



UNIVERSITY OF TARTU

Serverless

04.06.2020

Sander Kuusemets



Sisukord

- Sissejuhatus serverlessi
- Arhitektuur
- Positiivsed küljed
- OpenFaaS
- Kubeless
- Minu arvamus + negatiivsed küljed

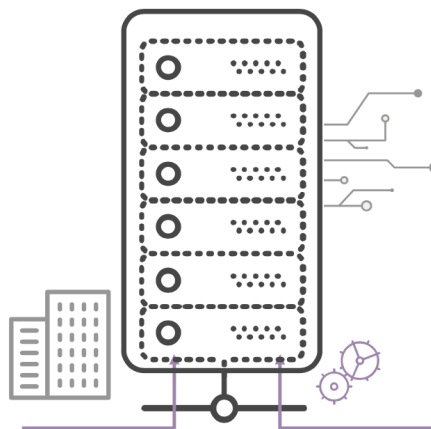
DevOps ajalugu

Kuni mitte nii pikk aeg tagasi:

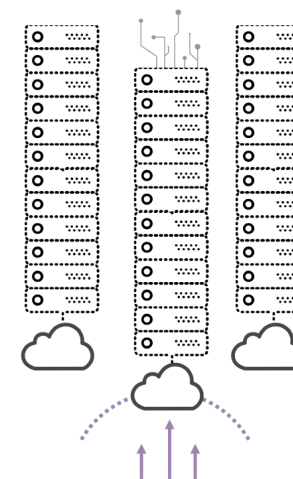
Füüsilised serverid:



Virtuaalsed serverid:



Virtuaalsed
pilveserverid:

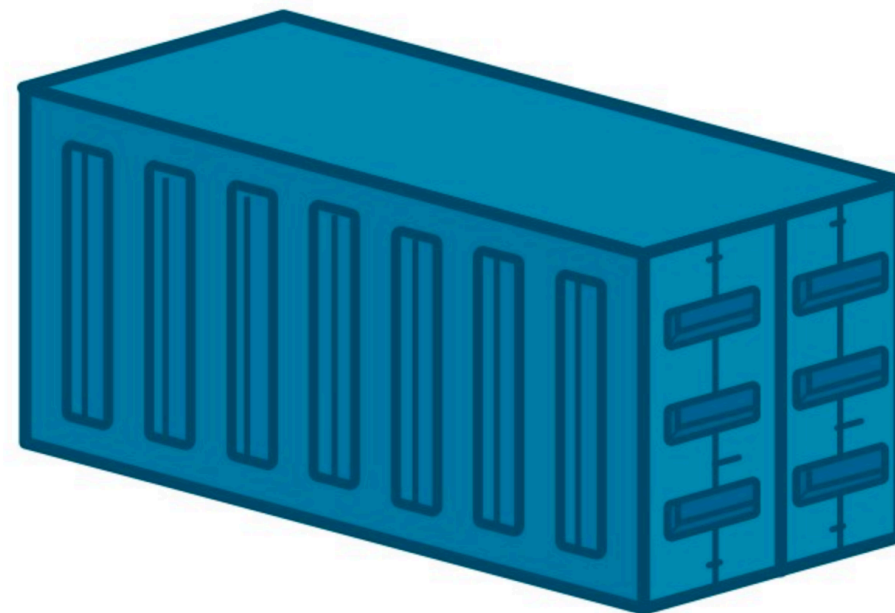


- + Iga samm oli parem, saavutades parema tiheduse ja võimekuse.
- + Virtuaalmasinad lubasid katastroofidest taastumise.
- + Asjad võtsid suuna riistvara poolsele iseseisvusele.

- Virtualisatsioonil on kõrge protsessimise lisakulu.
- Migreerimine, restartimine ja taastamine võttis aega.
- Aga mitte operatsioonisüsteemi poolsele iseseisvusele.

Siis tulid konteinerid..

- + Mitmed positiivsed küljed, nagu muutmatu infrastruktuur ja palju väiksem lisakulu (overhead).
- Manageerimise tööriistad polnud veel “üles kasvanud”, konteinereid hallati ja ehitati nagu VM-e.
- Pidi muretsema baasimage’ite ja pakkide eest, tutvustades keskkonda jälle lisakulu ja turvaprobleeme.



Järgmine loogiline samm.. Jätame konteinerid vahele!

- .. noh, mitte päris.
- Serverless ikka pakendab funktsioonid, mis on baasühik serverlessis, konteinerite sisse.
- Arendaja lihtsalt ei pea muretsema nende pakendamise või ehitamise eest, tööriistad teevad seda nende eest.

What is serverless?

Build and run applications without thinking about servers

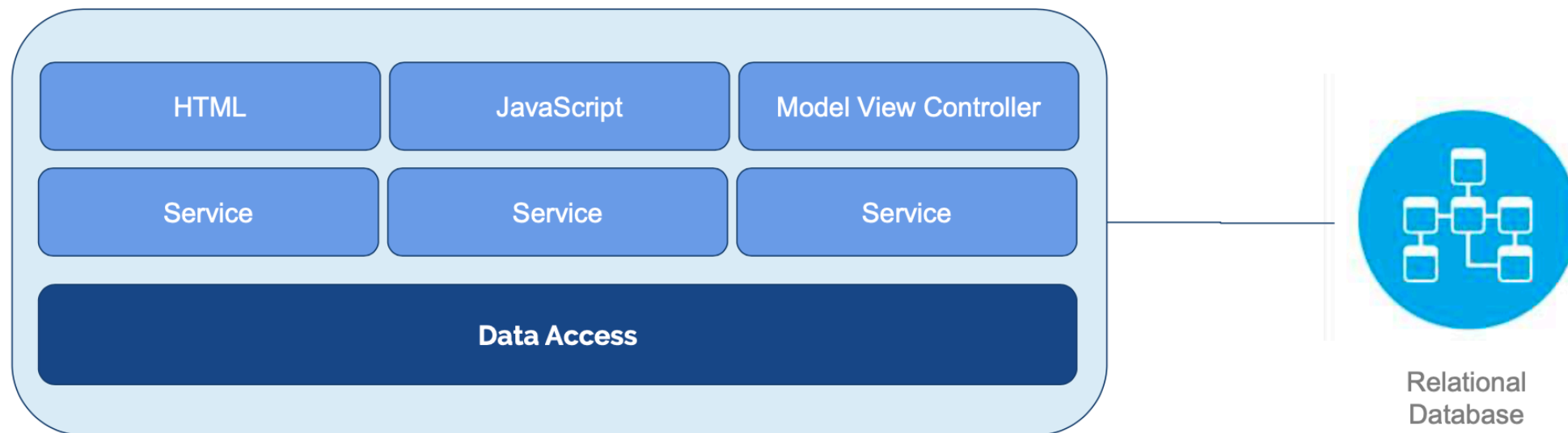




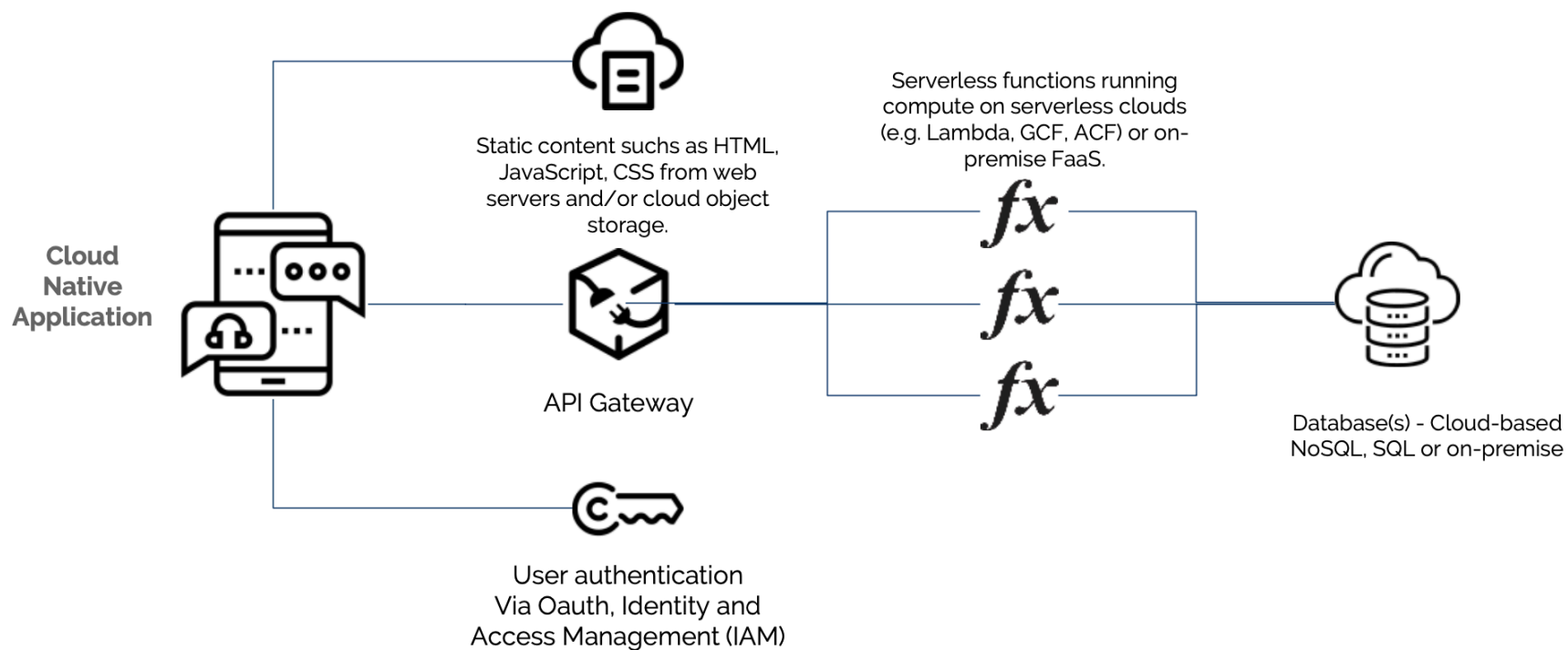
Mis siis on serverless?

- Kasutab funktsioone konteinerite asemel. Kirjutad funktsiooni, paigaldad selle pilve ja kutsud pilves välja.
- Pakub abstraktsiooni riistvarale, nii et arendajad ei pea sellele mõtlema.
- Sunnib peale olekuvaba (stateless), pilvepõhist mikroteenuste paradigmat.
- Sündmuste põhine.
- Suudab skaleeruda nii tuhandete, kui ka nulli replikani.

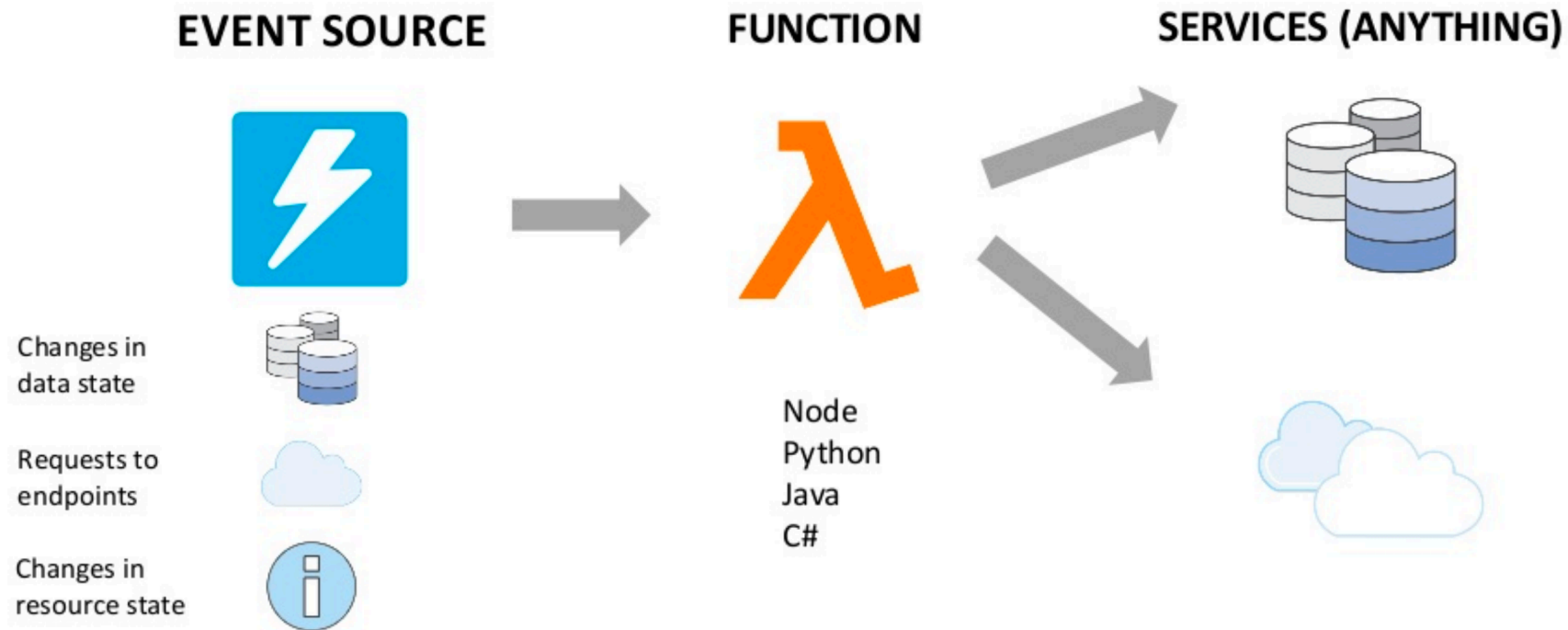
Monoliitne arhitektuur



Pilvepõhine arhitektuur

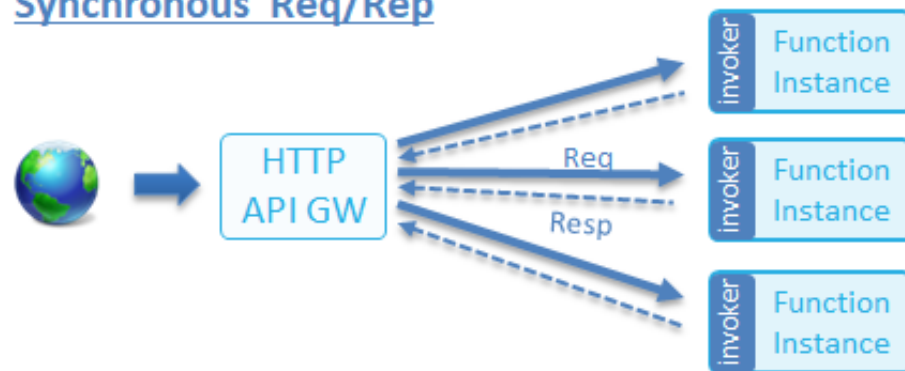


Serverless arhitektuur

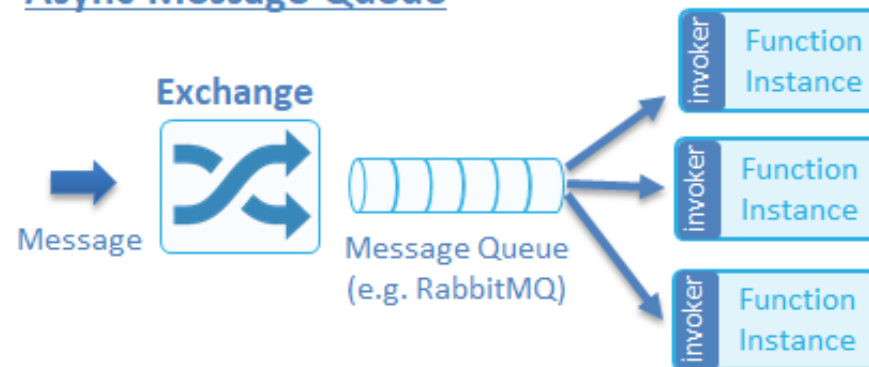


Serverless funktsiooni kutsumiste töövoog

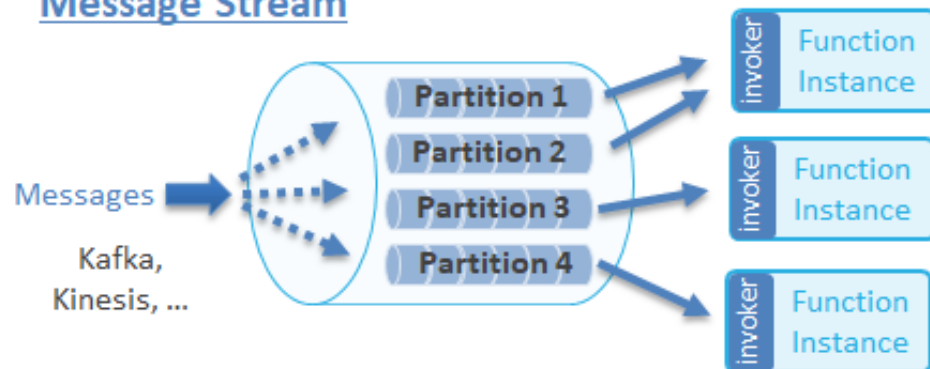
Synchronous Req/Rep



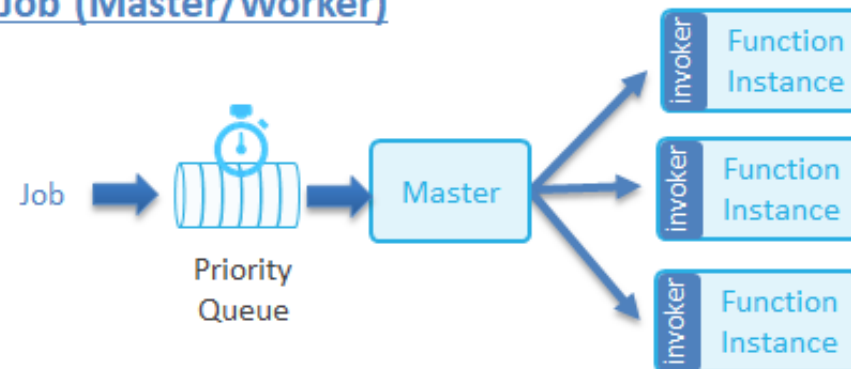
Async Message Queue



Message Stream



Job (Master/Worker)

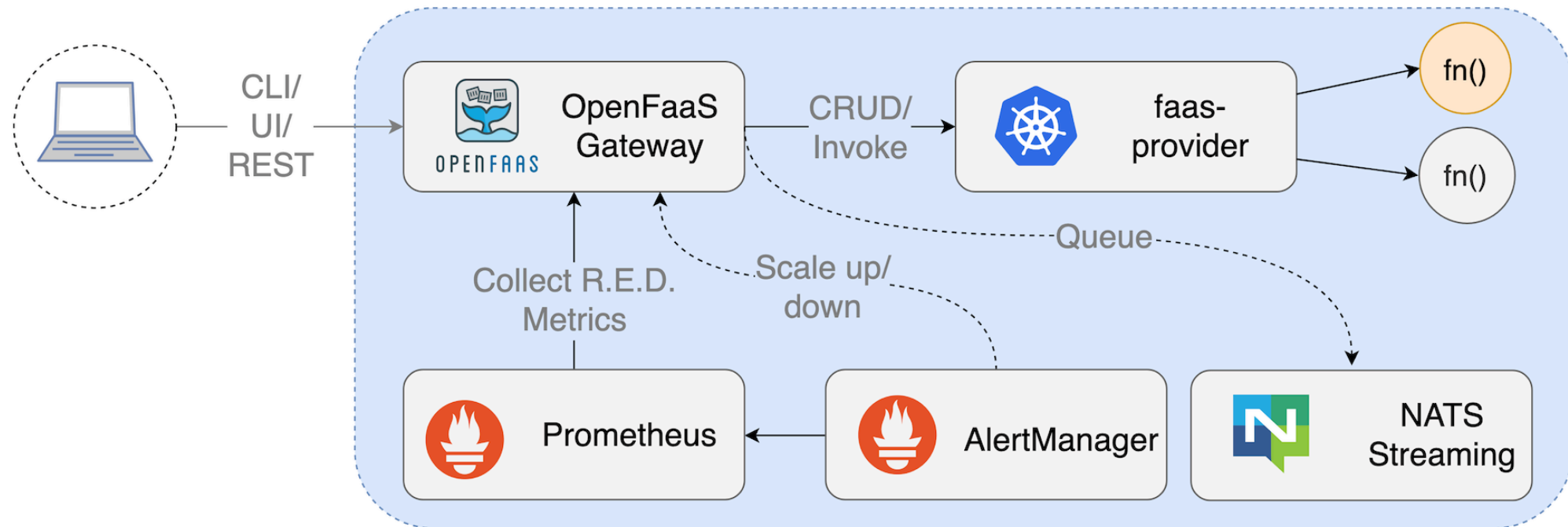




UNIVERSITY OF TARTU

Küsimusi?

OpenFaaS (Open source Function-as-a-Service)





OpenFaaS müügiutt (nende kodulehelt)

- Lihtne UI portaal ja ühe-clicki-install.
- Kirjuta funktsioone igas keeles Linux või Windowsi peal, ja paki Docker/OCI image formaati.
- Kaasaskantav – jookseb olemasoleva riistvara kui ka avaliku/privaatse pilve seljas. Nii Kuberneteses kui ka Docker Swarmis.
- Käsureatööriist YAML formaadis templaatimise ja funktsioonide defineerimisega.
- Automaatne skaleerumine vastavalt kasutusele.

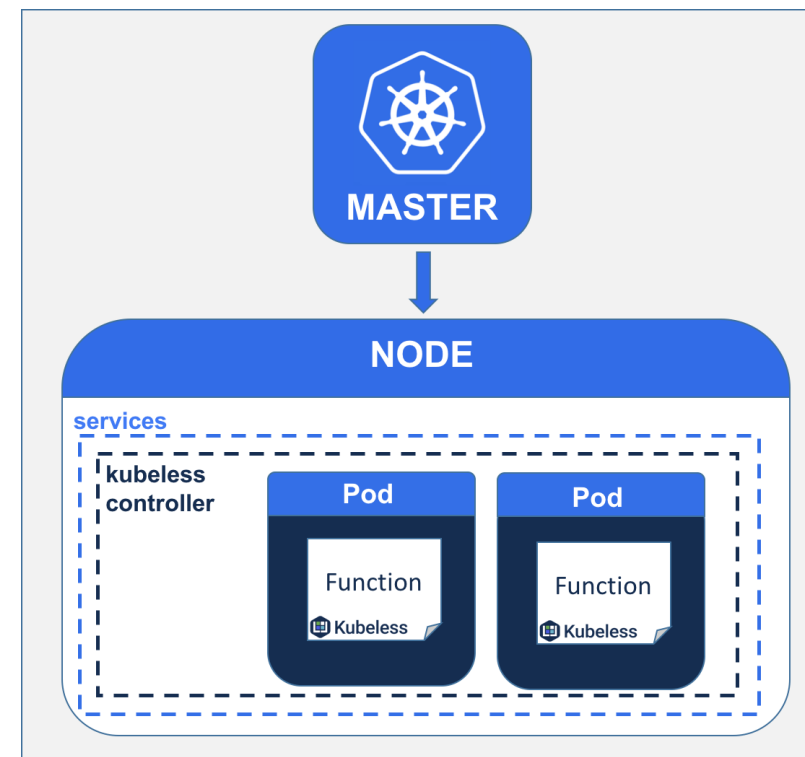


Demo

- Placeholder

Kubeless

- Teeb asju veidi lihtsamalt.
- Ehitatud algusest peale integreeruma Kubernetesega.
- Järgib AWS, GCS, Azure jms parimaid praktikaid.
- Omab palju vähem komponente.
- Võibolla mõnest poolest veel toores.



Minu arvamus serverless-ist ja selle negatiivsed küljed

- Teeb tegevuse väga otsekoheks arendajatele.
- .. mitte aga DevOps/SysOps/SRE inimeste jaoks.
- Ikka vaja hallata paigaldusi, kuid nüüd VM-ide või konteinerite asemel funktsioone.
- Nõuab väga-väga-väga head monitooringut ja teavituste infrastruktuuri.
- Lubab kasutada arenduse parimaid praktikaid paigaldamise jaoks – testimine, rolling releases, versioneerimine, katki läinud paigalduse tagasi kutsumine jms.



UNIVERSITY OF TARTU