



Bilag 12 - Fælles arkitekturramme for GD1-GD2-GD7

OIO Serviceprincipper

Version: 1.1

Status: i høring i PF for GD1 og GD2

Oprettet: 4. juni 2014

Dato: 4. juni 2014



Dokument historie

Version	Dato	Beskrivelse	Initialer
1.1	04-06-2014	Kopi af test fra hjemmeside http://arkitekturguiden.digitaliser.dk/serviceprincipper pr. 04.06.2014	DGN

Indholdsfortegnelse

Indhold

OIO Serviceprincipper	1
INDLEDNING	3
BAGGRUND	3
KONTEKST FOR ANVENDELSE	4
1. SERVICES UNDERSTØTTER FORRETNINGEN	5
2. SERVICES KAN GENBRUGES	5
3. SERVICES KAN KOMBINERES	6
4. SERVICES KOBLER LØST	6
5. SERVICES INDKAPSLER DATA/FUNKTIONALITET	7
6. SERVICES KAN ANVENDES UAFHÆNGIGT AF PLATFORM	8
7. SERVICES KAN ANVENDES UAFHÆNGIGT AF LOKATION	9
8. SERVICES ER VELBESKREVNE	9
9. SERVICES PUBLICERES	10
10. SERVICES VERSIONERES	11
11. SERVICES ER BASERET PÅ STANDARDER	12
12. SERVICES RETURNERER SIGENDE FEJLMEDDELSER	12



Indledning

Dette dokument indeholder principper for hvordan it-løsninger kan organiseres, udvikles, anskaffes og anvendes som forretningsunderstøttende, genbrugelige og fleksible services.

Principperne anvendes i relation til design af forretningservices og it-services.

Der er 12 principper:

1. Services understøtter forretningen
2. Services kan genbruges
3. Services kan kombineres
4. Services kobler løst
5. Services indkapsler data/funktionalitet
6. Services kan anvendes uafhængigt af platform
7. Services kan anvendes uafhængigt af lokation
8. Services er velbeskrevet
9. Services publiceres
10. Services versioneres
11. Services er baseret på standarder
12. Services returnerer sigende fejlmeddelelser.

De 12 principper er beskrevet nærmere med angivelse af rationale og implikationer nedenfor. Principperne kan også læses på OIO Arkitekturguiden, hvor der er supplerende information, der er relevant i forhold til principperne.

Link: <http://arkitekturguiden.digitaliser.dk/node/708>

Baggrund

Serviceprincipperne er udviklet i fællesoffentligt regi. Nærværende udgave af principperne er en revideret udgave af de principper, der blev formuleret i publikationen "Serviceorienteret arkitektur - Hvad og hvorfor" (ITST, 2006).

Principperne er i december 2013 blevet revideret af Digitaliseringsstyrelsen bl.a. med henblik på at sikre deres anvendelighed i gennemførelsen af Grunddataprogrammet.

Denne revision omfatter

- Titler og formuleringer for hvert princip er revideret, så de er nemme at forstå
- Rationale og implikationer for hvert enkelt princip er tydeliggjort, så det er nemmere at vurdere princippet i en konkret projektkontekst
- Der er tilføjet et princip om fejlmeddelelser
- Rækkefølgen af principperne er ændret, så de fremstår i en klarere indbyrdes sammenhæng.

Serviceprincipperne publiceres på [OIO arkitekturguiden på digitaliser.dk](http://arkitekturguiden.digitaliser.dk).



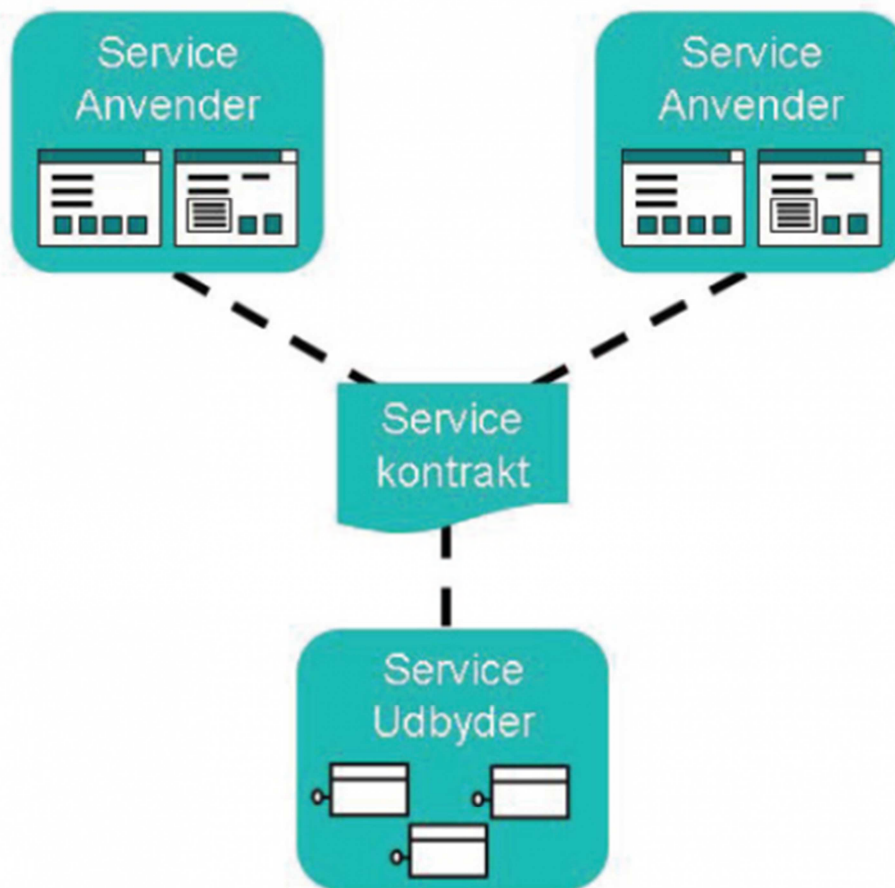
Digitaliseringsstyrelsen vil bestræbe sig på at holde principperne aktuelle og relevante.

Har du kommentarer, fx erfaringer med eller forslag til ændringer til principperne, er du meget velkommen til at skrive til redaktionen eller i arkitekturguidens gruppe [OIO EA forum](#) på digitaliser.dk.

Kontekst for anvendelse

Serviceprincipperne er udviklet til at understøtte sammenhængende it-arkitektur, hvor it-løsninger kan organiseres, udvikles, anskaffes og anvendes som forretningsunderstøttende, genbrugelige og fleksible services.

Serviceprincipperne bygger grundlæggende på det simple forhold, at der findes en serviceleverandør, som udstiller en formåen eller evne til at udføre en veldefineret og afgrænset aktivitet, en service, og at der på den anden side findes en serviceanvender, som efterspørger denne service. Servicen er beskrevet af en servicekontrakt, som kan konkretiseres teknisk i en servicespecifikation eller forretningsmæssigt i en Service Level Agreement (SLA).



En del af de beskrevne principper overlapper hinanden, ligesom der er sammenhæng mellem mange af principperne. Principperne kan ikke umiddelbart anvendes enkeltvis som milepæle eller ved vurdering af it-arkitekturens modenhed.



Disse principper fokuserer på indholdet og opbygningen af services, men kommunikationen mellem services - altså hvordan systemerne taler sammen, er beskrevet i [servicemønstre](#).

1. Services understøtter forretningen

Principformulering

Services understøtter forretningens processer

Rationale

Efterlevelse af dette princip vil bidrage til, at services bliver redskaber, der sikrer, at forretningen og it går i takt og arbejder efter de samme mål – de forretningsmæssige mål.

En bedre forståelse af forretningens processer og den direkte afspejling af disse i services kan være medvirkende til, at it-understøttelsen hurtigere kan tilpasses, når der sker ændringer i forretningen.

Implikationer

Forståelse og modellering af forretningsprocesser og forretningsbegreber bliver centrale aktiviteter i forbindelse med identificering, afgrænsning, fremstilling og ikke mindst brug af services.

Det skal bemærkes, at services også kan være af teknisk karakter. Tekniske services vil typisk understøtte den tekniske infrastruktur, eksempelvis print, arkivering, sikkerhed, netværk, osv.

2. Services kan genbruges

Principformulering

Services designes med genbrug for øje – også selv om en service ikke umiddelbart skal genbruges.

Rationale

Services kan ses som byggeklodser, der kan bruges i mange forskellige sammenhænge og sammensættes efter behov. Ved efterlevelse af dette princip kan man med en relativt lille indsats forberede eventuel senere genbrug. Andre serviceprincipper - eksempelvis "løst koblede services" og "services registreres og er synlige" - er i sig selv elementer, der fremmer muligheden for genbrug.

Genbrugelige services giver et potentiale for at øge den overordnede og tværgående produktivitet, f.eks. ved at der kan spares på indtastning og kvalitetssikring af data. Services med et højt genbrugspotentiale er som udgangspunkt mere værdifulde end services, hvor der forventes at være en lav grad af genbrug.



Implikationer

Dette betyder ikke, at services skal forberedes til alle tænkelige fremtidige anvendelsesmuligheder – hvilket jo selvsagt er umuligt. Det betyder derimod, at services bør udarbejdes med anvendelse af generelt accepterede arkitekturprincipper omkring generalisering og genbrug.

En mulig tilgang er en “service-first”-strategi, hvor man definerer hvilke services (byggeklodser), man har brug for, før man definerer det samlede system.

Der er selvfølgelig et økonomisk aspekt, hvis det koster ekstra at gøre servicen genbrugelig, ligesom der kan være høste/så problematikker, som man skal være opmærksom på i governancesammenhæng. Eksempelvis kan der i forhold til kapacitet og opetid være en udfordring, hvis en myndighed skal øge kapaciteten, mens en anden myndighed får gevinsten af servicen.

3. Services kan kombineres

Principformulering

Services kan sammensættes af andre services.

Rationale

Sammensætning af services giver stor fleksibilitet og mulighed for nye måder at it-understøtte forretningen på.

Sammensætning af services baserer sig på genbrug af allerede eksisterende services og kan således bidrage til højere produktivitet.

Kontekst

Dette princip muliggør en opdeling af services, hvor generelle standardiserede services gradvist sammensættes til mere komplekse services.

Såkaldte mash-ups er baseret på, at data fra forskellige, uafhængige kilder kan sammensættes til nye og uafhængige services med egen forretningslogik og funktionalitet. Et eksempel er, at stedbestemte data udstilles med en stedangivelse (fx koordinater eller adresse), så de nemt kan kombineres og indgå i nye forretningsammenhænge.

4. Services kobler løst

Principformulering

Services kobler systemer løst og med minimal afhængighed.

Rationale



En service skal kunne findes og anvendes under etablering af et minimum af afhængigheder mellem anvender og leverandør af en service.

En af de væsentligste fordele ved løst koblede services er, at det giver mulighed for at ændre i serviceimplementeringen, uden at det påvirker serviceanvenderne – dog under forudsætning af at servicekontrakten overholdes.

Løst koblede services giver også mulighed for at udskifte en service med en anden - igen under forudsætning af at servicekontrakten overholdes. Dette giver mulighed for at ibrugtage en service, der tilbyder andre og måske bedre kvaliteter, uden at det påvirker anvendelsen af servicen.

Implikationer

Servicekontrakten er den eneste fælles reference mellem anvender af en service og leverandøren af denne.

I praksis vil der dog i de fleste tilfælde være behov for en vis koordinering mellem en leverandør og anvendere af en service, specielt når der rent forretningsmæssigt er tale om et formelt eller tæt samarbejde.

Servicekontrakten skal overholdes ligesom der til nye versioner af servicen skal udarbejdes nye versioner af servicekontrakten.

Da anvenderen af en service ikke har kendskab til, hvordan servicen er implementeret, vil efterlevelse af dette princip være et væsentligt middel til at sikre en høj grad af løs kobling mellem serviceimplementeringen og serviceanvendelsen.

5. Services indkapsler data/funktionalitet

Principformulering

Services indkapsler data og funktionalitet fra det bagvedliggende system.

Rationale

Adskillelse af en service fra det bagvedliggende system gør det muligt at ændre i implementeringen, uden at det påvirker anvendere af servicen - blot servicekontrakten overholdes.

Adskillelse af service og systemimplementering giver serviceleverandøren væsentlig større frihed til at vedligeholde systemet uden at skulle inddrage og påvirke mulige anvendere af servicen.

Det kan samtidig være medvirkende til at give højere kvalitet, idet risikoen for utilsigtede sideeffekter bliver minimeret, da alle anvendere af en service bruger den samme offentliggjorte indgang til serviceimplementeringen.



Implikationer

Services er en abstraktion over forretningsfunktionalitet og information, der stilles til rådighed for serviceanvendere via en offentliggjort servicespecifikation.

Servicens funktionalitet er kun kendt og tilgængelig via det interface, den tilbyder. Servicen udgør dermed et abstraktionslag, der skjuler alle de komplicerede underliggende implementeringsdetaljer.

Specifikationen er servicens eksterne repræsentation, der realiseres i serviceimplementeringen. Da det bagvedliggende system kan være realiseret med væsentlig flere detaljer og anden struktur end specifikationen, vil det ofte være nødvendigt med en transformation som led i implementeringen af servicen, der oversætter fra det bagvedliggende system til noget, der er konsistent med specifikationen.

6. Services kan anvendes uafhængigt af platform

Principformulering

Anvendelse af en service bør kunne foregå uafhængigt af den platform, servicen er implementeret på.

Rationale

Platformsuafhængig anvendelse af services giver mulighed for at integrere forskellige systemer implementeret på forskellige platforme.

Det er samtidig medvirkende til at minimere omkostninger hos serviceanvender til implementering af services og vedligehold i forbindelse med versionering af services.

Implikationer

En platform er i denne sammenhæng en kombination af programmeringssprog, operativsystem, kommunikationsprotokoller m.m.

For at opnå en platformsuafhængig anvendelse af en service, er det nødvendigt at beskrive servicen i et platformsuafhængigt format, eksempelvis WSDL eller WADL, ligesom det er nødvendigt at anvende platformsuafhængige kommunikationsprotokoller, eksempelvis SOAP eller den REST-baserede AJAX tilgang.

Det er i denne sammenhæng væsentligt at understrege, at platformsuafhængig anvendelse af services ikke er ensbetydende med platformsuafhængige services. For at opnå platformsuafhængige services, stilles der yderligere krav til, at selve serviceimplementeringen er platformsuafhængig, så den frit kan flyttes fra platform til platform.



7. Services kan anvendes uafhængigt af lokation

Principformulering

Anvendelse af en service kan foregå uafhængigt af den fysiske lokation, servicen er implementeret på.

Rationale

Anvendelse af lokationsuafhængige services giver en række driftmæssige fordele i form af fleksibilitet og mulighed for at optimere driftsafviklingen.

Det giver, sammen med efterlevelsen af en række øvrige serviceprincipper, også mulighed for at vælge den konkrete forekomst af en service på afviklingstidspunktet.

Implikationer

Anvenderen behøver ikke at kende den lokation, hvorfra den enkelte service afvikles.

Lokationsuafhængig anvendelse af services kræver, at en service kaldes indirekte via sit navn eller anden form for logisk identitet, og at den fysiske lokation først bestemmes på afviklingstidspunktet.

Det kræver også, at implementeringen af servicen ikke på anden vis er bundet til en bestemt fysisk lokation.

Princippet understøtter bl.a. en cloudbaseret implementering.

8. Services er velbeskrevne

Principformulering

Services beskrives af en servicekontrakt, og serviceanvendelse sker på grundlag af denne.

Rationale

Dette princip medvirker til at forbedre kvaliteten i it-løsningerne, da risikoen for fejl og misforståelser bliver reduceret, når alle aktører er bekendt med deres præcise rettigheder og forpligtelser, når en service leveres eller anvendes.

Implikationer

Kontrakten kan have form af en servicespecifikation, som er en teknisk snitfladebeskrivelse, der indeholder en beskrivelse af dels den information, servicen har ansvaret for og manipulerer med, dels den adfærd servicen har. Specifikationen kan være beskrevet i et standardformat som fx

[WSDL](#) eller [WADL](#).



Kontrakten kan også beskrive de servicekvaliteter, som serviceleverandøren skal leve op til, og dermed også hvilke kvaliteter en serviceanvender kan forvente. Her bliver specifikationen i højere grad lig en kontrakt - eller en Service Level Agreement (SLA).

Endelig indeholder servicekontrakten også en specificering af, hvordan servicen fysisk skal findes og anvendes, samt eventuelle andre infrastrukturmæssige krav. I praksis stiller dette relativt store krav til beskrivelse, vedligeholdelse og versionering af servicekontrakten.

9. Services publiceres

Principformulering

Services skal registreres og publiceres i et servicekatalog.

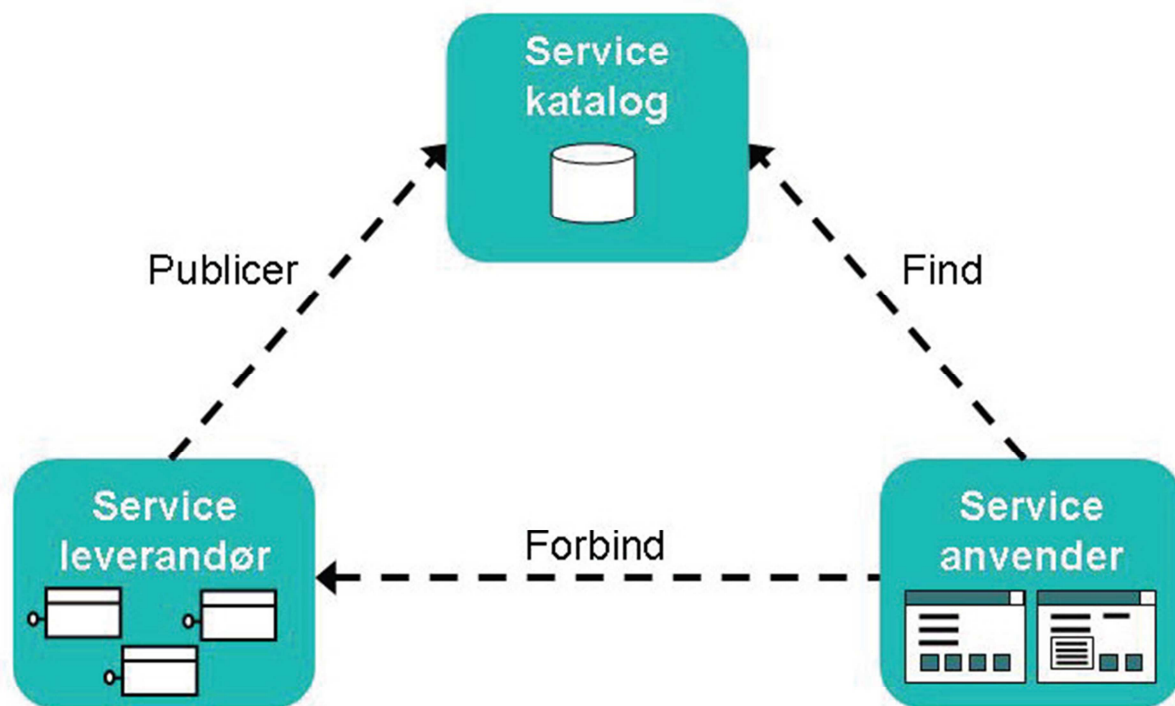
Rationale

Det gør det nemmere for anvendere at finde og anvende de rette services, når disse er registreret og publiceret på en ensartet måde i et servicekatalog.

Implikationer

Serviceleverandøren skal registrere og publicere servicespecifikationer i et servicekatalog. I servicekataloget skal serviceanvendere kunne finde de enkelte services, de skal bruge, og hente den information, der er nødvendig for at kunne anvende disse.

For at synliggøre services på en standardiseret måde, er det nødvendigt at stille krav til den information, der skal publiceres og til de mekanismer, der anvendes til publicering.



10. Services versioneres

Principformulering

Services, der ændres, skal versioneres, så det belaster serviceanvendere minimalt.

Rationale

Da en serviceleverandør ikke nødvendigvis har kendskab til alle serviceanvendere, er det i forbindelse med igangsætning af ændringer nødvendigt at versionere services.

Services skal kunne udvikle sig over tid uden at tvinge serviceanvenderen ind i omfattende og kostbare opgraderinger.

Implikationer

Flere versioner af den samme service kan eksistere på samme tid.

Services, der ændres, skal versioneres på en kontrolleret og gennemsigtig måde. Det betyder fx en klar versionsidentifikation (f.eks. et versionsnummer) og en beskrivelse af hver versions egenskaber og ændringer i forhold til tidligere versioner. Nye services / servicebeskrivelser publiceres i servicekataloget.

Serviceanvendere bør kunne abonnere på information om nye versioner eller udfasning af gamle.



Nye versioner af en service kan gradvist ibrugtages af forskellige serviceanvendere, efterhånden som der bliver brug for den ændrede funktionalitet.

Der kan naturligvis være lovmæssige krav, der medfører at alle serviceanvendere skal skifte serviceversion samtidig, og før eller siden skal alle anvendere givetvis under alle omstændigheder skifte til nyeste version, men versioneringen er medvirkende til, at dette kan ske fleksibelt i forhold til den enkelte anvender.

Versionering af services giver væsentlig bedre muligheder for fleksibel og effektiv videreudvikling, men stiller samtidig krav til infrastrukturen, der skal afvikle den serviceorienterede løsning, ligesom der stilles krav til implementeringen af de enkelte services.

11. Services er baseret på standarder

Principformulering

Services baseres på anvendelse af fælles begrebsdannelse og standarder.

Rationale

Anvendelse af standarder, både tekniske og forretningsmæssige, er afgørende for interoperabilitet og integration.

Implikationer

En standard er en specifikation, som der er enighed om at anvende. Standarder omfatter både tekniske specifikationer, der gør det muligt at fremstille, publicere, finde og anvende services på tværs af programmeringssprog og driftsplatforme, og forretningsmæssige specifikationer som eksempelvis taksonomier, standardiserede domænemodeller og referencemodeller, der beskriver begreber, aktører, opgaver, processer.

Fælles standarder kræver governance på rette niveau. Standarder kan være gældende lokalt for den enkelte offentlige myndighed, for en sektor eller på nationalt eller internationalt plan og det er vigtigt, at anvendelse og udarbejdelse af standarder foregår på det relevante niveau.

Hvor der skal sikres interoperabilitet på tværs af områder med hver sine standarder, kan det være nødvendigt at lave mapninger og transformationer mellem begrebsmodeller, datamodeller eller tekniske protokoller.

12. Services returnerer sigende fejlmeddelelser

Principformulering

Services skal returnere sigende fejlmeddelelser, som kan forstås umiddelbart af en bruger.



Rationale

Lettelse af fejlfinding i forbindelse med udvikling og tilpasning af applikationer, der anvender en service.

Implikationer

Serviceudbyderen skal opsætte en fejlhåndterings-mekanisme, som kan returnere en brugbar fejlmeddelelse til brugeren.

En fejlmeddelelse kan bestå af en fejlkode og en brugervendt besked.