


D0180 - SA-systemer - Tværgående

Version	Dato	Ansvarlig	Beskrivelse	Status
0.1	31 Oct 2019	D0180 - SA-systemer - Tværgående	Indført beskrivelse af fordeling af sync-services	Kladde
0.2	19 Dec 2019	D0180 - SA-systemer - Tværgående	Tilføjet historisk dokumentation af SA-integrationerne	Kladde
0.3	09 Jan 2020	D0180 - SA-systemer - Tværgående	Tilføjet historisk wsdl og skemes for Sync webservices	Kladde
0.4	12 Feb 2020	D0180 - SA-systemer - Tværgående	Tilføjet afklaringer for SyncMedarbejdere og SyncSkoledagskalendere	Kladde
0.5	09 Mar 2020	D0180 - SA-systemer - Tværgående	Tilføjet eksisterende valideringer for Synclokationer i efteruddannelse.	Kladde
0.6	03 Apr 2020	D0180 - SA-systemer - Tværgående	Tilføjet beskrivelse af operationer i SyncLokationer	Kladde
0.7	14 Apr 2020	D0180 - SA-systemer - Tværgående	Oprettet link til mødereferat for afklaring af valideringer i syncLokationer	Kladde
0.8	22 Apr 2020	D0180 - SA-systemer - Tværgående	Tilføjet dokumentation af SyncMedarbejder	Kladde
0.9	23 Apr 2020	D0180 - SA-systemer - Tværgående	Tilføjet dokumentation af commit logic i stamservices	Kladde
0.10	23 Apr 2020	D0180 - SA-systemer - Tværgående	Opdatering af generelle valideringer og synclokation valideringer til resultatet fra afklaringer + møder. Opdatering af tabelreferencer til at representere de som eksisterer i kursuskatalogen	Kladde
0.11	24 Apr 2020	D0180 - SA-systemer - Tværgående	Opdatering med beslutning om at ikke implementere EU-11 og EU-13	Kladde
0.12	09 Jun 2020	D0180 - SA-systemer - Tværgående	SyncBestilTilmeldingerTilAfhentning og SyncElever skal ikke videreføres fra EU.dk .	Kladde
0.13	03 Jul 2020	D0180 - SA-systemer - Tværgående	SyncTilmeldinger tilføjet	Kladde
0.14	06 Jul 2020	D0180 - SA-systemer - Tværgående	SyncSkoledagskalender tilføjet	Kladde
0.15	08 Sep 2020	D0180 - SA-systemer - Tværgående D0180 - SA-systemer - Tværgående D0180 - SA-systemer - Tværgående	SyncHold tilføjet	Kladde
0.16	09 Sep 2020	Mads Rangholm	<p>Markeret SyncSkolefag som udgået jævnfør</p> <p> ENPTV-3790 - Afklaring af behovet for SyncSkolefag RESOLVED og</p> <p>SyncBestilTilmeldingerTilAfhentning afklaret mundtligt med Marianne Guerry Larsen</p>	Kladde

0.17	02 Nov 2020	Simon Henriksen	Ændret fejltekst i Afhent-12 fejkoden i HentOpsamledeData så den er ens med det som er angivet i https://confluence.stil.dk/x/HwEYDQ	Kladde
0.18	23 Nov 2020	Jeppe Lillevang Salling	Tilføjet endpoints for udstillede services	Kladde
0.19	27 Nov 2020	Olivier Stéphane Bonde Petersen	Tilføjet beskrivelse af requests for SyncHold	Kladde
0.20	18 Dec 2020	Mads Rangholm	Færdiggjort udflytning af SyncServices til selvstændige sider samt færdiggjort beskrivelsen af tværgående design. Internt godkendt	NC Godkendt

Indholdsfortegnelse

- [Indholdsfortegnelse](#)
- [Introduktion](#)
 - [Overblik over migrering af SA-services fra EU.dk til ENPTV](#)
 - [Historisk dokumentation af SA-integrationerne \(CGI\)](#)
- [Integrationsmønstre](#)
- [Tværgående valideringer](#)
- [Tværgående behandling af requests](#)
 - [Eksekvering af operationer](#)
 - [Alt-eller-intet](#)

Introduktion

Denne side beskriver det overordnede design, for de services som udstilles specifikt til de StudieAdministrative (SA) systemer.

Historisk er disse services blevet udstillet af EfterUddannelse-systemet men leveres fremadrettet af to separate søster-systemer, navnlig KursusKataloget og Tilmelding, som del af voksenuddannelse.dk.

Hver enkelt service har sin egen underside, som detaljerer den specifikt.

Overblik over migrering af SA-services fra EU.dk til ENPTV

Webservices anvendt af SA-systemerne migreres til ENPTV undersystemer eller udgår som vist i tabellen nedenfor:

Service	Historisk	Aktuel		
	EU	Tilmelding	Kursuskatalog	Udgår
SyncBestilTilmeldingerTilAfhentning	X			X
SyncElever	X			X
SyncHentOpsamledeData	X			X
SyncHentOpsamledeDataV2	X	X		
SyncHentUdbud	X		X	
SyncHold	X		X	
SyncLokationer	X		X	
SyncMedarbejdere	X			X
SyncSkoledagskalendere	X		X	
SyncSkoleFag	X			X
SyncTilmeldinger	X	X		
SyncTilstededage	X	X		

Historisk dokumentation af SA-integrationerne (CGI)

Dokument	SOAP-Services beskrevet
Efteruddannelse uden EASY-A - Deleverance 1 - Design.docx	SyncLokationer, SyncSkoledagskalendere, SyncSkolefag, SyncMedarbejdere
Efteruddannelse uden EASY-A - Deleverance 2 - Design.docx	SyncHold
Efteruddannelse uden EASY-A - Deleverance 3A - Design.docx	SyncElever, SyncTilmeldinger, SyncTilstededage
Efteruddannelse uden EASY-A - Deleverance 3B - Design.docx	SyncTilmeldinger, SyncHentOpsamledeData, SyncBestilTilmeldingerTilAfhentning
Efteruddannelse uden EASY-A - Deleverance 4A - Design.docx	SyncHentUdbud

Integrationsmønster

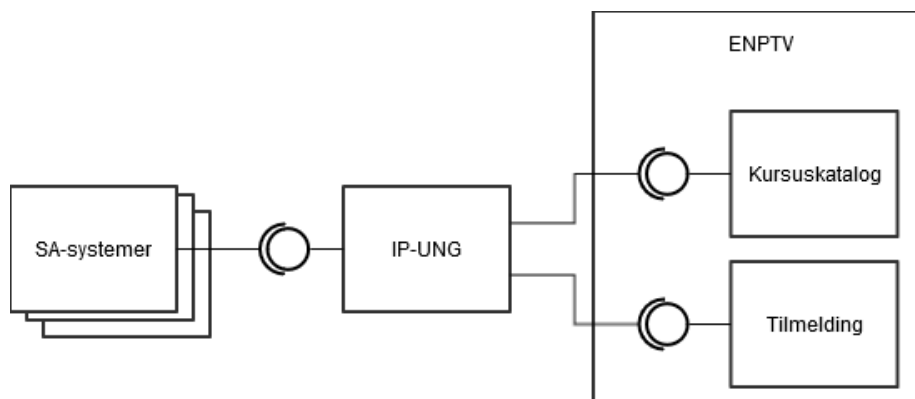
SA-systemerne kalder IP-UNG på service-endpoints som hver især er modelleret efter ENPTV's service-endpoints.

En forskel er at SA-systemerne benytter certifikat-sikkerhed til at identificere sig som system samt et institutionsnummer i selve request-payload.

IP-UNG validerer kombinationen af certifikat og institutionsnummer ift. om IP-UNG er bekendt med at skolen har givet SA-systemet lov til at kalde på deres vegne; en rettighedsmatrice.

Hvis SA-systemet har lov, mapper IP-UNG requestet til ENPTV's request-format, hvor IP-UNG udfylder institutionsnummeret, og afsender mod ENPTV.

Responset fra ENPTV mappes ligeledes den anden vej igennem IP-UNG og returneres til det kaldende SA-system.



Tværgående valideringer

Dette afsnit beskriver valideringer som går på tværs af alle SA-services.

Alle services deler en design-mønster, hvor svar på forespørgsler indeholder fejkoder på request-niveau, samt andre fejkoder på entitetsniveau, hvis sådanne indsendes i requestet.

Alle services, undtagen SyncHentOpsamledeData, benytter de samme fejkoder på request-niveau, som kan ses herunder i tabellen nedenfor.

Se [D0180 - SyncHentOpsamledeData](#) for komplet beskrivelse af request- såvel som entitetsfejkoder for denne service.

Fejkoder er nummerede, men kan ikke forventes at blive eksekveret i nummer-orden, hvorfor en anvender sagtens kan modtage koden "EU-10" for efterfølgende at modtage "EU-01".

Dette gør sig også gældende for fejkoder relateret til entiteter i de enkelte SA-services.

Fejlkode	Valideringsbrud i pseudokode	Fejltekst
EU-00	Denne anvendes som overordnet fejlkode, der indikerer at requestet er behandlet uden fejl.	Alle data er ajourført
EU-01	Denne anvendes som overordnet fejlkode for requestet Fejl i data. Generisk fejlkode som bliver brugt hvis der er en eller flere valideringsfejl i underliggende strukturer.	Der er fejl i data
EU-10	For hver webservice er der et max antal på, hvor mange <i>overordnede</i> elementer der må sendes i webservicen. Der er intet max antal for underliggende elementer. Skal kunne være konfigurerbart per service.	Der er #nnn elementer. Der må højst være#Max_antal_elementer
Skole-01	Værdien i tag InstNr findes ikke i tabellen amu_institution(tag InstNr)	Skole #InstNr kendes ikke

Tværgående behandling af requests

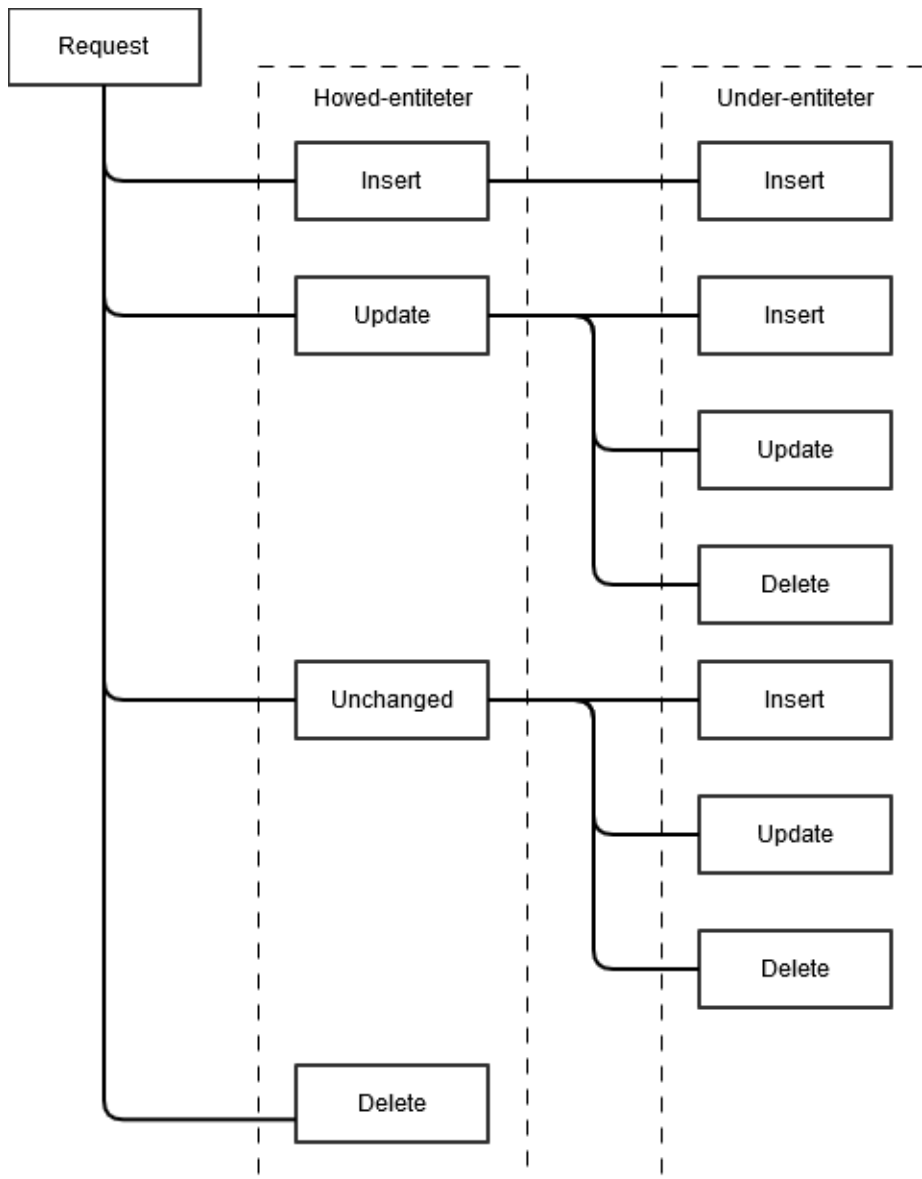
Requests mod en SA-service, SyncHentOpsamledeData undtaget, indeholder operationer som ønskes udført på ENTPV's datasæt.

Hver operation vedrør en entitet og alle hoved-entiteter og underliggende entiteter har deres egen operation.

En operation kan være følgende typer:

Operation	Beskrivelse
Insert	Indsættelse af entitet/underliggende entitet
Update	Opdatering af entitet/underliggende entitet
Delete	Sletning af entitet/underliggende entitet
Unchanged	Ingen ændringer, men entiteten benyttes som struktur til at placere operationer på underliggende entitet(er)

Eksempel på operationsombygning i et request:



Eksekvering af operationer

Her beskrives hvordan operationer i et request eksekveres i praksis, gennem udvalgte eksempler.

Operationer eksekveres i den rækkefølge, de er angivet i requestet-strukturen.

I nedenstående eksempler er entiteter angivet som "placeringer" hvor "1" er første hoved-entitet, "1.1" er første under-entitet under første hoved-entitet osv.

Eksempel: Oprettelse af 2 hoved-entiteter:

Flere operationer i et enkelt request kan godt vedrøre de samme hoved- og under-entiteter.

Når der er underliggende entiteter i et request, eksekveres operationer på disse via et top-down princip for at sikre eksekveringen af alle operationer.

Hoved-entiteter og alle deres underliggende operationer færdigbehandles før der fortsættes til næste hoved-entitet.

Placering i request	Operation	Eksekvering
1	Insert	1
1.1	Insert	2
1.1.1	Insert	4
1.2	Insert	3
2	Insert	5
2.1	Insert	6

Her ses at placeringen 1.2 udføres før placering 1.1.1 fordi dybden i strukturen for placering 1.2 er lavere, selvom placering 1.1.1 forekommer først i requestet.

Eksempel: Sletning kombineret med opdatering af underliggende entitet

Der er gyldigt for anvender både at bede om at slette et over-element, mens man ønsker at opdatere et under-element.

I dette eksempel vil sletningen af placering 1.1 igangsætte sletning af placering 1.1.1 hvilket betyder at operationen på placering 1.1.1 ignoreres, da placering 1.1.1 ikke kan eksistere uden placering 1.1.

Placering i request	Ønsket Operation	Eksekvering
1	Update	1
1.1	Delete	2
1.1.1	Update	-

Eksempel: 1 hoved-entitet, 4 sekvensielle operationer

Når anvendere sender lister af operationer, har de (teoretisk) mulighed for at udføre flere forskellige typer operationer på samme entitet. Dette kunne f.eks. være et hold, som blev oprettet ved en fejl og slettet igen, hvor de to handlinger fremsendes til SyncHold.

Placering i request	Ønsket Operation	Eksekvering
1	Insert	1
2	Update	2
3	Delete	3
4	Insert	4

I dette eksempel eksekveres operationerne i rækkefølge og på den vis kan et SA-system oprette, opdatere, slette og genoprette en hoved-entitet i et enkelt request. I dette tilfælde er rækkefølgen kritisk da alle operationer valideres ift. entitetens tilstand efter forrige operation.

Man kan f.eks. ikke slette eller opdatere noget, som ikke er der, eller indsætte oveni nogen eksisterende uden at møde valideringsfejl.

Dette begreb gælder også for underliggende entiteter, hvor rækkefølgen af operationer dikterer validering og eksekvering for den enkelte entitet.

Alt-eller-intet

I SA-services behandles alle operationer som en samlet handling som enten går godt eller skidt.

Det betyder i praksis at hvis 9 af 10 uafhængige operationer i et request går godt, men den sidste fejler, så rulles alle ændringer i data for alle 10 operationer tilbage til udgangspunktet og fejlkoder returneres.

Kun log bibevares.