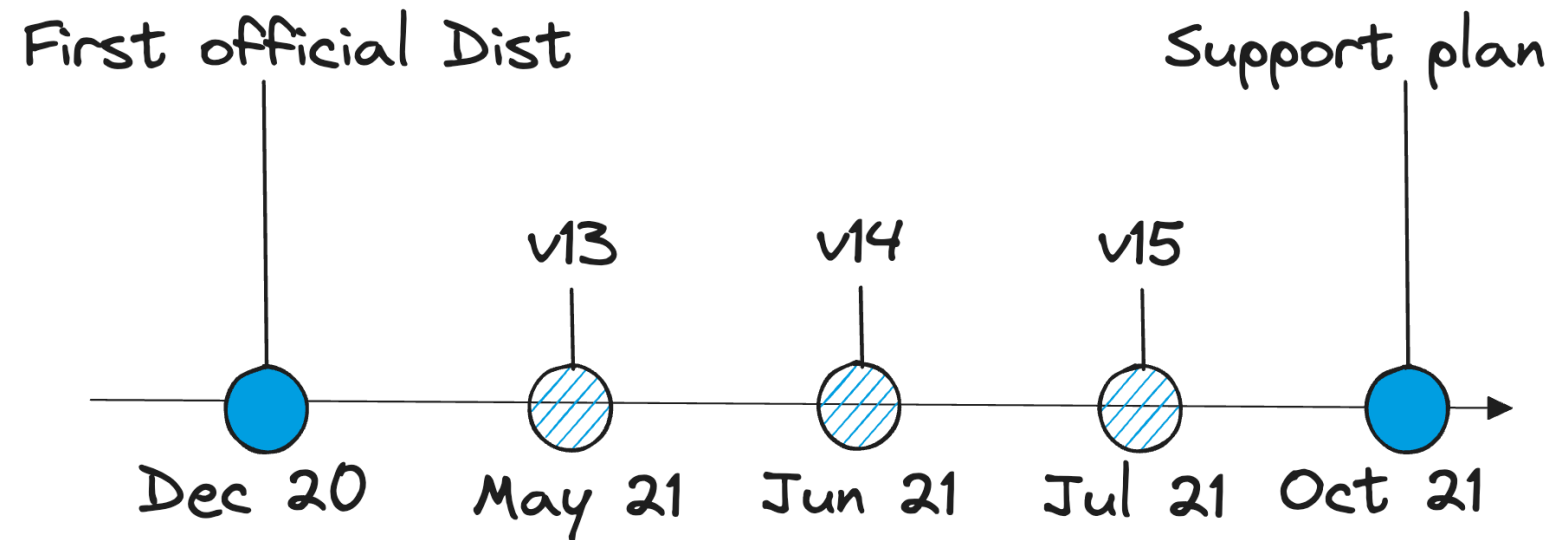


Simplicity **wins.**

```
<profile>
  <id>quarkus</id>
  <modules>
    <module>quarkus</module>
  </modules>
</profile>
```



Distribution	Startup Time (s)	Memory Footprint (RSS/MB)
Wildfly	First 12.1 Second 8.1	First 646 Second 512
Quarkus	First 7.6 Second 3.1	First 428 Second 320



New deployment *strategy*.

Static Configuration

Example:

Database Vendor e.g. postgres

Provide the possibility for runtime optimisations.

*Options are picked up at **build time** with:*
`kc.sh builder --db=postgres`

Dynamic Configuration

Example:

Database connection string.

Allows for deployment of the same image to different environments.

*Options are picked up on **startup** with:*
`ks.sh -db-password=v3ryS3cur3!`



Providers stay the **same**.

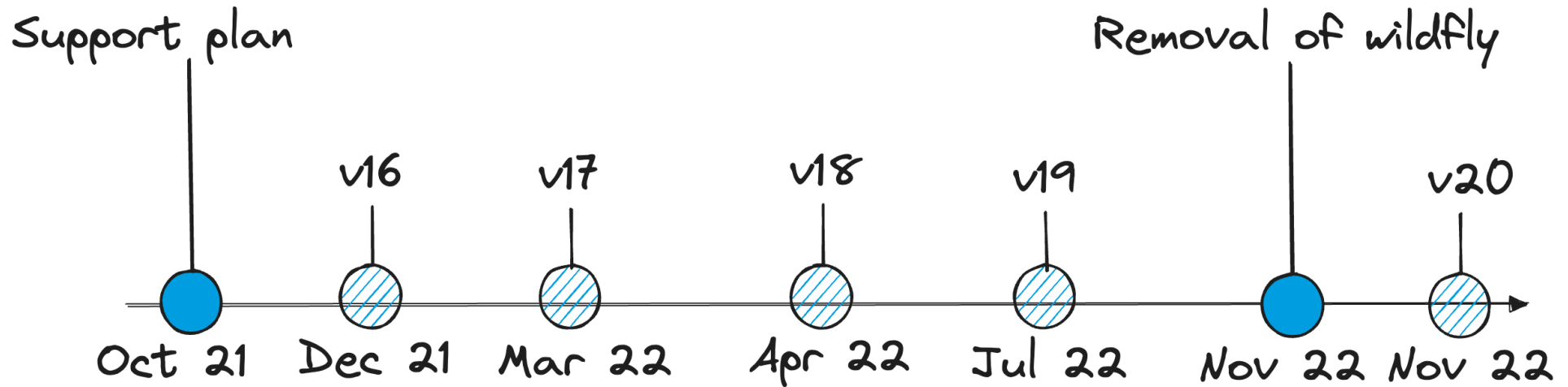
- *Provided as jar files*
- *Included automatically*
 - *When running the new build command*
- *Still improves startup*
 - *No load on startup*



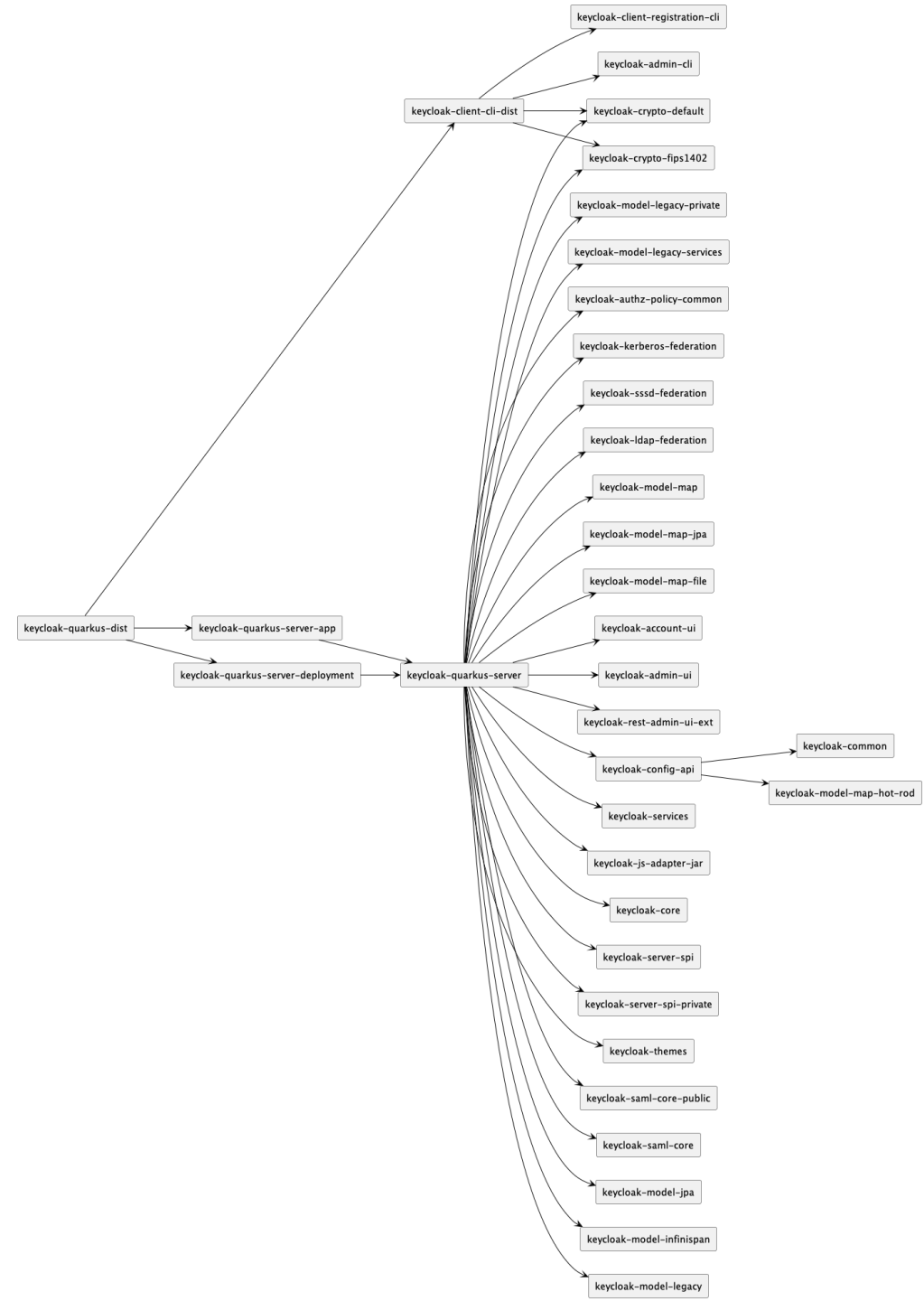
"In Keycloak 17 we will make the Quarkus distribution fully supported, and deprecate the WildFly distribution."

On v18:

"We're also planning on removing WildFly support from the code-base at this point."



Relevant architecture.



*The actual **modules**.*

- *Deployment*
 - *Build-time code*
- *Runtime*
 - *Runtime code*
- *Server*
 - *Module to generate artifacts*





*How the transition went **down**.*

Keycloak

- › *Short overview of the project*
- › *The announcement of Keycloak.X*

How it went down

- › *Digging into releases*
- › *Architectural transformation*
- › *Everything is under constant change*

Learning from a big project

- › *Did it work out?*
- › *What can one take away from this?*



*Did this really **Work**?*

~~It~~ depends.



*What can we **learn**?*

Taking time is key.

Prototyping the solution.

Parallel releases are not extra effort.

Support windows should be longer than you think.





Thanks! Any questions?

Sources:

<https://www.keycloak.org/2019/09/2019-roadmap.html>

<https://www.keycloak.org/2019/10/keycloak-x>

<https://www.keycloak.org/2020/12/first-keycloak-x-release.adoc>

<https://www.keycloak.org/2021/10/keycloak-x-update.html>

<https://www.keycloak.org/2022/03/releases.html>

<https://www.keycloak.org/2023/07/keycloak-2200-released.html>

<https://www.keycloak.org/migration/migrating-to-quarkus>

<https://github.com/keycloak/keycloak-community/blob/main/design/keycloak.x/README.md>

<https://github.com/keycloak/keycloak.git>



KILIAN SCHRÖDER

Solution engineer

kilian.schroeder@open200.com