STS, KMD I2 og STIL

Integrationskomponent

**Version**: 1.1.0

**Date**: 18.02.2021

**Author**: BSG

Indhold

[1 Indledning 3](#_Toc64534297)

[2 Løsningsarkitektur 3](#_Toc64534298)

[2.1 Scope for leverancen 4](#_Toc64534299)

[2.2 Skeduleret dataudveksling 5](#_Toc64534300)

[2.3 KMD I2 integrationen 5](#_Toc64534301)

[2.4 STS Organisation integrationen 5](#_Toc64534302)

[2.5 AD integrationen (rolle-register) 5](#_Toc64534303)

[2.6 OS2rollekatalog integration (rolle-register) 6](#_Toc64534304)

[3 Ikke-funktionelle krav 6](#_Toc64534305)

[3.1 Anvendte teknologier 6](#_Toc64534306)

[3.2 Licens 6](#_Toc64534307)

[3.3 Deployment 6](#_Toc64534308)

[3.4 Kildekode placering 6](#_Toc64534309)

# Indledning

Nærværende dokument er en løsningsbeskrivelse af den leverede integration til KMD I2. Integrationskomponent kan levere data til KMD I2 snitfladen, der populerer KMD institution.

På baggrund af denne populering af data, sker der en opdatering af stamdata registre i STIL, der igen anvendes af forskellige fagsystemer, og slutmålet er at sikre at denne vedligehold af data sker ét sted og kun ét sted (i kommunens autoritative organisationsregister), og derefter replikerings rund i alle de systemer som måtte have interesse i at bruge disse data.

Ved at gøre brug af KMD I2 snitfladen, indfries dette mål.

Løsningsbeskrivelsen kan med fordel læses i samspil med kravspecifikationen.

* KMD I2 integrationskomponent.pdf

# Løsningsarkitektur

Den leverde løsning baserer sig på en containeriseret løsningsarkitektur, hvor hver enkelt opgave der skal løses, implementeres som en selvstændig komponent, der kan deployes i sin egen container.

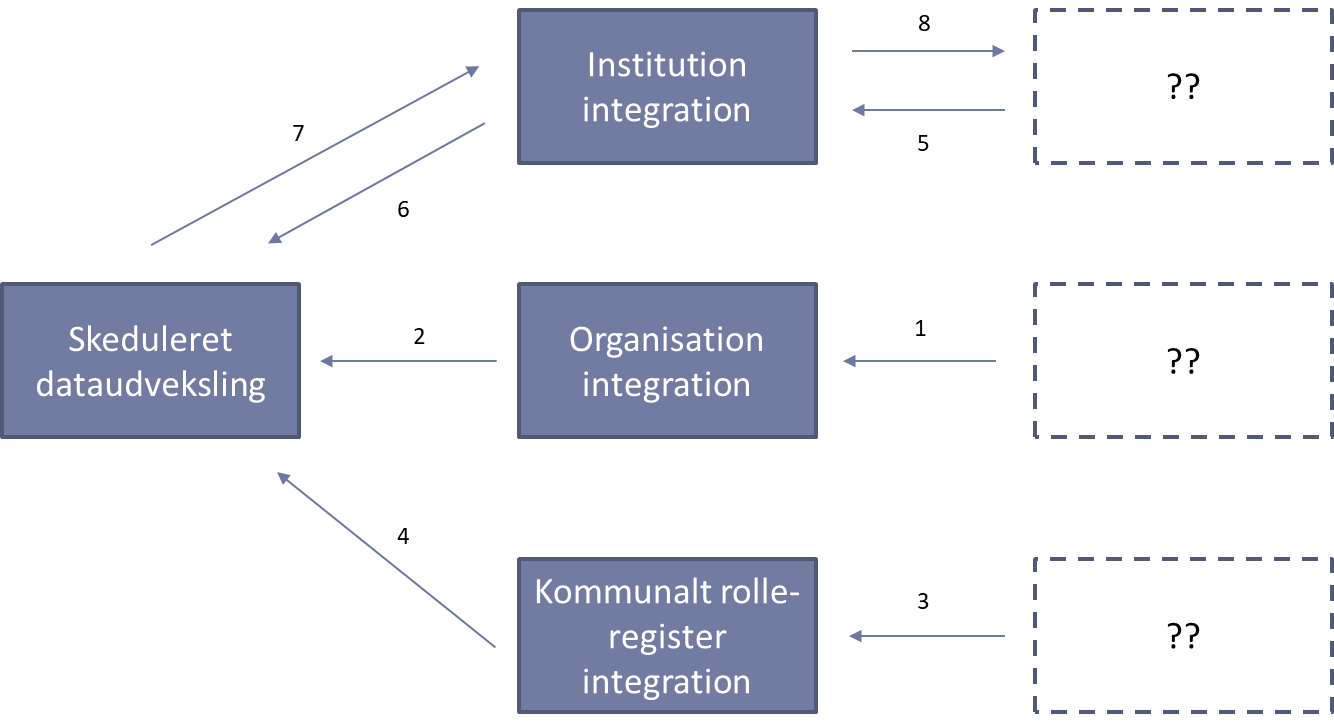
Konkret anvendes Docker som containerteknologi til den tilbudte SaaS drift, men der er ingen hårde bindinger til Docker, så hvis man ønsker at drifte løsningen selv, i en anden deployment teknologi, så er dette muligt.

Følgende opgaver løftes i hver sin container

* Integration til et institutionssystem (KMD I2 eller Tabulex)
* Integration til et organisationssystem (fx STS Organisation, OS2mo, ??)
* Integration til kommunalt rolle-register (fx AD, OS2rollekatalog, ??)
* Skeduleret dataudveksling

De 3 integrations-containere vil formodentligt kunne anvendes i anden kontekst, da de ikke holder nogen forretningslogik, men alene er designet til dataudveksling. Den skedulerede dataudvekslingscontainer holder forretningslogikken, og kan ikke genbruges i anden kontekst.

Den skedulerede dataudveksling er designet så den ikke behøves kende den faktisk kilde bag ved hver integrations-container, men overlader det valg til den konkrete integration.



Ovenstående illustration af container opdelingen, viser også flowet mellem de enkelte containere

1+2) Den skedulerede dataudvekslingscontainer beder om at få en fuld udlæsning af organisationsdata fra organisationsintegration-containeren, der henter data fra den kilde som containeren integrerer til

3+4) Den skedulerede dataudvekslingscontainer beder om at få en fuld udlæsning af rettighedstildelinger fra den kommunale rolleregisterintegration-container, der henter data fra den kilde som containeren integrerer til

5+6) Den skedulerede dataudvekslingscontainer beder om at få en fuld udlæsning af de data som pt eksisterer i institutionsløsningen, via institutionsintegrations-containeren, der henter data fra den kilde som containeren integrerer til

7+8) Den skedulerede dataudvekslingscontainer foretager den fornødne sammenligning af data, og sender relevante opdateringer til institutionsintegrations-containeren, der opdaterer den kilde som containeren integrerer til

## Scope for leverancen

Nærværende leverance reducerer scope for udviklingen til følgende integrations-containere (samt den skedulerede dataudvekslingscontainer), men den samlede løsning vil løbende kunne udvides med nye integrationer

* Integrations-container til KMD I2
* Integrations-container til STS Organisation
* Integrations-container til AD (rolle-register)
* Integrations-container til OS2rollekatalog (rolle-register)

Bemærk at der leveres 2 forskellige integrationer til rolle-registre. I et konkret drifts-scenarie skal blot én af disse anvendes.

## Skeduleret dataudveksling

Denne komponent udfører de kravsatte opdateringer, og afvikles skeduleret efter et CRON udtryk. Bemærk at der foretages en fuld synkronisering af data ved hver kørsel, og man derfor ikke bør opsætte skeduleringsintervallet til at afvikles oftere end integrationsparterne (KMD I2, STS Organisation, AD, osv) kan håndtere.

## KMD I2 integrationen

Denne integration udfører oprettelse, nedlæggelse og opdateringer af data i KMD I2 på baggrund af instrukser fra den skedulerede dataudveksling.

## STS Organisation integrationen

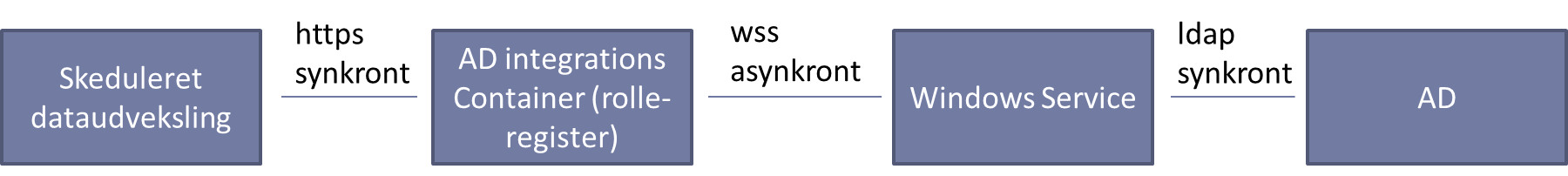
Denne integration baserer sig på OS2sync. Der er leveret både en vejledning i opsætning af OS2sync, samt en stubbet udgave af OS2sync der kan anvendes til testformål.

## AD integrationen (rolle-register)

Denne integration er leveret som 2 komponenter. Der er både en egentlig container komponent, som er integrationspunkt, samt en Windows Service, der installeres lokalt i kommunens windows server infrastruktur.

Den lokale komponent agerer dataudvekslingspunkt mellem containeren og AD’et.

Formålet med Windows Servicen er at undgå en direkte forbindelse fra integrations-containeren og til kommunens AD. Dette er typisk uønsket at kommunens driftsafdeling, og ved at selve integrationen til AD foregår via en Windows Service, kan kommunen overvåge integrationen, og styre rettigheder via hvad end metode som kommunen bruger til at styre Windows Services adgange til resourcer (herunder AD).



Forbindelsen mellem containeren og Windows Servicen etablers vha såkaldte sikre WebSockets (wss protokollen), hvilket undgår alle klassiske problemer med firewalls. Containeren kan derfor køre udenfor kommunens netværk, men stadig komme i forbindelse med AD’et via Windows Servicen, uden at der skal åbnes for nogen indgående netværks-trafik.

En forudsætning for anvendelsen af denne integration, er at brugerne i STS Organisation er opmærket med den AD konto de har i kommunens AD, så denne kan anvendes som opslagsnøgle.

Integrationen konfigureres med en liste af AD grupper, som afspejler tilhørende roller. Medlemsskab af disse AD grupper vil resultere i at integrationen outputter en rolletildeling på brugeren.

## OS2rollekatalog integration (rolle-register)

OS2rollekatalog udstiller et API til opslag på rettigheder på en bruger jf denne bruges UUID (det unikke ID man får fra STS Organisation).

Integration kan udtrække oplysninger om rettigheder via OS2rollekatalogs API, og udstille dem til den lokale container infrastruktur via samme snitflade som AD containeren.

# Ikke-funktionelle krav

## Anvendte teknologier

De enkelte containere er udviklet i hhv Java og .NET Core. Afhængig af den enkelte integration, anvendes det programmeringssprog (Java eller .NET Core) som er mest praktisk til integrationen.

## Licens

Kildekoden er tilgængelig som Open Source under MPL 2.0 licensen.

Der anvendes alene udbredte, vedligeholdte og tilgængelige Open Source frameworks, og der er ingen anvendelsen af licenseret software og/eller frameworks.

## Deployment

De enkelte komponenter i løsningen er udviklet som selvstændige services, der kan deployes vha container teknologi. Der medfølger dokumentation på deployment vha Docker som docker-compose filer.

Løsningen kan dermed deployes i ethvert afviklingsmiljø der kan håndtere containers, herunder forskellige PaaS/IaaS miljøer som Azure og AWS.

## Kildekode placering

Kildekoden publiceres, sammen med byggescripts og dokumentation, i et offentligt tilgængelig GIT repository. Hvis Kunden ikke udpeger et sådan, publiceres de i leverandørens GIT repository, men med offentlige læseadgang.